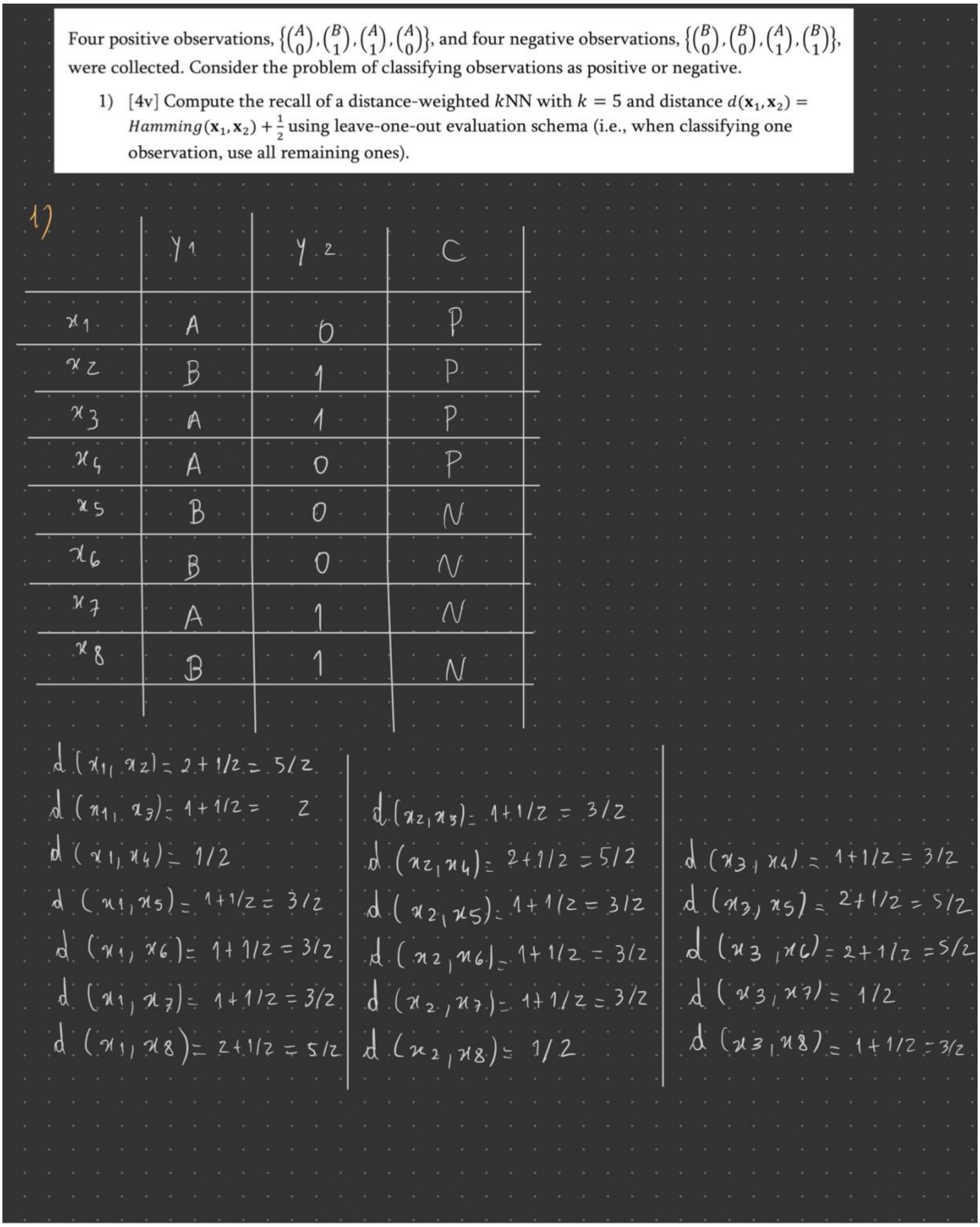
Aprendizagem 2021/22

Homework I – Group XXX

I. Pen-and-paper

1) Answer 1



Four positive observations, $\left\{\binom{A}{0},\binom{B}{1},\binom{A}{1},\binom{A}{0}\right\}$, and four negative observations, $\left\{\binom{B}{0},\binom{B}{0},\binom{A}{1},\binom{B}{1}\right\}$, were collected. Consider the problem of classifying observations as positive or negative.

1) [4v] Compute the recall of a distance-weighted kNN with $k = 5$ and distance $d(\mathbf{x}_1,\mathbf{x}_2) = Hamming(\mathbf{x}_1,\mathbf{x}_2) + \frac{1}{2}$ using leave-one-out evaluation schema (i.e., when classifying one observation, use all remaining ones).

1)

	y_1	y_2	C
x_1	A	0	P
x_2	B	1	P
x_3	A	1	P
x_4	A	0	P
x_5	B	0	N
x_6	B	0	N
x_7	A	1	N
x_8	B	1	N

$d(x_1, x_2) = 2 + 1/2 = 5/2$

$d(x_1, x_3) = 1 + 1/2 = \; 2$

$d(x_1, x_4) = 1/2$

$d(x_1, x_5) = 1 + 1/2 = 3/2$

$d(x_1, x_6) = 1 + 1/2 = 3/2$

$d(x_1, x_7) = 1 + 1/2 = 3/2$

$d(x_1, x_8) = 2 + 1/2 = 5/2$

$d(x_2, x_3) = 1 + 1/2 = 3/2$

$d(x_2, x_4) = 2 + 1/2 = 5/2$

$d(x_2, x_5) = 1 + 1/2 = 3/2$

$d(x_2, x_6) = 1 + 1/2 = 3/2$

$d(x_2, x_7) = 1 + 1/2 = 3/2$

$d(x_2, x_8) = 1/2$

$d(x_3, x_4) = 1 + 1/2 = 3/2$

$d(x_3, x_5) = 2 + 1/2 = 5/2$

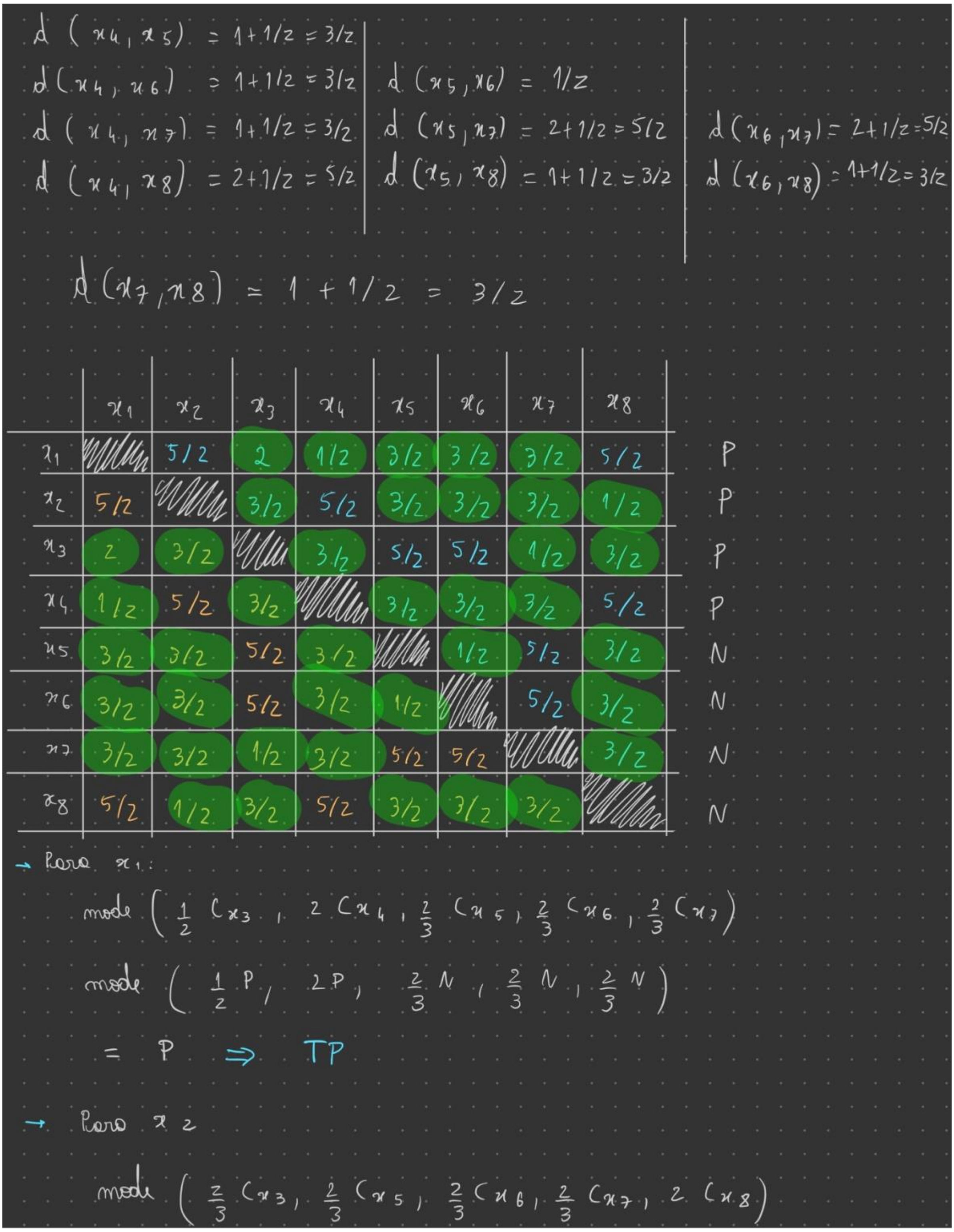
$d(x_3, x_6) = 2 + 1/2 = 5/2$

$d(x_3, x_7) = 1/2$

$d(x_3, x_8) = 1 + 1/2 = 3/2$

Aprendizagem 2021/22

Homework I – Group XXX



$d(x_4, x_5) = 1 + 1/2 = 3/2$

$d(x_4, x_6) = 1 + 1/2 = 3/2$

$d(x_4, x_7) = 1 + 1/2 = 3/2$

$d(x_4, x_8) = 2 + 1/2 = 5/2$

$d(x_5, x_6) = 1/2$

$d(x_5, x_7) = 2 + 1/2 = 5/2$

$d(x_5, x_8) = 1 + 1/2 = 3/2$

$d(x_6, x_7) = 2 + 1/2 = 5/2$

$d(x_6, x_8) = 1 + 1/2 = 3/2$

$d(x_7, x_8) = 1 + 1/2 = 3/2$

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

→ Para x_1:

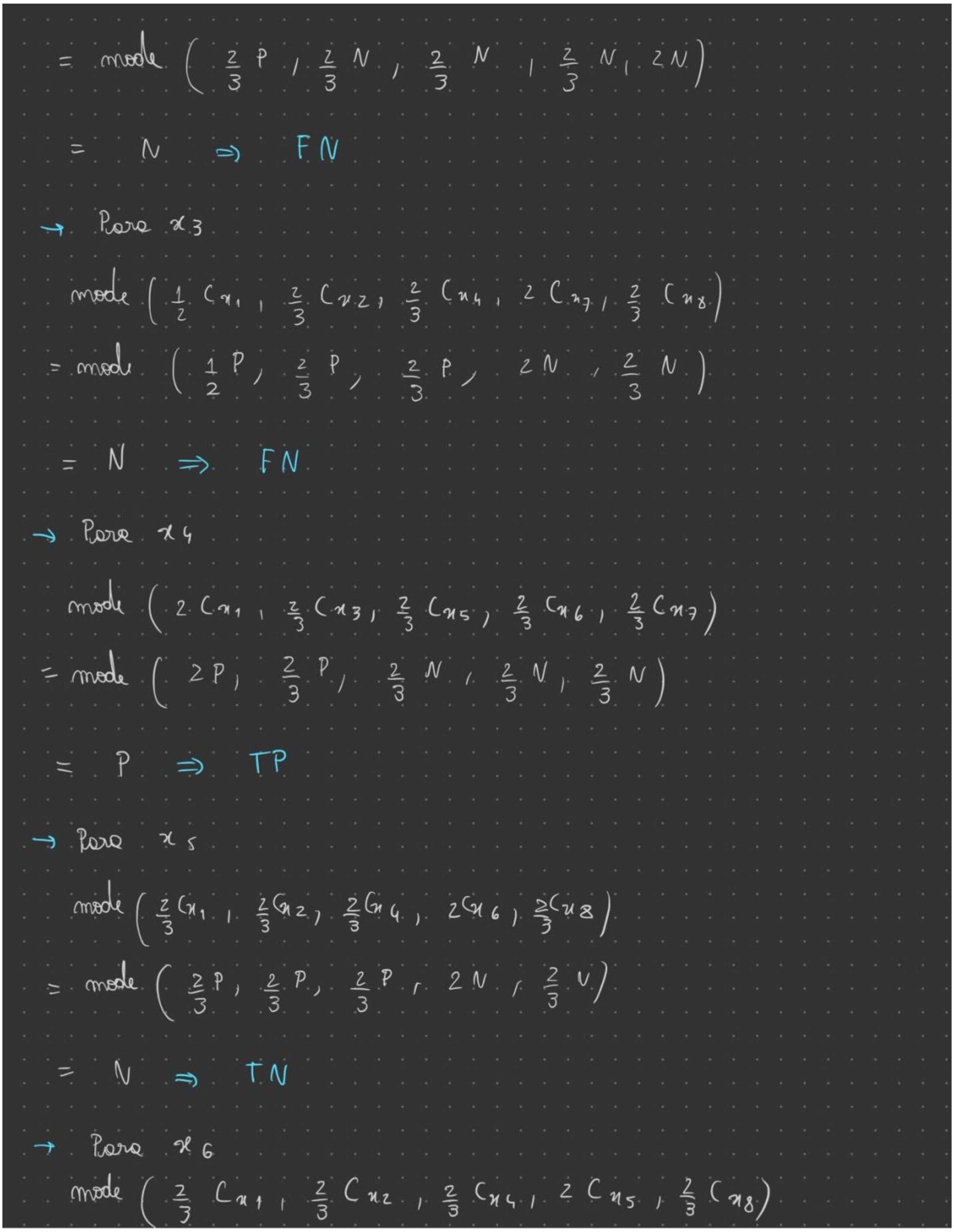
$\text{mode}\left(\frac{1}{2} C x_3,\ 2 C x_4,\ \frac{2}{3} C x_5,\ \frac{2}{3} C x_6,\ \frac{2}{3} C x_7\right)$

$\text{mode}\left(\frac{1}{2} P,\ 2 P,\ \frac{2}{3} N,\ \frac{2}{3} N,\ \frac{2}{3} N\right)$

$= P \Rightarrow TP$

→ Para x_2

$\text{mode}\left(\frac{2}{3} C x_3,\ \frac{2}{3} C x_5,\ \frac{2}{3} C x_6,\ \frac{2}{3} C x_7,\ 2\ C x_8\right)$



$= \text{mode}\left(\frac{2}{3}P, \frac{2}{3}N, \frac{2}{3}N, \frac{2}{3}N, 2N\right)$

$= N \Rightarrow FN$

→ Para x_3

$\text{mode}\left(\frac{1}{2}C_{x_1}, \frac{2}{3}C_{x_2}, \frac{2}{3}C_{x_4}, 2C_{x_7}, \frac{2}{3}C_{x_8}\right)$

$= \text{mode}\left(\frac{1}{2}P, \frac{2}{3}P, \frac{2}{3}P, 2N, \frac{2}{3}N\right)$

$= N \Rightarrow FN$

→ Para x_4

$\text{mode}\left(2C_{x_1}, \frac{2}{3}C_{x_3}, \frac{2}{3}C_{x_5}, \frac{2}{3}C_{x_6}, \frac{2}{3}C_{x_7}\right)$

$= \text{mode}\left(2P, \frac{2}{3}P, \frac{2}{3}N, \frac{2}{3}N, \frac{2}{3}N\right)$

$= P \Rightarrow TP$

→ Para x_5

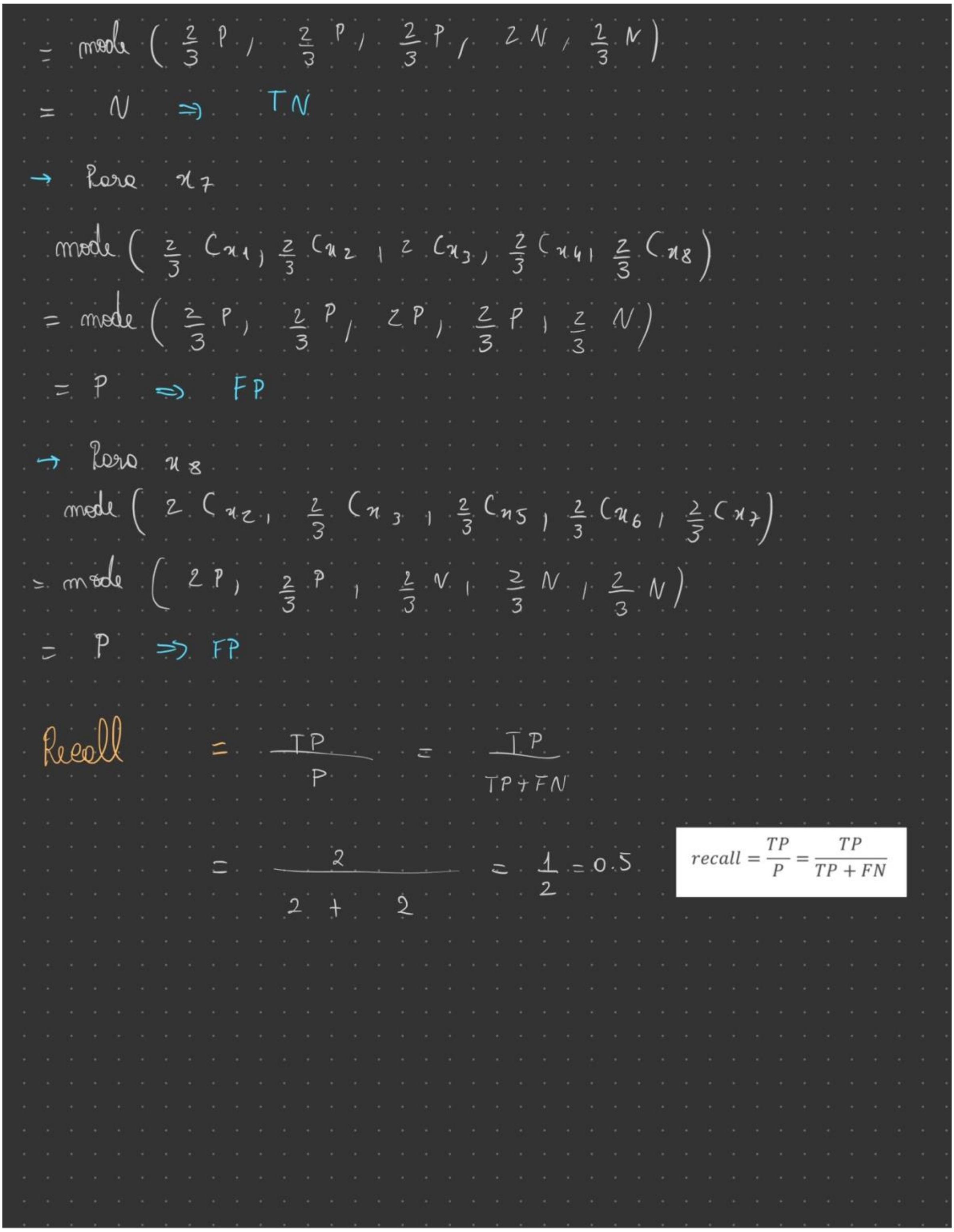
$\text{mode}\left(\frac{2}{3}C_{x_1}, \frac{2}{3}C_{x_2}, \frac{2}{3}C_{x_4}, 2C_{x_6}, \frac{2}{3}C_{x_8}\right)$

$= \text{mode}\left(\frac{2}{3}P, \frac{2}{3}P, \frac{2}{3}P, 2N, \frac{2}{3}V\right)$

$= N \Rightarrow TN$

→ Para x_6

$\text{mode}\left(\frac{2}{3}C_{x_1}, \frac{2}{3}C_{x_2}, \frac{2}{3}C_{x_4}, 2C_{x_5}, \frac{2}{3}C_{x_8}\right)$



$= \text{mode}\left(\frac{2}{3}P, \frac{2}{3}P, \frac{2}{3}P, 2N, \frac{2}{3}N\right)$

$= N \Rightarrow TN$

→ Para x_7

$\text{mode}\left(\frac{2}{3}C_{x_1}, \frac{2}{3}C_{x_2}, 2\,C_{x_3}, \frac{2}{3}C_{x_4}, \frac{2}{3}C_{x_8}\right)$

$= \text{mode}\left(\frac{2}{3}P, \frac{2}{3}P, 2P, \frac{2}{3}P, \frac{2}{3}N\right)$

$= P \Rightarrow FP$

→ Para x_8

$\text{mode}\left(2\,C_{x_2}, \frac{2}{3}C_{x_3}, \frac{2}{3}C_{x_5}, \frac{2}{3}C_{x_6}, \frac{2}{3}C_{x_7}\right)$

$= \text{mode}\left(2P, \frac{2}{3}P, \frac{2}{3}N, \frac{2}{3}N, \frac{2}{3}N\right)$

$= P \Rightarrow FP$

Recall $= \frac{TP}{P} = \frac{TP}{TP+FN}$

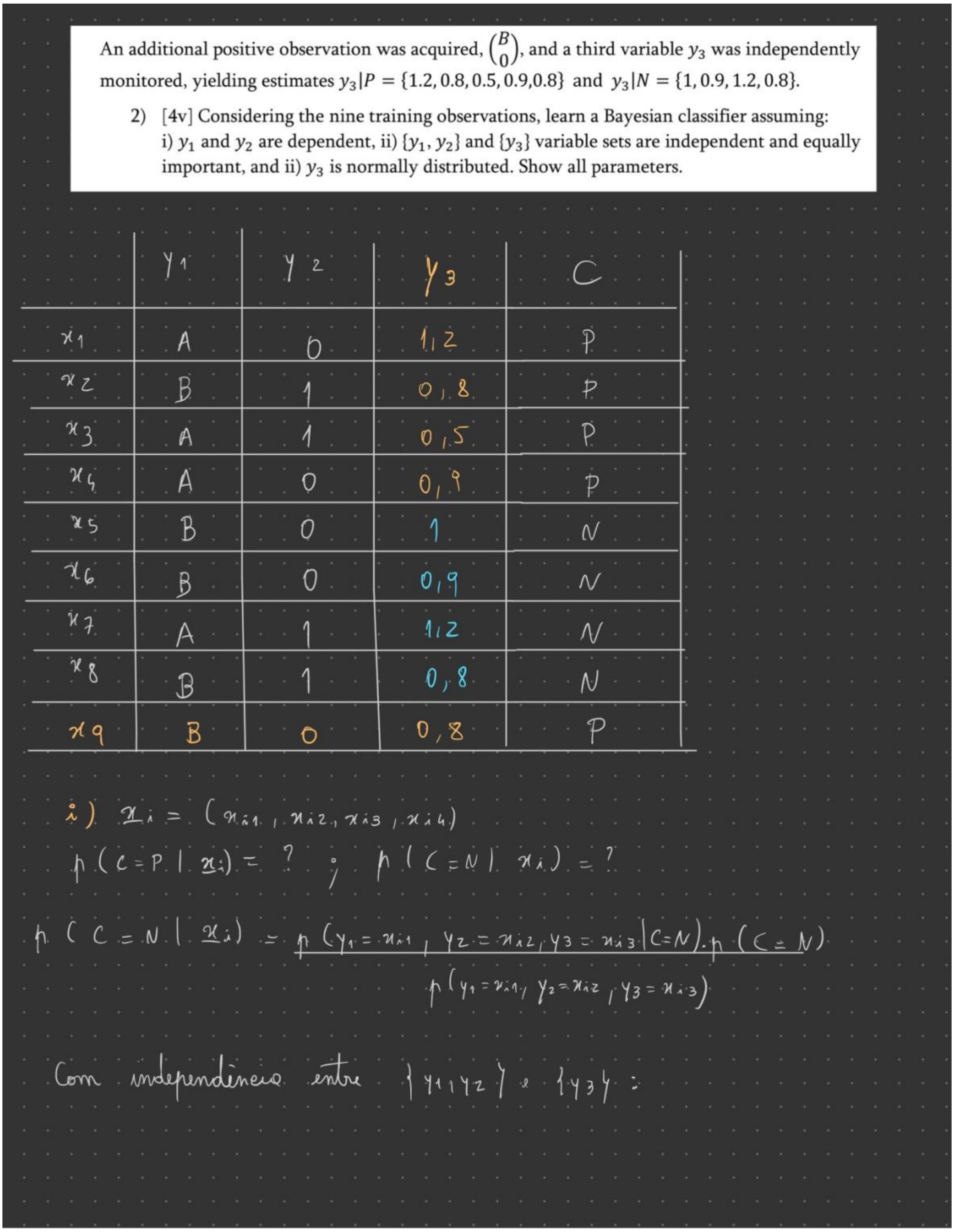
$= \frac{2}{2+2} = \frac{1}{2} = 0.5$

$$recall = \frac{TP}{P} = \frac{TP}{TP+FN}$$

Aprendizagem 2021/22

Homework I – Group XXX

2) Answer 2



An additional positive observation was acquired, $\binom{B}{0}$, and a third variable y_3 was independently monitored, yielding estimates $y_3|P = \{1.2, 0.8, 0.5, 0.9, 0.8\}$ and $y_3|N = \{1, 0.9, 1.2, 0.8\}$.

2) [4v] Considering the nine training observations, learn a Bayesian classifier assuming: i) y_1 and y_2 are dependent, ii) $\{y_1, y_2\}$ and $\{y_3\}$ variable sets are independent and equally important, and ii) y_3 is normally distributed. Show all parameters.

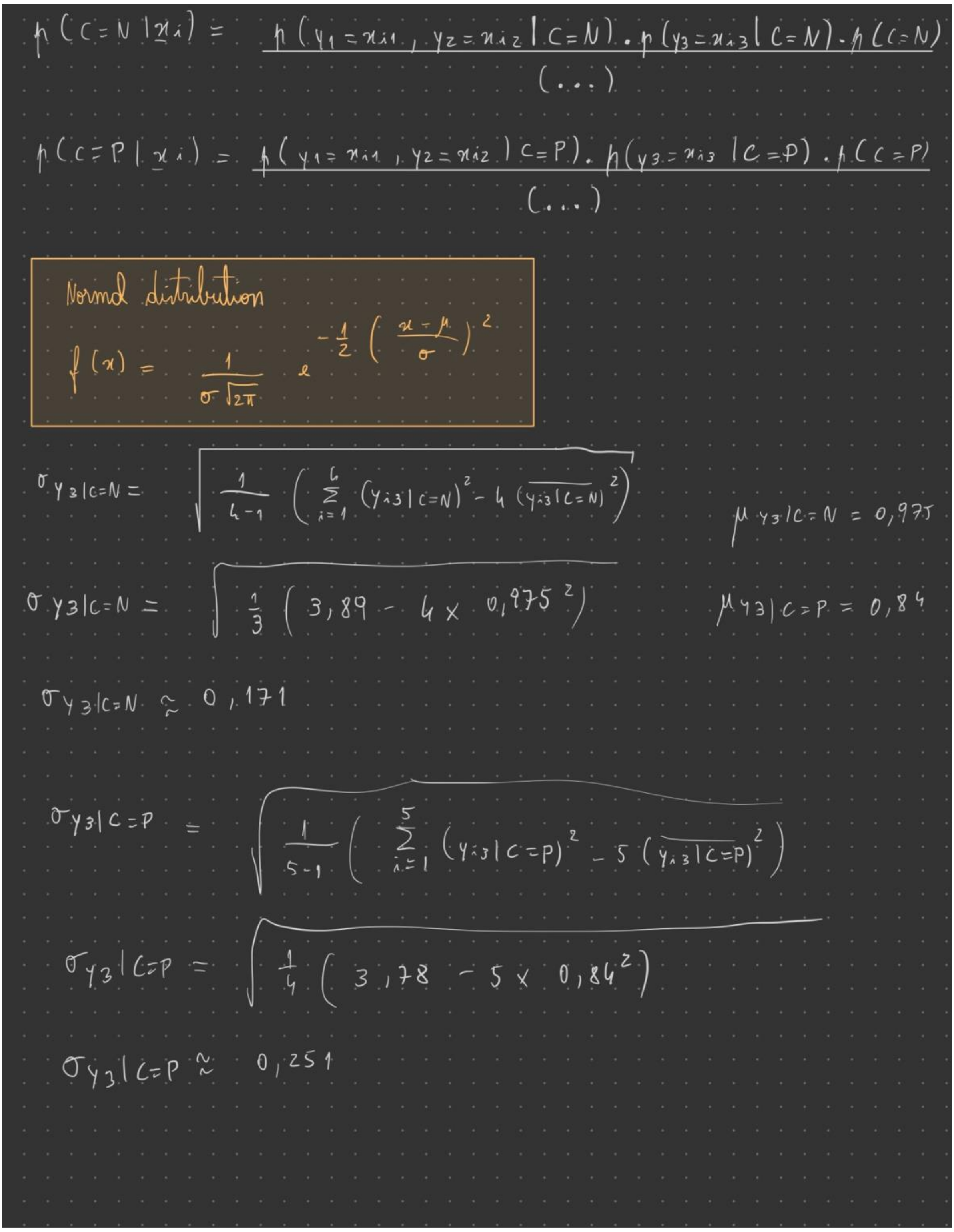
	y_1	y_2	y_3	C
x_1	A	0	1,2	P
x_2	B	1	0,8	P
x_3	A	1	0,5	P
x_4	A	0	0,9	P
x_5	B	0	1	N
x_6	B	0	0,9	N
x_7	A	1	1,2	N
x_8	B	1	0,8	N
x_9	B	0	0,8	P

i) $\underline{x}_i = (x_{i1}, x_{i2}, x_{i3}, x_{i4})$

$p(C=P \mid \underline{x}_i) = ?$; $p(C=N \mid x_i) = ?$

$$p(C=N \mid \underline{x}_i) = \frac{p(y_1 = x_{i1}, y_2 = x_{i2}, y_3 = x_{i3} \mid C=N) \cdot p(C=N)}{p(y_1 = x_{i1}, y_2 = x_{i2}, y_3 = x_{i3})}$$

Com independência entre $\{y_1, y_2\}$ e $\{y_3\}$:



$$p(C=N \mid \underline{x}_i) = \frac{p(y_1=x_{i1}, y_2=x_{i2} \mid C=N) \cdot p(y_3=x_{i3} \mid C=N) \cdot p(C=N)}{(\ldots)}$$

$$p(C=P \mid \underline{x}_i) = \frac{p(y_1=x_{i1}, y_2=x_{i2} \mid C=P) \cdot p(y_3=x_{i3} \mid C=P) \cdot p(C=P)}{(\ldots)}$$

Normal distribution

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

$$\sigma_{y_3|C=N} = \sqrt{\frac{1}{4-1}\left(\sum_{i=1}^{4} (y_{i3}|C=N)^2 - 4\,(\overline{y_{i3}|C=N})^2\right)}$$

$$\mu_{y_3|C=N} = 0{,}975$$

$$\sigma_{y_3|C=N} = \sqrt{\frac{1}{3}\left(3{,}89 - 4 \times 0{,}975^2\right)}$$

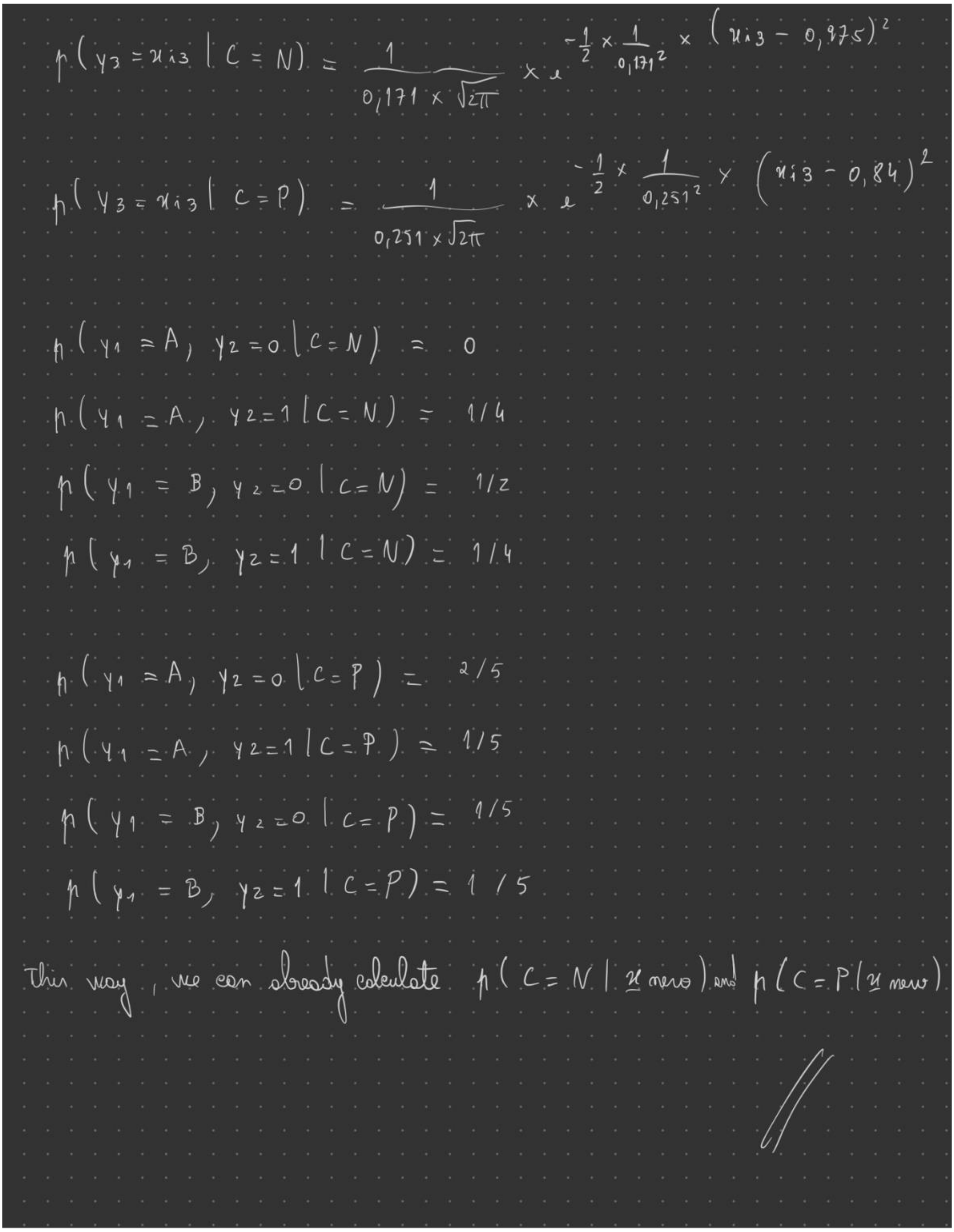
$$\mu_{y_3|C=P} = 0{,}84$$

$$\sigma_{y_3|C=N} \approx 0{,}171$$

$$\sigma_{y_3|C=P} = \sqrt{\frac{1}{5-1}\left(\sum_{i=1}^{5} (y_{i3}|C=P)^2 - 5\,(\overline{y_{i3}|C=P})^2\right)}$$

$$\sigma_{y_3|C=P} = \sqrt{\frac{1}{4}\left(3{,}78 - 5 \times 0{,}84^2\right)}$$

$$\sigma_{y_3|C=P} \approx 0{,}251$$



$$p(y_3 = x_{i3} \mid C = N) = \frac{1}{0{,}171 \times \sqrt{2\pi}} \times e^{-\frac{1}{2} \times \frac{1}{0{,}171^2} \times (x_{i3} - 0{,}975)^2}$$

$$p(y_3 = x_{i3} \mid C = P) = \frac{1}{0{,}251 \times \sqrt{2\pi}} \times e^{-\frac{1}{2} \times \frac{1}{0{,}251^2} \times (x_{i3} - 0{,}84)^2}$$

$$p(y_1 = A, y_2 = 0 \mid C = N) = 0$$

$$p(y_1 = A, y_2 = 1 \mid C = N) = 1/4$$

$$p(y_1 = B, y_2 = 0 \mid C = N) = 1/2$$

$$p(y_1 = B, y_2 = 1 \mid C = N) = 1/4$$

$$p(y_1 = A, y_2 = 0 \mid C = P) = 2/5$$

$$p(y_1 = A, y_2 = 1 \mid C = P) = 1/5$$

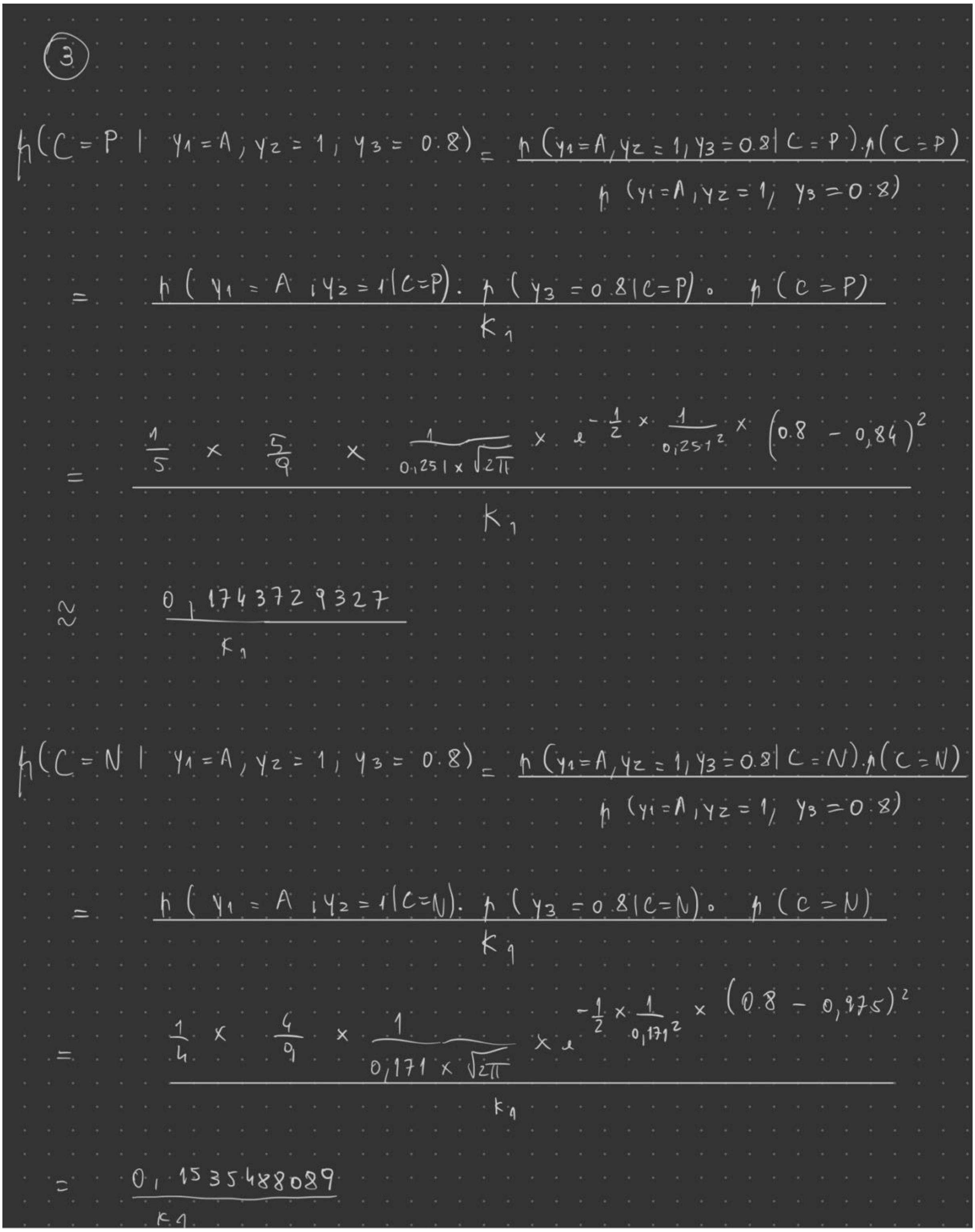
$$p(y_1 = B, y_2 = 0 \mid C = P) = 1/5$$

$$p(y_1 = B, y_2 = 1 \mid C = P) = 1/5$$

This way, we can already calculate $p(C = N \mid \underline{x}\ \text{new})$ and $p(C = P \mid \underline{x}\ \text{new})$

//

3) Answer 3



3

$$p(C=P \mid y_1=A; y_2=1; y_3=0.8) = \frac{p(y_1=A, y_2=1, y_3=0.8 \mid C=P) . p(C=P)}{p(y_1=A, y_2=1; y_3=0.8)}$$

$$= \frac{p(y_1=A; y_2=1 \mid C=P) . p(y_3=0.8 \mid C=P) \circ p(C=P)}{K_1}$$

$$= \frac{\frac{1}{5} \times \frac{5}{9} \times \frac{1}{0{,}251 \times \sqrt{2\pi}} \times e^{-\frac{1}{2} \times \frac{1}{0{,}251^2} \times (0.8 - 0{,}84)^2}}{K_1}$$

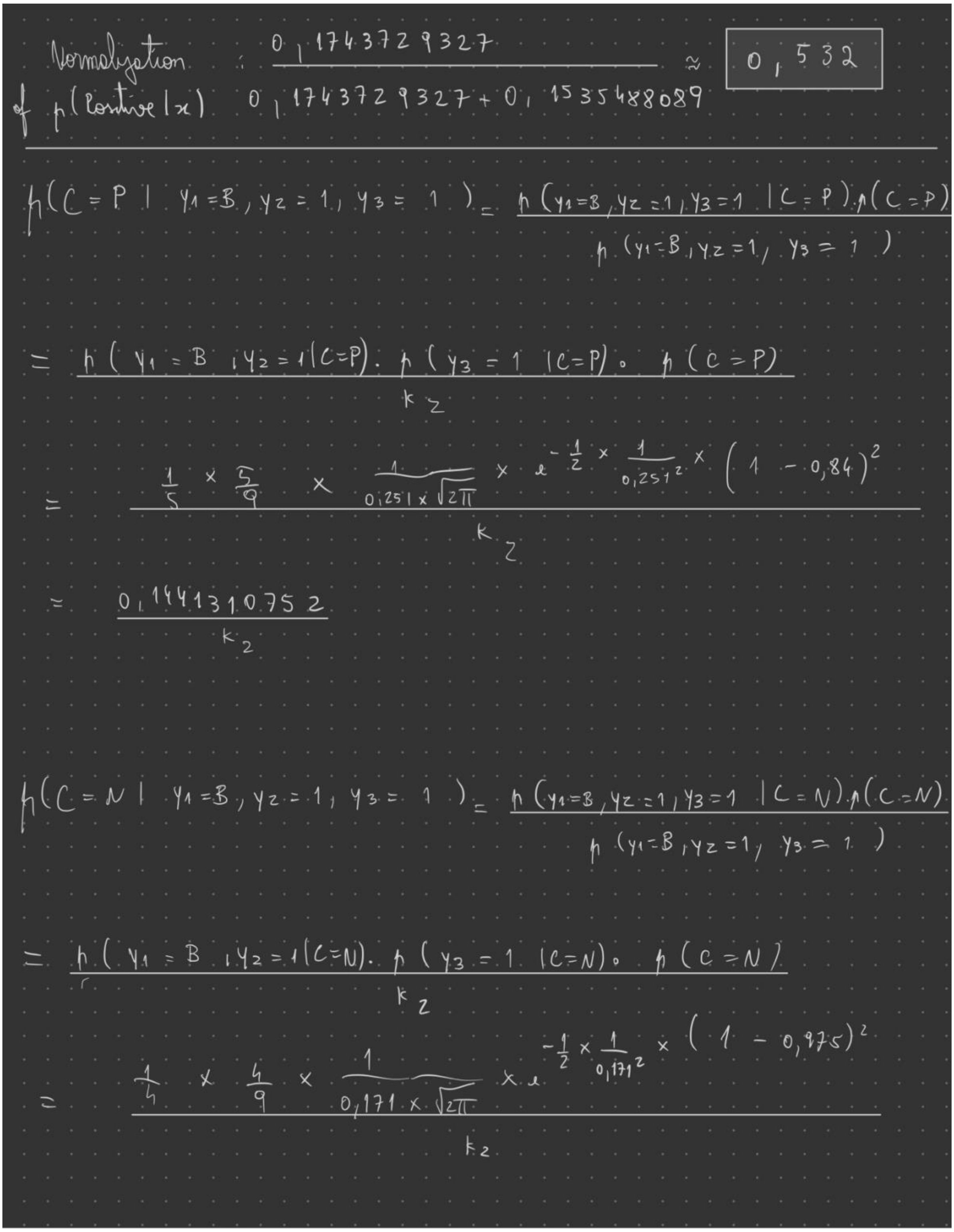
$$\approx \frac{0{,}1743729327}{K_1}$$

$$p(C=N \mid y_1=A; y_2=1; y_3=0.8) = \frac{p(y_1=A, y_2=1, y_3=0.8 \mid C=N) . p(C=N)}{p(y_1=A, y_2=1; y_3=0.8)}$$

$$= \frac{p(y_1=A; y_2=1 \mid C=N) . p(y_3=0.8 \mid C=N) \circ p(C=N)}{K_1}$$

$$= \frac{\frac{1}{4} \times \frac{4}{9} \times \frac{1}{0{,}171 \times \sqrt{2\pi}} \times e^{-\frac{1}{2} \times \frac{1}{0{,}171^2} \times (0.8 - 0{,}975)^2}}{K_1}$$

$$= \frac{0{,}1535488089}{K_1}$$



Normalization of $p(\text{Positive} \mid x)$: $\dfrac{0{,}1743729327}{0{,}1743729327 + 0{,}1535488089} \approx \boxed{0{,}532}$

$$p(C=P \mid y_1=B, y_2=1, y_3=1) = \frac{p(y_1=B, y_2=1, y_3=1 \mid C=P) \cdot p(C=P)}{p(y_1=B, y_2=1, y_3=1)}$$

$$= \frac{p(y_1=B, y_2=1 \mid C=P) \cdot p(y_3=1 \mid C=P) \circ p(C=P)}{k_2}$$

$$= \frac{\frac{1}{5} \times \frac{5}{9} \times \frac{1}{0{,}251 \times \sqrt{2\pi}} \times e^{-\frac{1}{2} \times \frac{1}{0{,}251^2} \times (1-0{,}84)^2}}{k_2}$$

$$\simeq \frac{0{,}1441310752}{k_2}$$

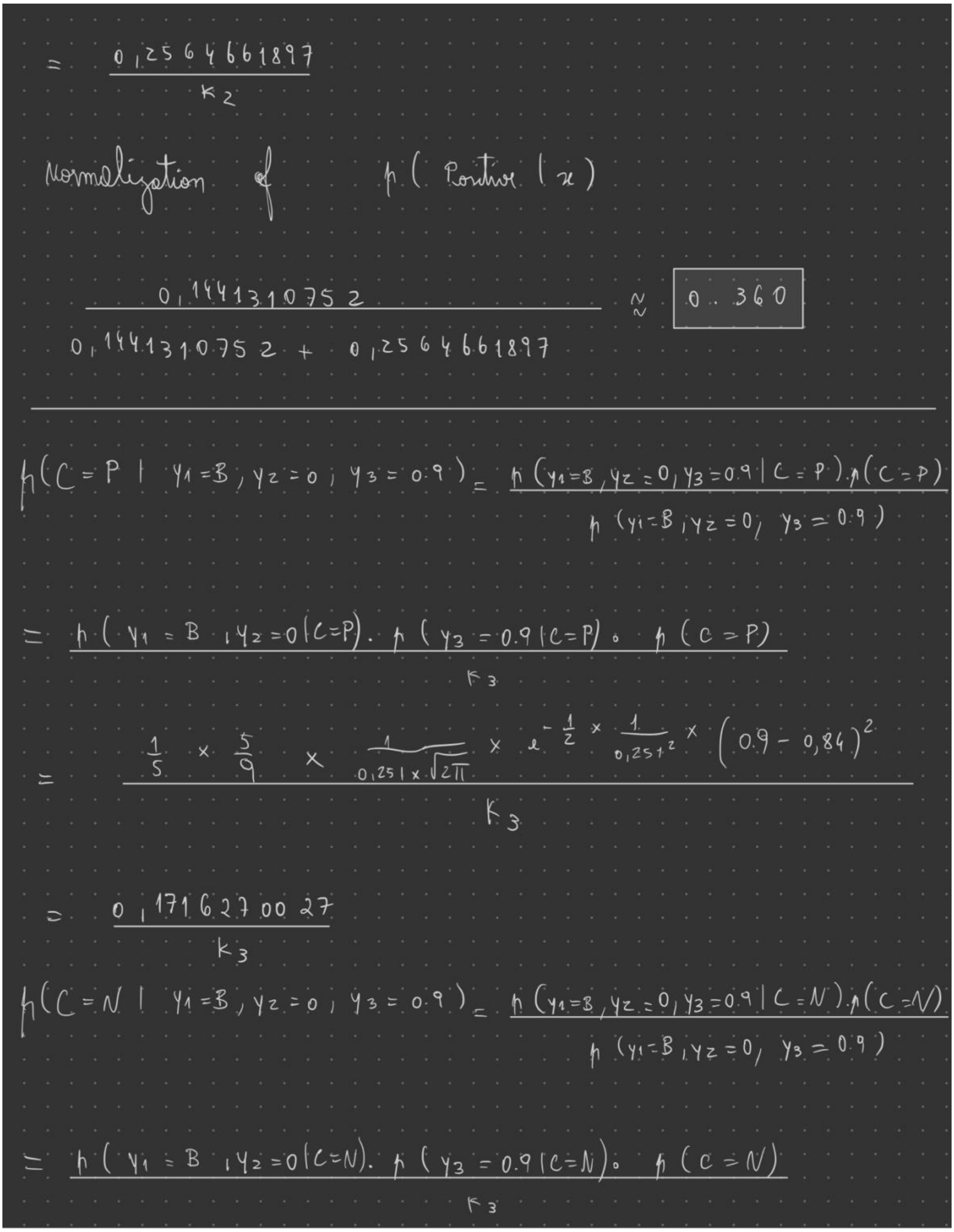
$$p(C=N \mid y_1=B, y_2=1, y_3=1) = \frac{p(y_1=B, y_2=1, y_3=1 \mid C=N) \cdot p(C=N)}{p(y_1=B, y_2=1, y_3=1)}$$

$$= \frac{p(y_1=B, y_2=1 \mid C=N) \cdot p(y_3=1 \mid C=N) \circ p(C=N)}{k_2}$$

$$= \frac{\frac{1}{4} \times \frac{4}{9} \times \frac{1}{0{,}171 \times \sqrt{2\pi}} \times e^{-\frac{1}{2} \times \frac{1}{0{,}171^2} \times (1-0{,}975)^2}}{k_2}$$

Aprendizagem 2021/22

Homework I – Group XXX



$$= \frac{0,2564661897}{k_2}$$

Normalization of $p(\text{Positive} \mid x)$

$$\frac{0,1441310752}{0,1441310752 + 0,2564661897} \approx \boxed{0.360}$$

$$p(C = P \mid y_1 = B, y_2 = 0, y_3 = 0.9) = \frac{p(y_1 = B, y_2 = 0, y_3 = 0.9 \mid C = P) . p(C = P)}{p(y_1 = B, y_2 = 0, y_3 = 0.9)}$$

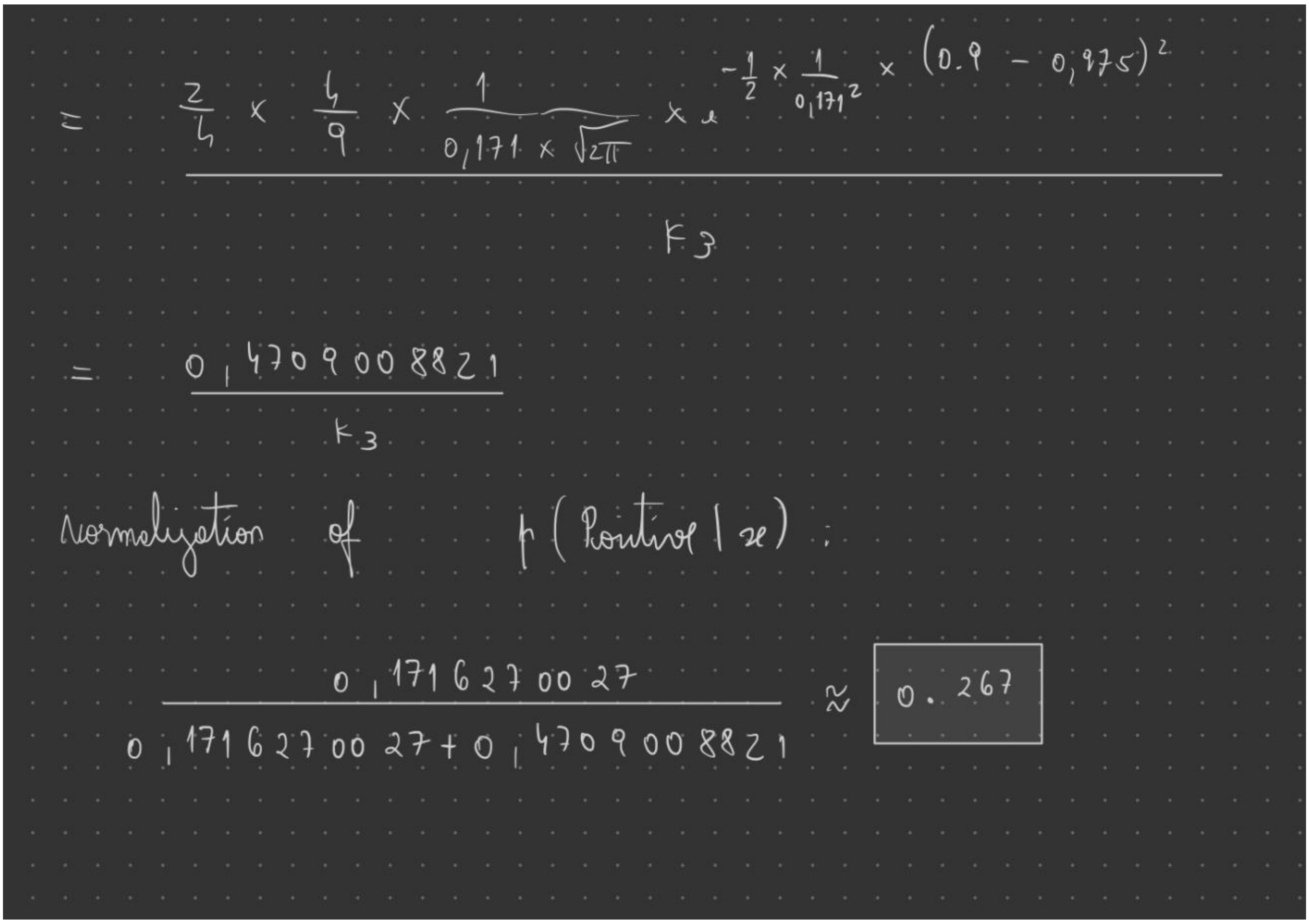
$$= \frac{p(y_1 = B, y_2 = 0 \mid C = P) . p(y_3 = 0.9 \mid C = P) \circ p(C = P)}{k_3}$$

$$= \frac{\frac{1}{5} \times \frac{5}{9} \times \frac{1}{0,251 \times \sqrt{2\pi}} \times e^{-\frac{1}{2} \times \frac{1}{0,251^2} \times (0.9 - 0,84)^2}}{k_3}$$

$$= \frac{0,1716270027}{k_3}$$

$$p(C = N \mid y_1 = B, y_2 = 0, y_3 = 0.9) = \frac{p(y_1 = B, y_2 = 0, y_3 = 0.9 \mid C = N) . p(C = N)}{p(y_1 = B, y_2 = 0, y_3 = 0.9)}$$

$$= \frac{p(y_1 = B, y_2 = 0 \mid C = N) . p(y_3 = 0.9 \mid C = N) \circ p(C = N)}{k_3}$$



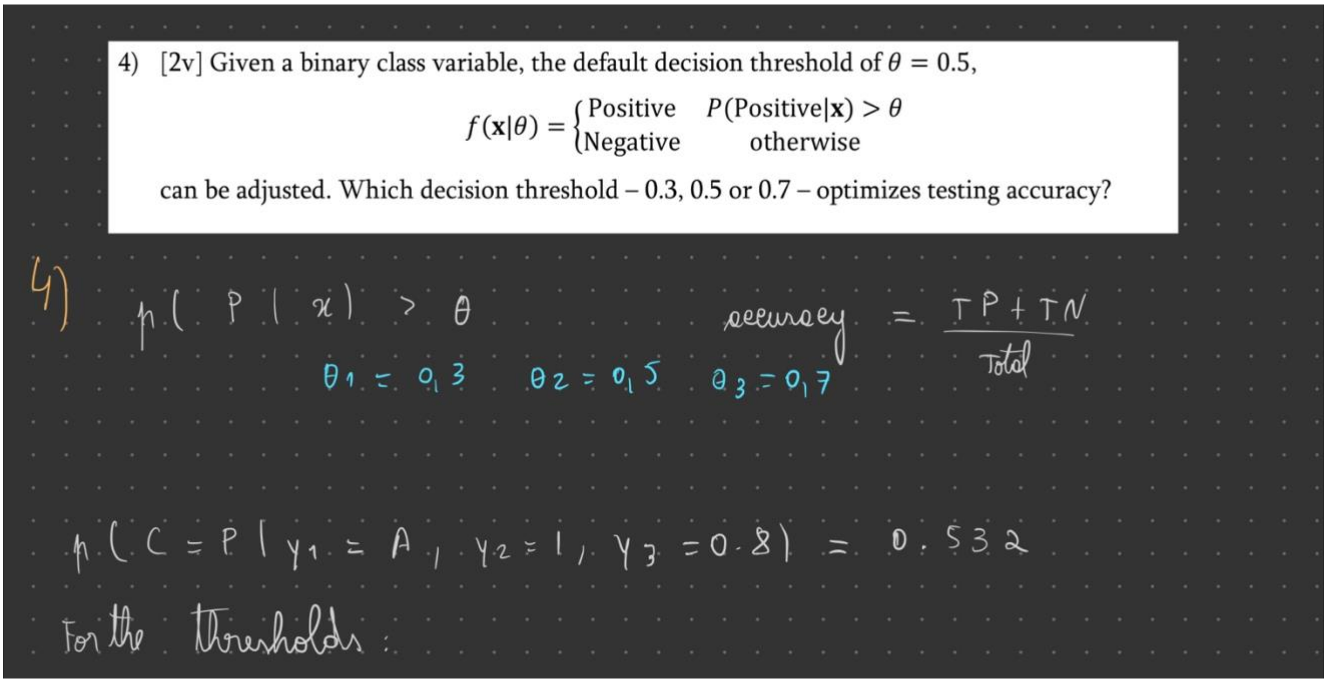
$$= \frac{\frac{2}{4} \times \frac{4}{9} \times \frac{1}{0{,}171 \times \sqrt{2\pi}} \times e^{-\frac{1}{2} \times \frac{1}{0{,}171^2} \times (0.9 - 0{,}975)^2}}{K_3}$$

$$= \frac{0{,}4709008821}{K_3}$$

normalization of $p(\text{Positive} \mid x)$:

$$\frac{0{,}1716270027}{0{,}1716270027 + 0{,}4709008821} \approx \boxed{0.267}$$

4) Answer 4



4) [2v] Given a binary class variable, the default decision threshold of $\theta = 0.5$,

$$f(\mathbf{x}|\theta) = \begin{cases} \text{Positive} & P(\text{Positive}|\mathbf{x}) > \theta \\ \text{Negative} & \text{otherwise} \end{cases}$$

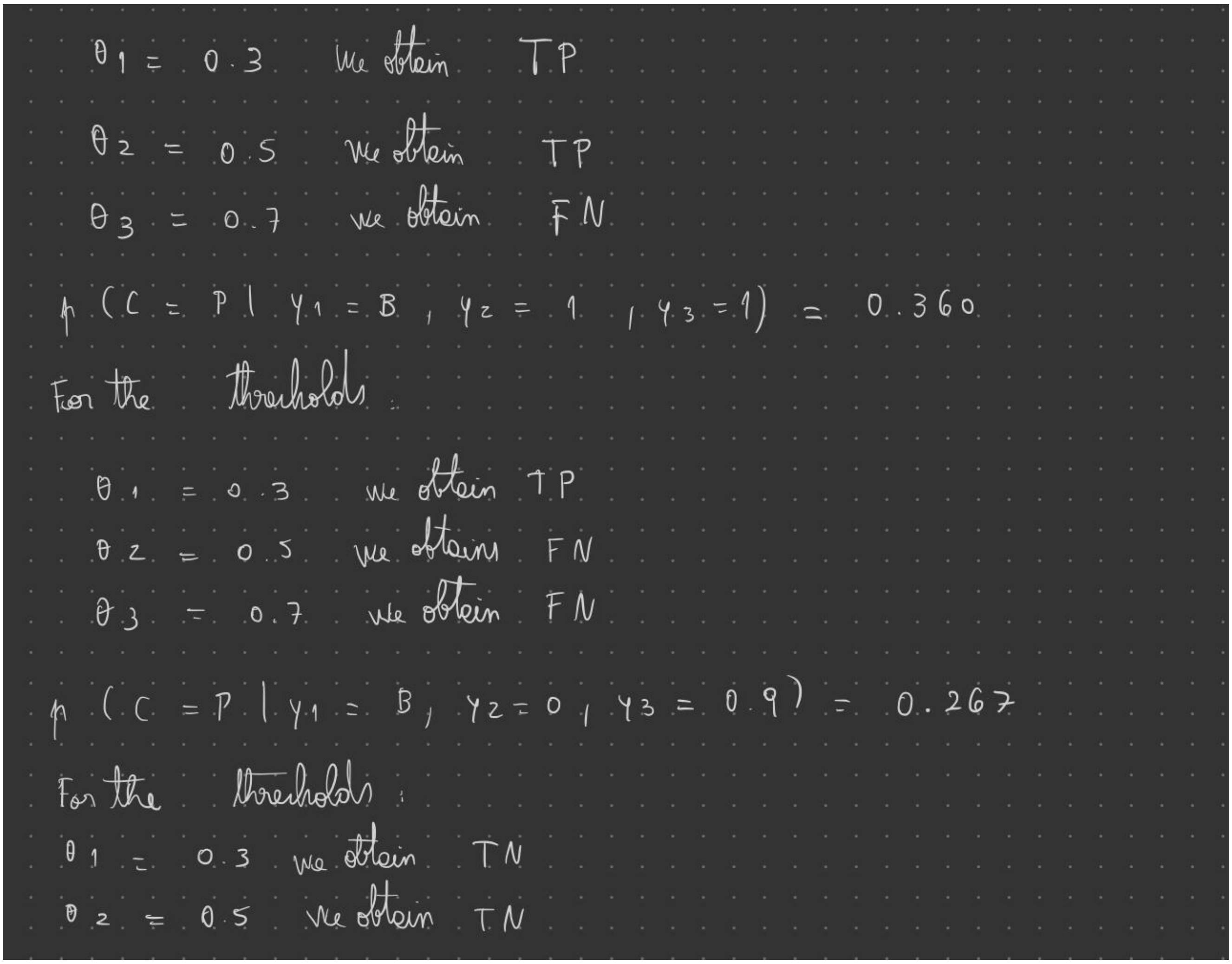
can be adjusted. Which decision threshold – 0.3, 0.5 or 0.7 – optimizes testing accuracy?

4) $p(P \mid x) > \theta$ accuracy $= \frac{TP + TN}{\text{Total}}$

$\theta_1 = 0{,}3$ $\theta_2 = 0{,}5$ $\theta_3 = 0{,}7$

$p(C = P \mid y_1 = A, y_2 = 1, y_3 = 0.8) = 0.532$

For the thresholds:



$\theta_1 = 0.3$ we obtain TP

$\theta_2 = 0.5$ we obtain TP

$\theta_3 = 0.7$ we obtain FN

$p(C = P \mid y_1 = B, y_2 = 1, y_3 = 1) = 0.360$

For the thresholds:

$\theta_1 = 0.3$ we obtain TP

$\theta_2 = 0.5$ we obtains FN

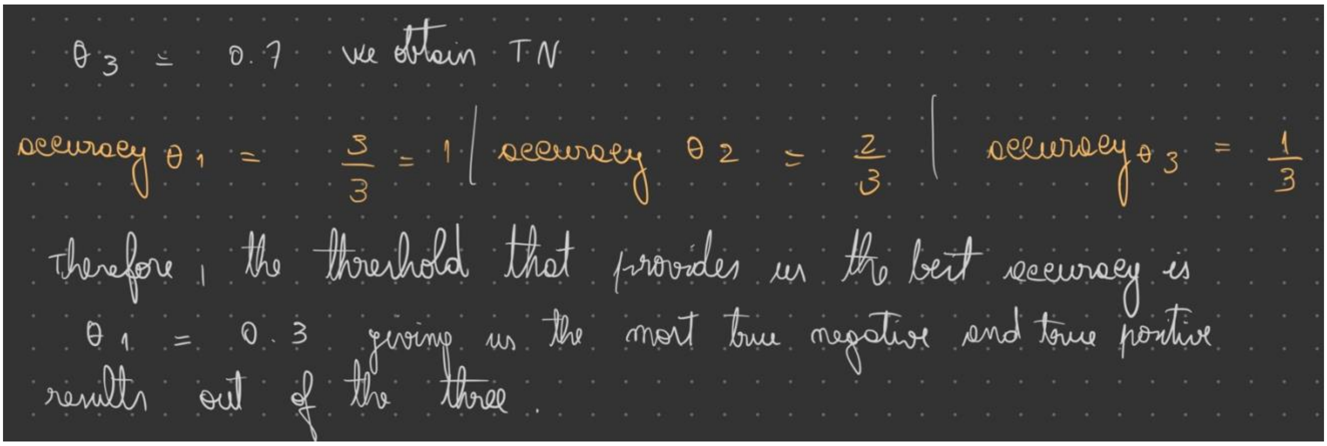
$\theta_3 = 0.7$ we obtain FN

$p(C = P \mid y_1 = B, y_2 = 0, y_3 = 0.9) = 0.267$

For the thresholds:

$\theta_1 = 0.3$ we obtain TN

$\theta_2 = 0.5$ we obtain TN



$\theta_3 = 0.7$ we obtain TN

accuracy $\theta_1 = \frac{3}{3} = 1$ | accuracy $\theta_2 = \frac{2}{3}$ | accuracy $\theta_3 = \frac{1}{3}$

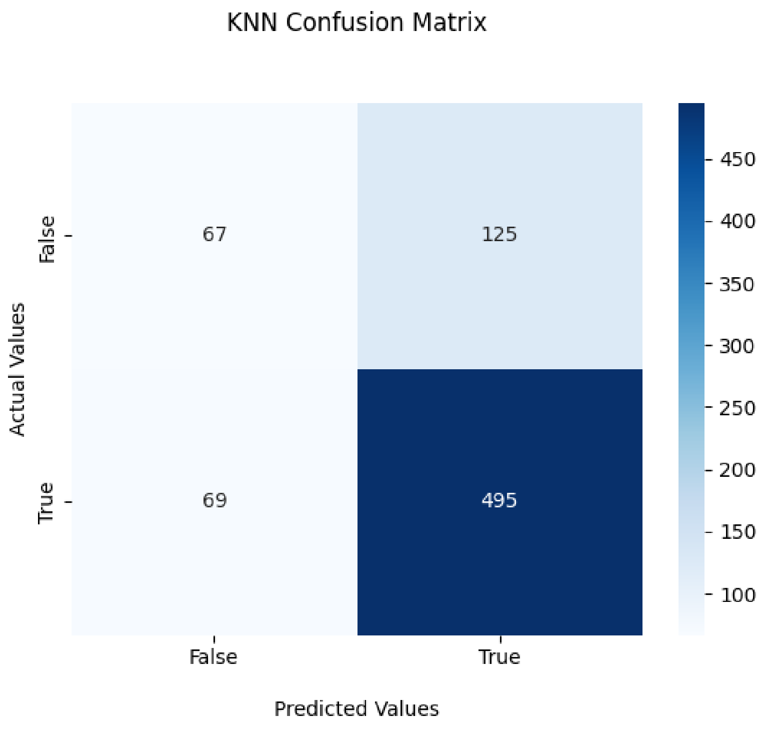
Therefore, the threshold that provides us the best accuracy is $\theta_1 = 0.3$ giving us the most true negative and true positive results out of the three.

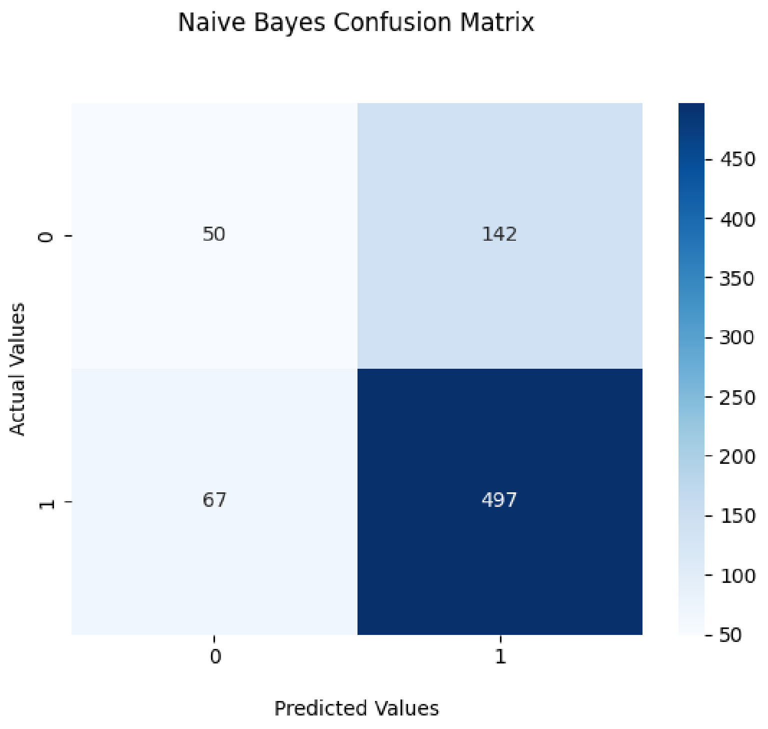
II. Programming and critical analysis

Aprendizagem 2021/22

Homework I – Group XXX

1) Answer 1


KNN Confusion Matrix


Naive Bayes Confusion Matrix

Aprendizagem 2021/22

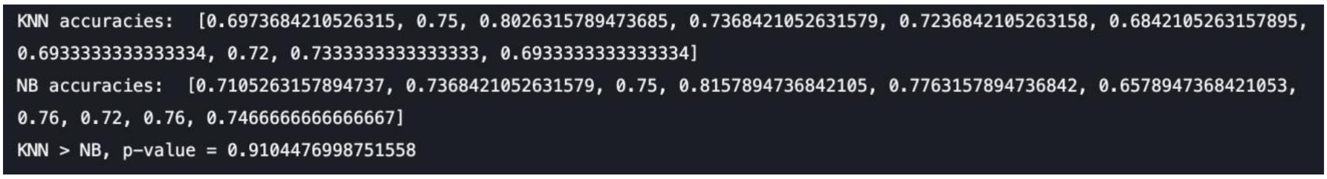
Homework I – Group XXX

2) Answer 2

Assuming the following hypothesis:

- H0: “kNN is statistically similar to Naïve Bayes regarding accuracy”
- H1: “*k*NN is statistically superior to Naïve Bayes regarding accuracy”

The obtained p-value reveals that we should reject H0 for significance levels approximately above 91.1%. The higher the p-value, the stronger evidence that the null hypothesis should not be rejected. Therefore, it is highly unlikely that kNN is statistically superior to Naïve Bayes regarding accuracy. Taking a look at the obtained accuracies for both kNN and Naïve Bayes, we can observe that the Naïve Bayes experiment displays higher overall accuracies which corroborates the obtained p-value.



```
KNN accuracies:  [0.6973684210526315, 0.75, 0.8026315789473685, 0.7368421052631579, 0.7236842105263158, 0.6842105263157895,
0.6933333333333334, 0.72, 0.7333333333333333, 0.6933333333333334]
NB accuracies:  [0.7105263157894737, 0.7368421052631579, 0.75, 0.8157894736842105, 0.7763157894736842, 0.6578947368421053,
0.76, 0.72, 0.76, 0.7466666666666667]
KNN > NB, p-value = 0.9104476998751558
```

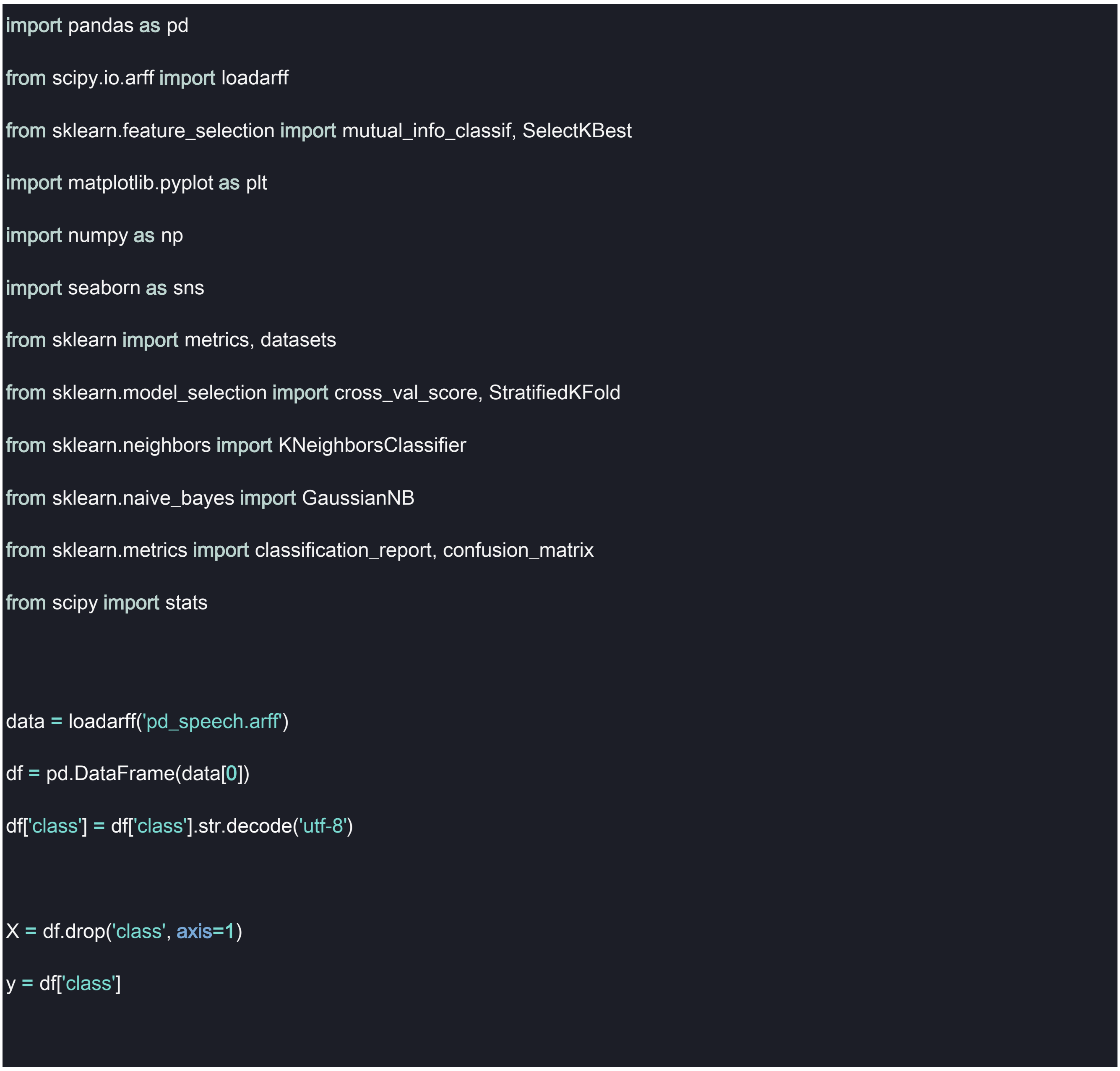
3) Answer 3

Most of the time, Naïve Bayes is highly accurate when applied to large amounts of data. These are some reasons why Naïve Bayes could outperform kNN in terms of accuracy:

- kNN struggles the most when the number of inputs is very large. With the increase of dimensions comes an exponential increase in the volume of the input space. In high dimensions, points that appear to have many similarities, might be far away from each other.
- In addition, kNN could lead to overfitting when using small values for k as the algorithm gets too accustomed to the training data. On the other hand, high values for k could lead to underfitting where no good predictive patterns are found in the training data.

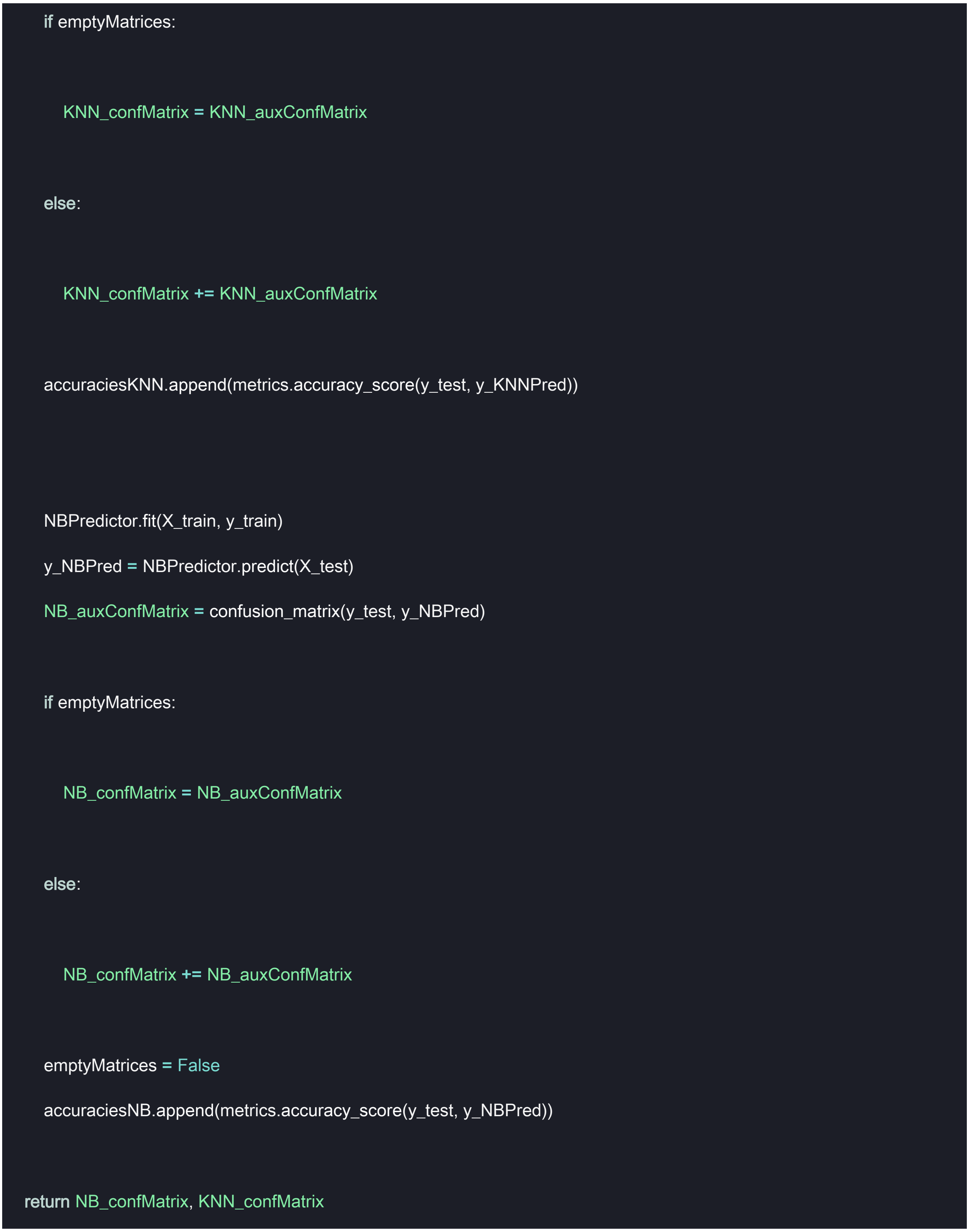
- Naïve Bayes, contrary to kNN, shines when solving multi-class prediction problems. If the independence assumption holds, a Naïve Bayes classifier performs better than most other models and less training data is required.
- Finally, Naïve Bayes works better when input variables are categorical, therefore, the more categorical variables, the more likely it is to outperform other models.

III. APPENDIX



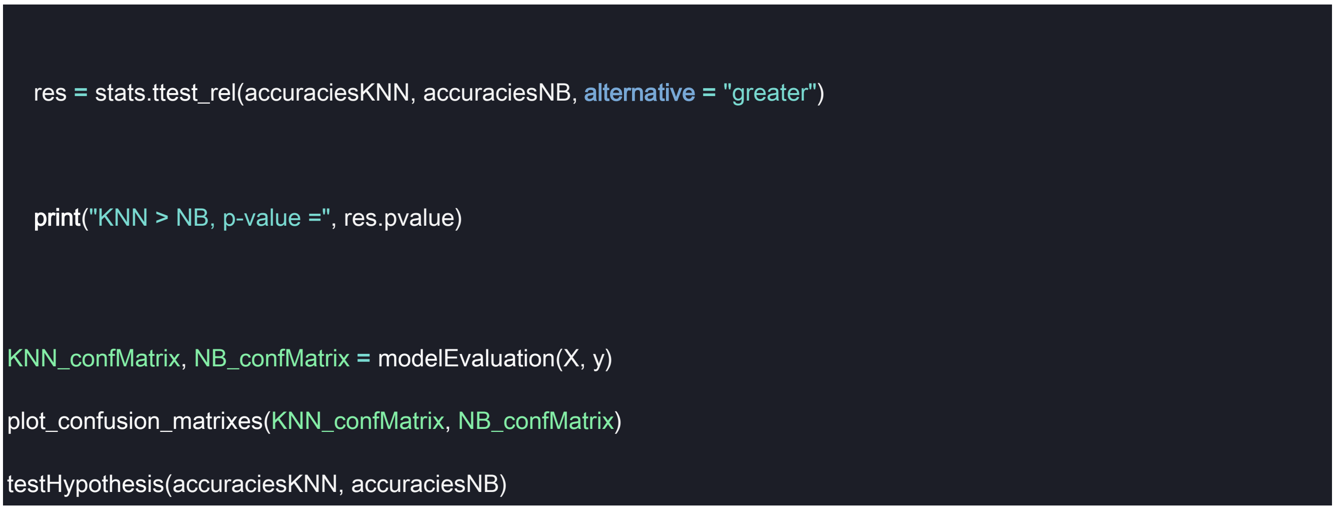
```
import pandas as pd
from scipy.io.arff import loadarff
from sklearn.feature_selection import mutual_info_classif, SelectKBest
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
from sklearn import metrics, datasets
from sklearn.model_selection import cross_val_score, StratifiedKFold
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import classification_report, confusion_matrix
from scipy import stats


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']
```

```
classNames = ['class']

accuraciesKNN = []

accuraciesNB = []



def modelEvaluation(X, y):


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


    KNN_confMatrix = np.zeros((2, 2))

    NB_confMatrix = np.zeros((2, 2))

    emptyMatrices = True


    KNNPredictor = KNeighborsClassifier(n_neighbors = 5, metric = 'euclidean', weights = 'uniform')

    NBPredictor = GaussianNB()


    splitFolds = folds.split(X, y)


    # iterate per fold

    for train_k, test_k in splitFolds:


        X_train, X_test = X.iloc[train_k], X.iloc[test_k]

        y_train, y_test = y.iloc[train_k], y.iloc[test_k]


        KNNPredictor.fit(X_train, y_train)

        y_KNNPred = KNNPredictor.predict(X_test)

        KNN_auxConfMatrix = confusion_matrix(y_test, y_KNNPred)
```



```
    if emptyMatrices:

        KNN_confMatrix = KNN_auxConfMatrix

    else:

        KNN_confMatrix += KNN_auxConfMatrix

    accuraciesKNN.append(metrics.accuracy_score(y_test, y_KNNPred))


    NBPredictor.fit(X_train, y_train)
    y_NBPred = NBPredictor.predict(X_test)
    NB_auxConfMatrix = confusion_matrix(y_test, y_NBPred)

    if emptyMatrices:

        NB_confMatrix = NB_auxConfMatrix

    else:

        NB_confMatrix += NB_auxConfMatrix

    emptyMatrices = False
    accuraciesNB.append(metrics.accuracy_score(y_test, y_NBPred))

return NB_confMatrix, KNN_confMatrix
```

```
def plot_confusion_matrixes(KNN_confMatrix, NB_confMatrix):

    ax1 = sns.heatmap(KNN_confMatrix, annot = True, fmt = "d", cmap = 'Blues')

    ax1.set_title('KNN Confusion Matrix\n\n');
    ax1.set_xlabel('\nPredicted Values')
    ax1.set_ylabel('Actual Values');


    ax1.xaxis.set_ticklabels(['False','True'])
    ax1.yaxis.set_ticklabels(['False','True'])

    plt.show()

    ax2 = sns.heatmap(NB_confMatrix, annot = True, fmt = "d", cmap = 'Blues')

    ax2.set_title('Naive Bayes Confusion Matrix\n\n');
    ax2.set_xlabel('\nPredicted Values')
    ax2.set_ylabel('Actual Values');

    plt.show()

def testHypothesis(accuraciesKNN, accuraciesNB):

    print("KNN accuracies: ", accuraciesKNN)
    print("NB accuracies: ", accuraciesNB)
```



```
    res = stats.ttest_rel(accuraciesKNN, accuraciesNB, alternative = "greater")

    print("KNN > NB, p-value =", res.pvalue)

KNN_confMatrix, NB_confMatrix = modelEvaluation(X, y)
plot_confusion_matrixes(KNN_confMatrix, NB_confMatrix)
testHypothesis(accuraciesKNN, accuraciesNB)
```

END