Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Range

Text Areas : PlainText

Possible Answers :

0.24 to 0.26

Question Number : 187 Question Id : 640653351400 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 1.5

Question Label : Short Answer Question

What is the value of w_{22}?

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Range

Text Areas : PlainText

Possible Answers :

0.80 to 0.82

MLP

Section Id :	64065322141
Section Number :	11
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	20
Number of Questions to be attempted :	20
Section Marks :	50
Display Number Panel :	Yes

Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	64065350426
Question Shuffling Allowed :	No

Question Number : 188 Question Id : 640653351417 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "MACHINE LEARNING PRACTICE"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?
CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE TOP FOR THE SUBJECTS REGISTERED BY YOU)

Options :

6406531166312. ✔ Yes

6406531166313. ✖ No

Sub-Section Number :	2
Sub-Section Id :	64065350427
Question Shuffling Allowed :	Yes

Question Number : 189 Question Id : 640653351432 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 1

Question Label : Multiple Choice Question

Which of the following vectorizer techniques can help to convert a large collection of text documents to a matrix?

Options :

6406531166357. ✖ CountVectorizer

6406531166358. ✔ HashingVectorizer

6406531166359. ✖ Both CountVectorizer and HashingVectorizer

6406531166360. ✖ None of these

Sub-Section Number : 3

Sub-Section Id : 64065350428

Question Shuffling Allowed : Yes

Question Number : 190 Question Id : 640653351418 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

What is the output of the following block of code?

```
from sklearn.metrics import confusion_matrix
y_true = [0,2,0,2]
y_pred = [0, 0, 2, 0]
confusion_matrix(y_true, y_pred)
```

Options :

6406531166314. ✔
```
array([[1, 1],
       [2, 0]])
```

6406531166315. ✖
```
array([[2, 0],
       [1, 1]])
```

6406531166316. ✖
```
array([[1, 1],
       [0, 2]])
```

6406531166317. ✖
```
array([[0, 2],
       [0, 2]])
```

Question Number : 191 Question Id : 640653351419 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Which of the following options represent the output score for the following block of code ?

```
from sklearn.linear_model import RidgeClassifier
X=[[0, 1],[0, 1]]
y=[[0,1],[0,1]]
clf = RidgeClassifier().fit(X, y)
clf.score(X, y)
```

Options :

6406531166318. ✖ 0.0

6406531166319. ✔ 1.0

6406531166320. ✖ -1.0

6406531166321. ✖ 0.5

Question Number : 192 Question Id : 640653351424 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Which of the options below represent the correct method to preprocess class labels in multi-class setup?

Options :

6406531166328. ✖

```
from sklearn.datasets import make_multilabel_classification as clf
def plot_2d(ax, n_labels=1, n_classes=3, length=50):
    X, Y, p_c, p_w_c = clf(
        n_samples=150,
        n_features=2,
        n_labels=n_labels,
        return_distributions=True,
        random_state32,
    )
```

6406531166329. ✖

```
from sklearn.datasets import make_multilabel_classification
from sklearn.multioutput import MultiOutputClassifier
from sklearn.linear_model import LogisticRegression
# Apply multi-label classification and logistic regression
X, y = make_multilabel_classification(n_classes=3, random_state=0)
clf = MultiOutputClassifier(LogisticRegression()).fit(X, y)
```

6406531166330. ✔
```
from sklearn.preprocessing import LabelBinarizer
# Using LabelBinarizer transformation
# to convert the class label to multi-class
y = np.array(['apple', 'pear', 'apple', 'orange'])
y_dense = LabelBinarizer().fit_transform(y)
# Note: The resulting label vector has shape of (n, k)
```

6406531166331. ✖ None of these

Question Number : 193 Question Id : 640653351428 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Which of the following options will be the output of the following code block?

```
from sklearn.feature_extraction.text import TfidfVectorizer
corpus = [
    'This is the first document.',
    'And is this the second one?',
    ]
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(corpus)
vectorizer.get_feature_names_out()
```

Options :

6406531166344. ✔ array(['and', 'document', 'first', 'is', 'one', 'second', 'the', 'this'], dtype=object)

6406531166345. ✖ array(['and', 'document', 'first', 'is', 'one', 'the', 'second', 'the', 'this', '?', '.'], dtype=object)

6406531166346. ✖

array(['and', 'first', 'is', 'meanwhile', 'one', 'second', 'text', 'the', 'third', 'this'], dtype=object)

6406531166347. ✖ array(['first', 'is', 'text', 'the', 'this', '?'], dtype=object)

Question Number : 194 Question Id : 640653351429 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Assume that X is a feature matrix with shape (1000,5) and y is the label vector with two classes: 0 and 1. Assume that 730 examples of training data belong to class 0. What will be the output of the following code?

```
base_clf = DummyClassifier(strategy='most_frequent')
base_clf.fit(X,y)
print(base_clf.score(X,y))
```

Options :

6406531166348. ✖ 0.65

6406531166349. ✔ 0.73

6406531166350. ✖ 0.37

6406531166351. ✖ None of these

Question Number : 195 Question Id : 640653351434 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

Rakesh is solving a regression problem with a KNN model. He has considered k = 3 for his model. Based on the graph shown below, what would you suggest him out of the following options?

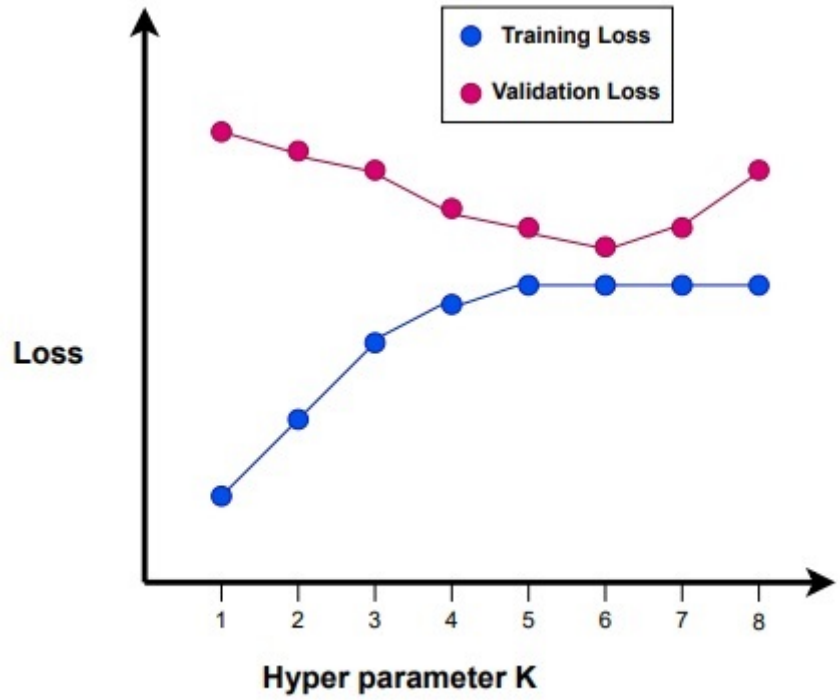

Figure: 3

Options :

6406531166365. ✔ His model is suffering from overfitting. So, he should increase the value of k.

6406531166366. ✖ His model is suffering from underfitting. So, he should increase the value of k.

6406531166367. ✖ His model is suffering from overfitting. So, he should decrease the value of k.

6406531166368. ✖ His model is suffering from underfitting. So, he should decrease the value of k.

6406531166369. ✖ No suggestions, Rakesh has already chosen the best value of k.

Question Number : 196 Question Id : 640653351440 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

/1 penalty in LinearSVC classifier leads to coef_vectors which are-

Options :

6406531166383. ✖ dense.

6406531166384. ✔ sparse.

6406531166385. ✖ Unique class labels.

6406531166386. ✖ /1 penalty can’t be set via a parameter in LinearSVC.

Sub-Section Number : 4

Sub-Section Id : 64065350429

Question Shuffling Allowed : Yes

Question Number : 197 Question Id : 640653351420 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Which of the following options will be the possible output for the given block of code?

```
from sklearn.linear_model import Perceptron
X = [[2, 2], [2, 4], [4,4], [4, 2]]
y = [1, 2, 1, 2]
clf = Perceptron(tol = None, random_state=0)
clf.fit(X, y)
print(clf.score(X, y))
```

Options :

6406531166322. ✖ 1

6406531166323. ✖ 0

6406531166324. ✔ 0.5

6406531166325. ✖ 0.40

Question Number : 198 Question Id : 640653351425 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Which of the following options represent the correct output of the following code?

```
import numpy as np
from sklearn.preprocessing import LabelBinarizer
y = np.array(['Rajni', 'Amitabh', 'Rajni', 'Mithun'])
y_dense = LabelBinarizer().fit_transform(y)
print(y_dense)
```

Options :

6406531166332. ✖

```
[[1]
 [0]
 [1]]
```

6406531166333. ✖

```
[[0 0 0 1]
 [1 0 0 0]
 [0 0 1 0]
 [0 1 0 0]]
```

6406531166334. ✖

```
[[1]
 [1]
 [1]
 [0]]
```

6406531166335. ✔

```
[[0 0 1]
 [1 0 0]
 [0 0 1]
 [0 1 0]]
```

Question Number : 199 Question Id : 640653351426 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Based on the given receiver operating characteristic curves in Figure 2, answer which of the following statements are TRUE ?

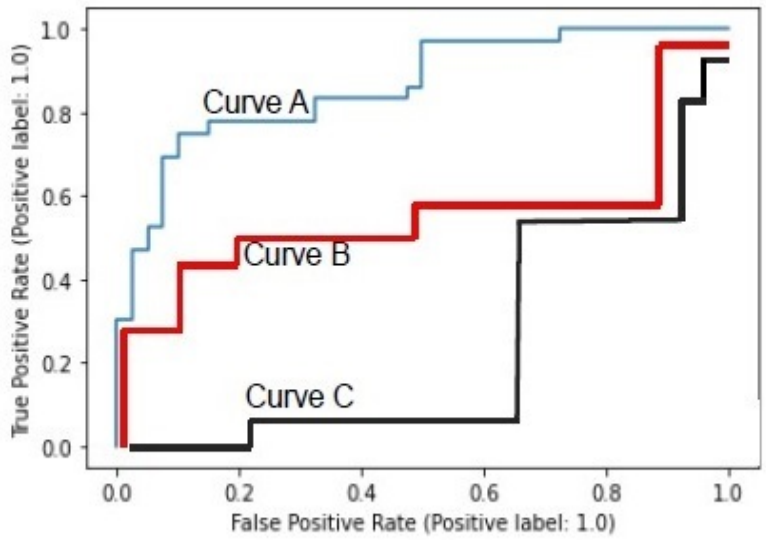

Figure 2: ROC curves

Options :

6406531166336. ✖ The classifier shown in 'Curve B' is better than the other two classifiers.

6406531166337. ✔ The classifier shown in 'Curve A' is better than the other two classifiers.

6406531166338. ✖ The classifier shown in 'Curve C' is better than the other two classifiers.

6406531166339. ✖ All 3 curves demonstrate 3 classifiers of the same quality.

Question Number : 200 Question Id : 640653351427 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

For a text classification task, which one of the following code snippets will be better suitable than the others?

Options :

6406531166340. ✖
```
from sklearn.naive_bayes import MultimodalNB
mnb = MultimodalNB()
mnb.fit(X_train, y_train)
```

6406531166341. ✔
```
from sklearn.naive_bayes import ComplementNB
cnb = ComplementNB()
cnb.fit(X_train, y_train)
```

6406531166342. ✖
```
from sklearn.naive_bayes import GaussianNB
gnb = GaussianNB()
gnb.fit(X_train, y_train)
```

6406531166343. ✖
```
from sklearn.naive_bayes import BernoulliNB
bnb = BernoulliNB()
bnb.fit(X_train, y_train)
```

Question Number : 201 Question Id : 640653351430 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Which of the following will be the correct output of the code snippet given below?

```
from sklearn.neighbors import KNeighborsClassifier
X = [[2,3], [5,6], [10,11], [15,16],[20,21]]
y = [0, 0, 1, 1, 2]
neigh = KNeighborsClassifier(n_neighbors=3)
neigh.fit(X, y)
print(neigh.predict([[8,9]]))
```

Options :

6406531166352. ✔ 0

6406531166353. ✖ 1

6406531166354. ✖ 2

6406531166355. ✖ None

Question Number : 202 Question Id : 640653351435 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

What is the output of the following block of code?

```
import numpy as np
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
X = np.array([[2, 4], [4, 2]])
y = np.array([-1, 1])
from sklearn.svm import SVC
clf = make_pipeline(StandardScaler(), SVC(kernel='linear'))
clf.fit(X, y)
print(clf.predict([[-2,2]]))
```

Options :

6406531166370. ✖ array[1]

6406531166371. ✖ array[3]

6406531166372. ✖ array[2]

6406531166373. ✔ array[-1]

Question Number : 203 Question Id : 640653351439 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Choice Question

Consider the following code snippet and mark the correct output.

```
from sklearn.svm import SVC
import numpy as np
X = np.array([[-1, -1], [-2, -2], [1, 1], [2, 2]])
y = np.array([-1, -1, 1, 1])
clf = SVC(kernel='linear')
clf.fit(X, y)
print(clf.n_support_)
```

Options :

6406531166379. ✖ [-1,1]

6406531166380. ✖ [-1,-1,1,1]

6406531166381. ✔ [1,1]

6406531166382. ✖ [2,2]

Sub-Section Number : 5

Sub-Section Id : 64065350430

Question Shuffling Allowed : Yes

Question Number : 204 Question Id : 640653351433 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Multiple Select Question

Which of the following code blocks will correctly train a large scale regression model using partial_fit, if shape of X_train,Y_train are given as (8000,900,10) and (8000,900) respectively. Note that the dataset has been transformed into batches of size 900 each?

Options :

6406531166361. ✖
```
from sklearn.linear_model import SGDRegressor
regressor = SGDRegressor(random_state=10)
for i in range(X_train.shape[[0]]):
    X_batch, Y_batch = X_train[i], Y_train[i]
    regressor.partial_fit(X_batch, Y_batch)
```

6406531166362. ✔
```
from sklearn.linear_model import SGDRegressor
regressor = SGDRegressor(random_state=10)
for i in range(Y_train.shape[0]):
    X_batch, Y_batch = X_train[i], Y_train[i]
    regressor.partial_fit(X_batch, Y_batch)
```

6406531166363. ✖
```
from sklearn.linear_model import SGDRegressor
regressor = SGDRegressor(random_state=10)
for i in range(Y_train.shape[[0]]):
    X_batch, Y_batch = X_train[i], Y_train[i]
    regressor.partial_fit(X_batch, Y_batch)
```

```
from sklearn.linear_model import SGDRegressor
regressor = SGDRegressor(random_state=10)
for i in range(X_train.shape[0]):
    X_batch, Y_batch = X_train[i], Y_train[i]
    regressor.partial_fit(X_batch, Y_batch)
```

6406531166364. ✔

Sub-Section Number : 6

Sub-Section Id : 64065350431

Question Shuffling Allowed : Yes

Question Number : 205 Question Id : 640653351431 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Short Answer Question

What will be the output of the following code snippet?
(Hint: Pairwise method Computes the pairwise distances.)

```
from sklearn.metrics import DistanceMetric
dist = DistanceMetric.get_metric('manhattan')
X = [[5, 6,],[3,2],
     [4, 5,]]
print(dist.pairwise(X).max())
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

6

Sub-Section Number : 7

Sub-Section Id : 64065350432

Question Shuffling Allowed : No

Question Id : 640653351421 Question Type : COMPREHENSION Sub Question Shuffling

Allowed : No Group Comprehension Questions : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Question Numbers : (206 to 207)

Question Label : Comprehension

Based on the confusion matrix of a specific dataset (MNIST) given in Figure 1, answer the given subquestions.

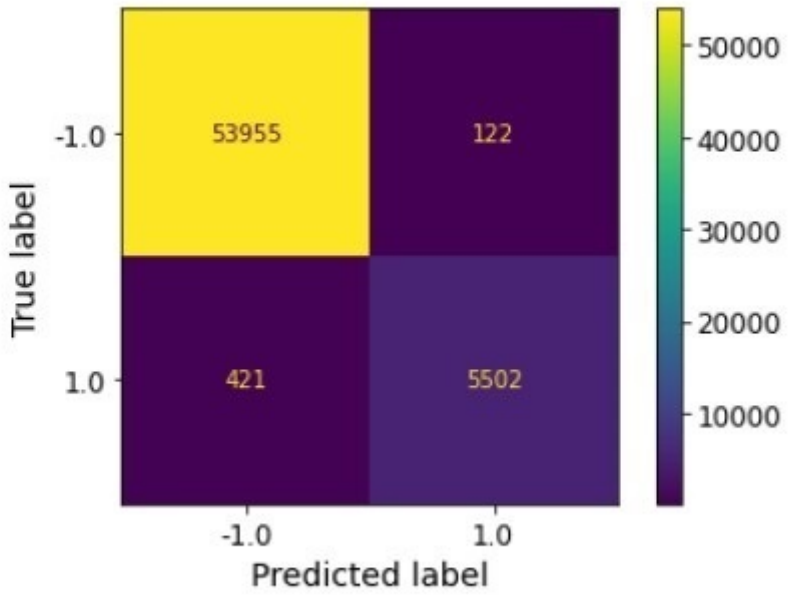

Figure 1: MNIST dataset confusion matrix

Sub questions

Question Number : 206 Question Id : 640653351422 Question Type : SA Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 1.5

Question Label : Short Answer Question

Calculate the precision value (up to 3 decimal places) from the confusion matrix.

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Range

Text Areas : PlainText

Possible Answers :

0.970 to 0.980

Question Number : 207 Question Id : 640653351423 Question Type : SA Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 1.5

Question Label : Short Answer Question

Calculate the recall value (up to 3 decimal places) from the confusion matrix.

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Range

Text Areas : PlainText

Possible Answers :

0.925 to 0.935

Sub-Section Number :	8
Sub-Section Id :	64065350433
Question Shuffling Allowed :	No

Question Id : 640653351436 Question Type : COMPREHENSION Sub Question Shuffling Allowed : No Group Comprehension Questions : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Question Numbers : (208 to 209)

Question Label : Comprehension

Prashant wrote the following code to train the data shown in the following Figure :

```
from sklearn.svm import SVC
import numpy as np
X_train = np.array([[5, 7],
                    [6, 5],
                    [7, 7],
                    [7, 6],
                    [7, 5],
                    [3, 7],
                    [4, 5],
                    [3, 1],
                    [2, 4],
                    [1, 3]])
y_train=np.array([0,0,0,0,0,1,1,1,1,1])

clf = SVC(kernel='linear')
clf.fit(X_train, y_train)
print(clf.support_vectors_)
```

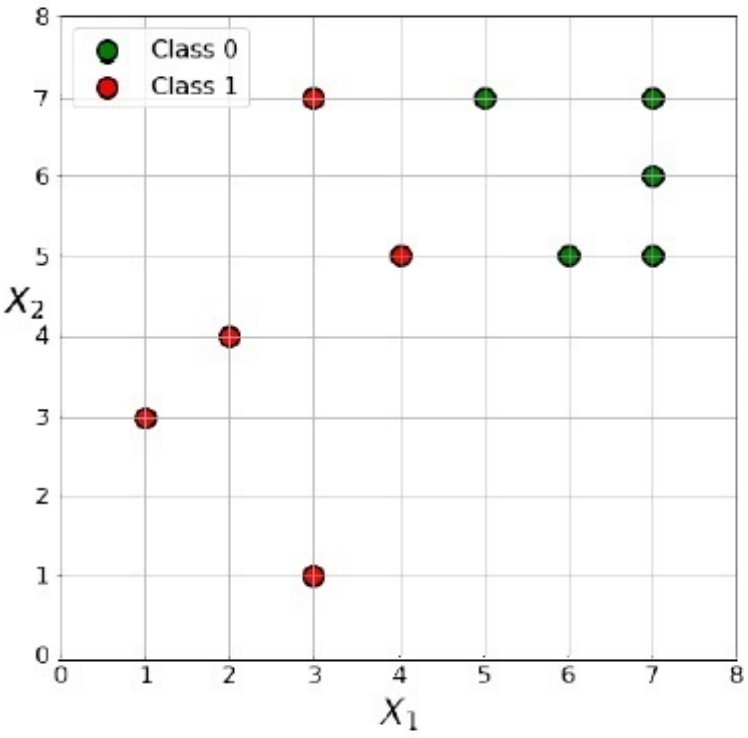

Figure: 4

Sub questions

Question Number : 208 Question Id : 640653351437 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 2

Question Label : Multiple Choice Question

What is the output that Prashant will get?

Options :

6406531166374. ✖ Indices of support vectors.

6406531166375. ✖ Number of support vectors.

6406531166376. ✖ Weights assigned to the features when kernel="linear"

6406531166377. ✔ ndarray of support vectors.

Question Number : 209 Question Id : 640653351438 Question Type : SA Calculator : None

Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 3

Question Label : Short Answer Question

Enter the output of the following block of code.

```
from sklearn.svm import SVC
SVC_classifier = SVC(kernel='linear')
clf = SVC_classifier.fit(X_train,y_train)
print(len(clf.support_vectors_))
```

Response Type : Numeric

Evaluation Required For SA : Yes

Show Word Count : Yes

Answers Type : Equal

Text Areas : PlainText

Possible Answers :

4

BDM

Section Id :	64065322142
Section Number :	12
Section type :	Online
Mandatory or Optional :	Mandatory