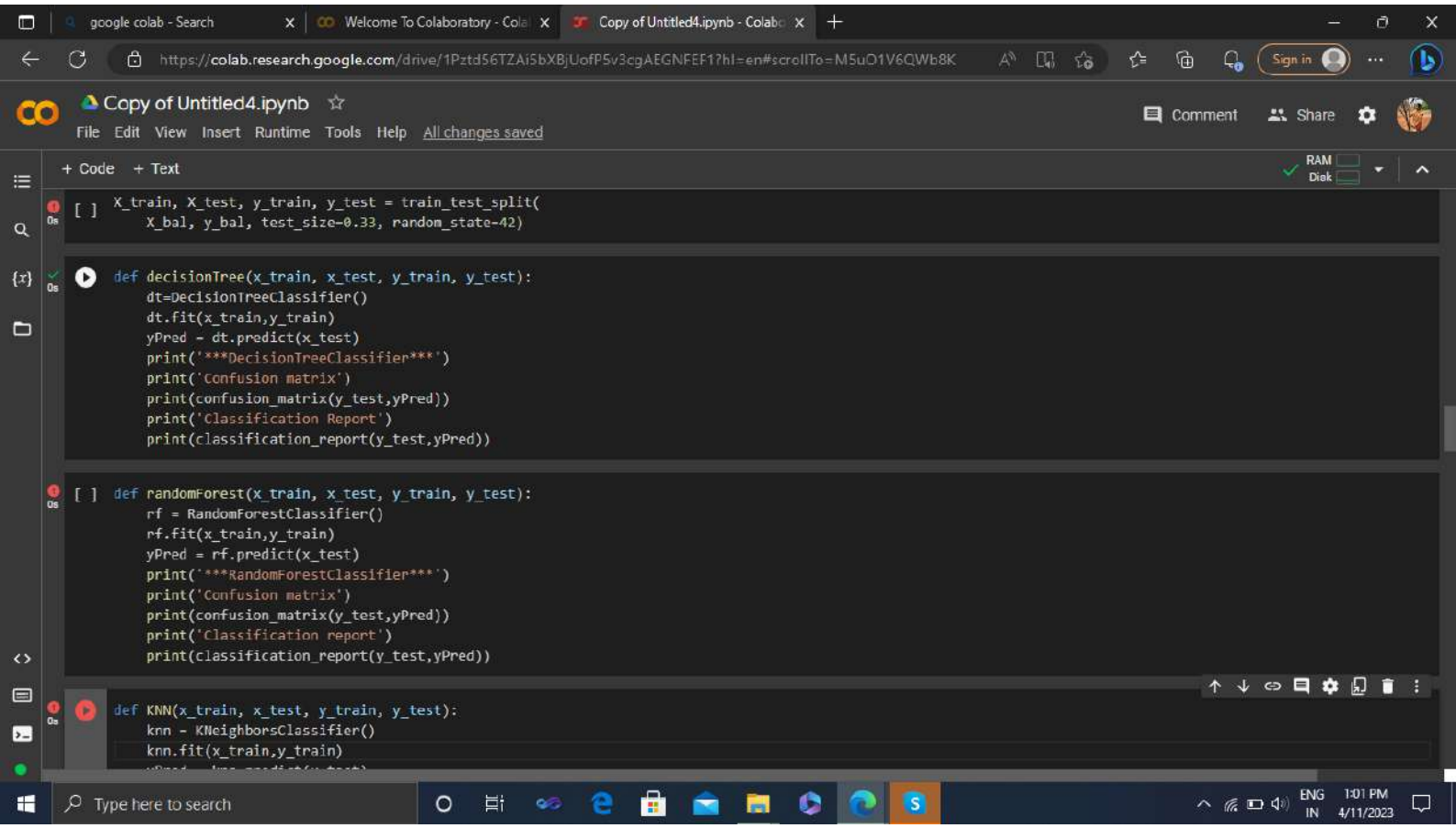
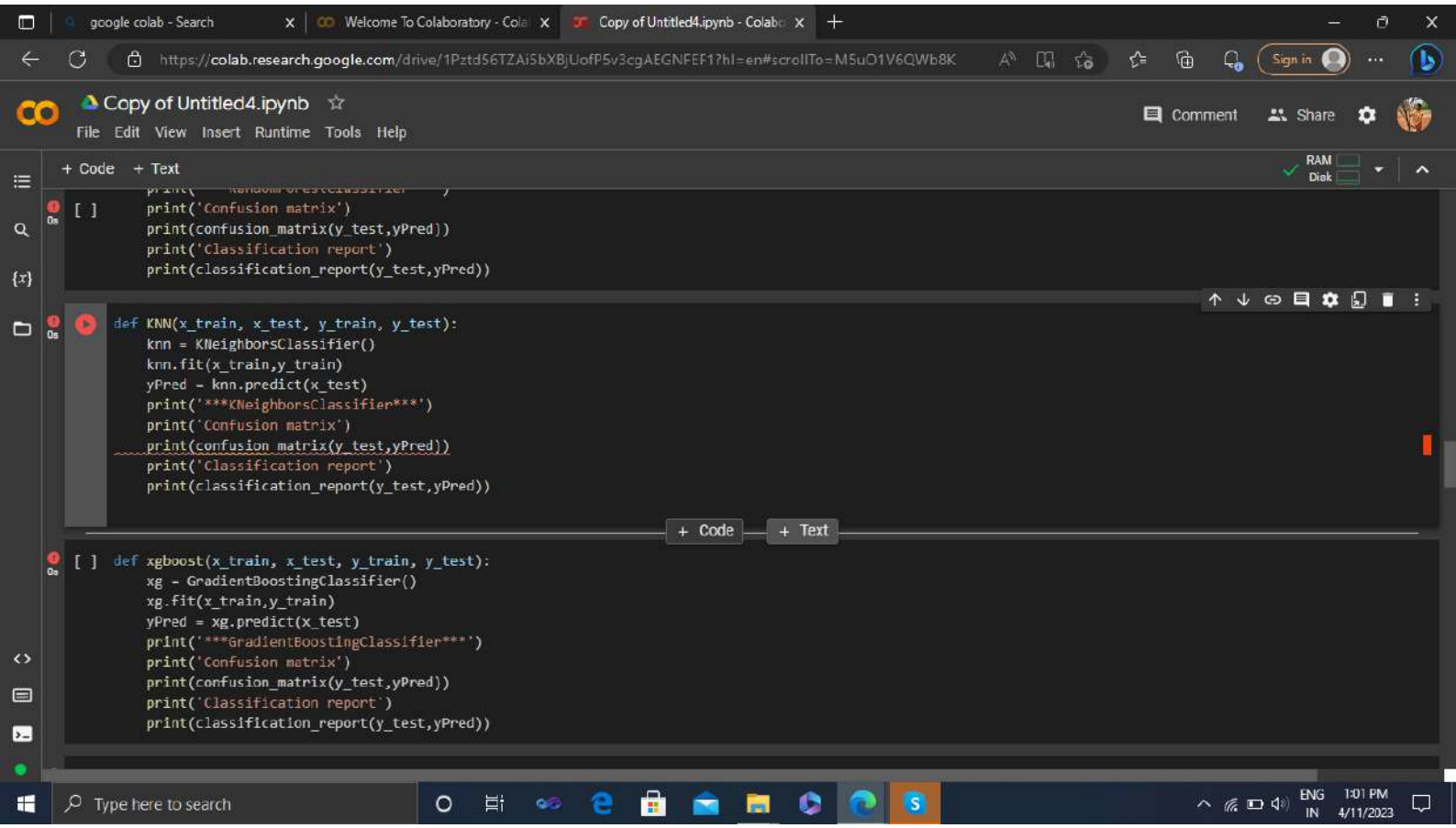
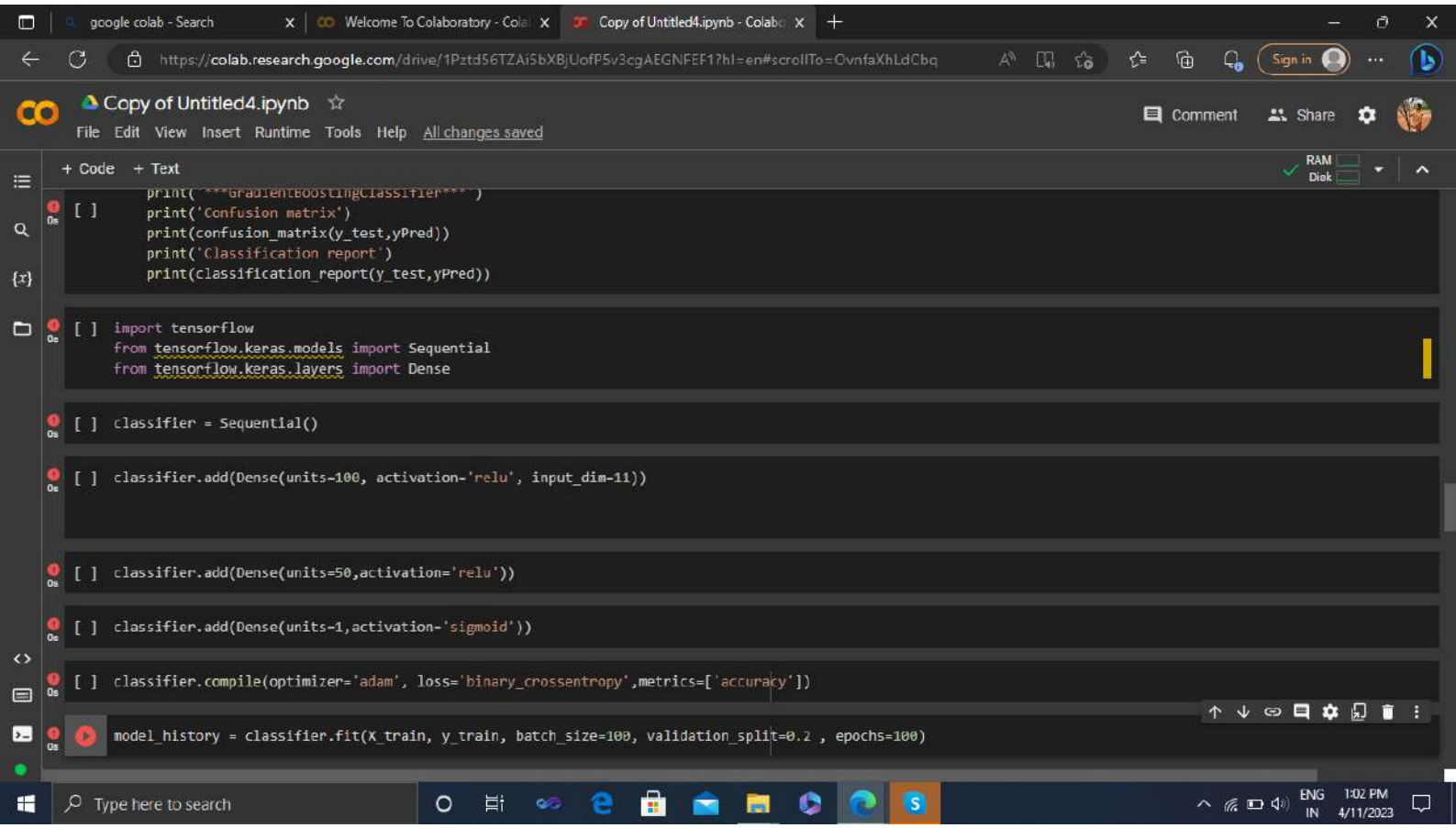
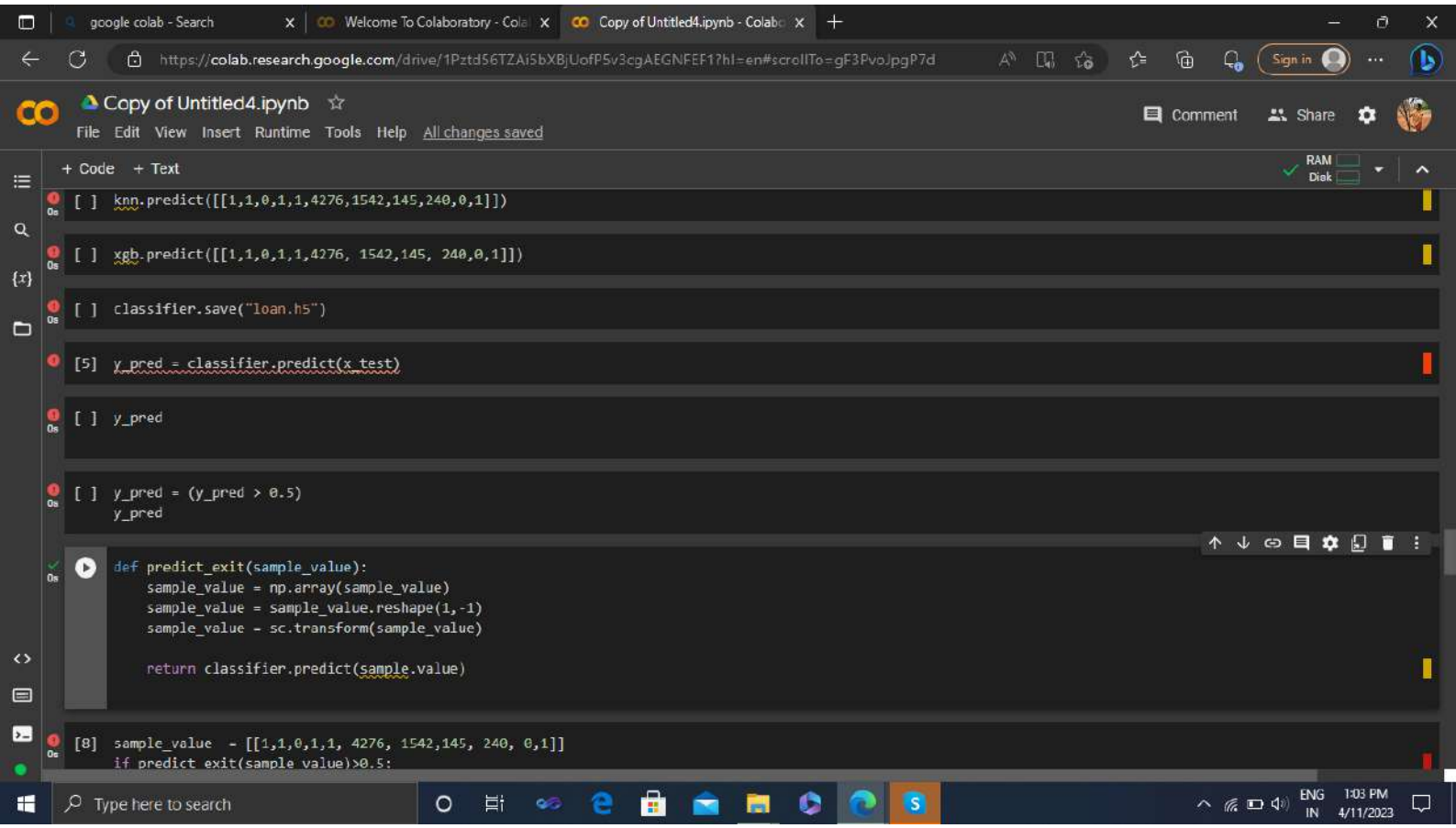
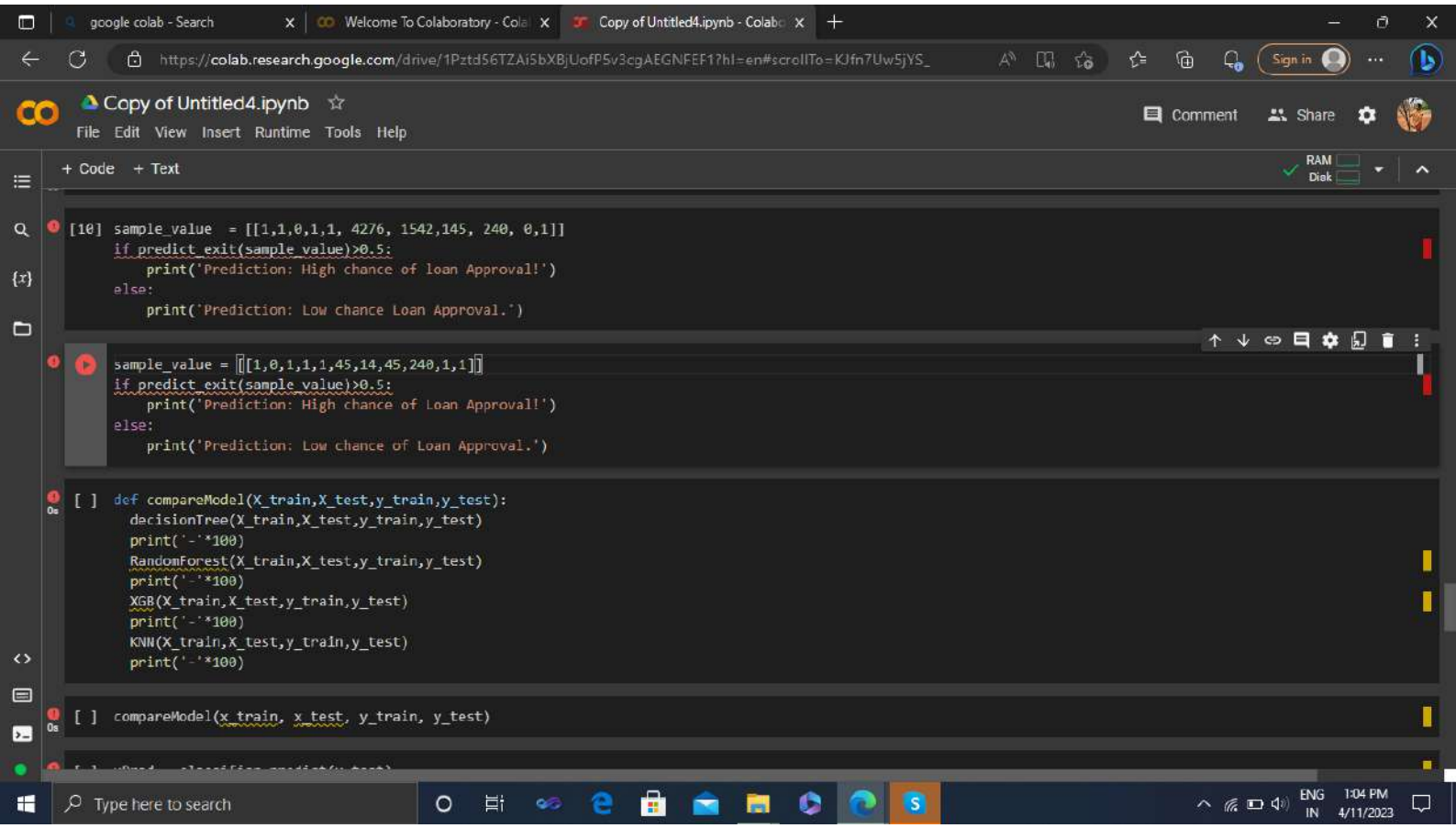

google colab - Search
Welcome To Colaboratory - Cola
Copy of Untitled4.ipynb - Colabo
https://colab.research.google.com/drive/1Pztd56TZAi5bXBjUofP5v3cgAEGNFEF1?hl=en#scrollTo=M5uO1V6QWb8K
Sign in
Copy of Untitled4.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
[ ] X_train, X_test, y_train, y_test = train_test_split(
X_bal, y_bal, test_size=0.33, random_state=42)
def decisionTree(x_train, x_test, y_train, y_test):
dt=DecisionTreeClassifier()
dt.fit(x_train,y_train)
yPred = dt.predict(x_test)
print('***DecisionTreeClassifier***')
print('Confusion matrix')
print(confusion_matrix(y_test,yPred))
print('Classification Report')
print(classification_report(y_test,yPred))
[ ] def randomForest(x_train, x_test, y_train, y_test):
rf = RandomForestClassifier()
rf.fit(x_train,y_train)
yPred = rf.predict(x_test)
print('***RandomForestClassifier***')
print('Confusion matrix')
print(confusion_matrix(y_test,yPred))
print('Classification report')
print(classification_report(y_test,yPred))
def KNN(x_train, x_test, y_train, y_test):
knn = KNeighborsClassifier()
knn.fit(x_train,y_train)
Type here to search
ENG
IN
1:01 PM
4/11/2023


google colab - Search
Welcome To Colaboratory - Cola
Copy of Untitled4.ipynb - Colabo
https://colab.research.google.com/drive/1Pztd56TZAi5bXBjUofP5v3cgAEGNFEF1?hl=en#scrollTo=M5uO1V6QWb8K
Sign in
Copy of Untitled4.ipynb
File Edit View Insert Runtime Tools Help
Comment
Share
RAM
Disk
+ Code + Text
[ ] print('Confusion matrix')
print(confusion_matrix(y_test,yPred))
print('Classification report')
print(classification_report(y_test,yPred))
def KNN(x_train, x_test, y_train, y_test):
knn = KNeighborsClassifier()
knn.fit(x_train,y_train)
yPred = knn.predict(x_test)
print('***KNeighborsClassifier***')
print('Confusion matrix')
print(confusion_matrix(y_test,yPred))
print('Classification report')
print(classification_report(y_test,yPred))
+ Code + Text
[ ] def xgboost(x_train, x_test, y_train, y_test):
xg = GradientBoostingClassifier()
xg.fit(x_train,y_train)
yPred = xg.predict(x_test)
print('***GradientBoostingClassifier***')
print('Confusion matrix')
print(confusion_matrix(y_test,yPred))
print('Classification report')
print(classification_report(y_test,yPred))
Type here to search
ENG
IN
1:01 PM
4/11/2023


google colab - Search
Welcome To Colaboratory - Cola
Copy of Untitled4.ipynb - Colabo
https://colab.research.google.com/drive/1Pztd56TZAi5bXBjUofP5v3cgAEGNFEF1?hl=en#scrollTo=OvnfaXhLdCbq
Sign in
Copy of Untitled4.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
print('***GradientBoostingClassifier***')
print('Confusion matrix')
print(confusion_matrix(y_test,yPred))
print('Classification report')
print(classification_report(y_test,yPred))
import tensorflow
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense
classifier = Sequential()
classifier.add(Dense(units=100, activation='relu', input_dim=11))
classifier.add(Dense(units=50,activation='relu'))
classifier.add(Dense(units=1,activation='sigmoid'))
classifier.compile(optimizer='adam', loss='binary_crossentropy',metrics=['accuracy'])
model_history = classifier.fit(X_train, y_train, batch_size=100, validation_split=0.2 , epochs=100)
Type here to search
ENG
IN
1:02 PM
4/11/2023


google colab - Search
Welcome To Colaboratory - Cola
Copy of Untitled4.ipynb - Colabo
https://colab.research.google.com/drive/1Pztd56TZAi5bXBjUofP5v3cgAEGNFEF1?hl=en#scrollTo=gF3PvoJpgP7d
Sign in
Copy of Untitled4.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
[ ] knn.predict([[1,1,0,1,1,4276,1542,145,240,0,1]])
[ ] xgb.predict([[1,1,0,1,1,4276, 1542,145, 240,0,1]])
[ ] classifier.save("loan.h5")
[5] y_pred = classifier.predict(x_test)
[ ] y_pred
[ ] y_pred = (y_pred > 0.5)
y_pred
def predict_exit(sample_value):
sample_value = np.array(sample_value)
sample_value = sample_value.reshape(1,-1)
sample_value = sc.transform(sample_value)
return classifier.predict(sample.value)
[8] sample_value = [[1,1,0,1,1, 4276, 1542,145, 240, 0,1]]
if predict_exit(sample_value)>0.5:
Type here to search
ENG
IN
1:03 PM
4/11/2023


google colab - Search
Welcome To Colaboratory - Cola
Copy of Untitled4.ipynb - Colabo
https://colab.research.google.com/drive/1Pztd56TZAi5bXBjUofP5v3cgAEGNFEF1?hl=en#scrollTo=KJfn7Uw5jYS_
Sign in
Copy of Untitled4.ipynb
File Edit View Insert Runtime Tools Help
Comment
Share
+ Code + Text
RAM
Disk
[10] sample_value = [[1,1,0,1,1, 4276, 1542,145, 240, 0,1]]
if predict_exit(sample_value)>0.5:
print('Prediction: High chance of loan Approval!')
else:
print('Prediction: Low chance Loan Approval.')
sample_value = [[1,0,1,1,1,45,14,45,240,1,1]]
if predict_exit(sample_value)>0.5:
print('Prediction: High chance of Loan Approval!')
else:
print('Prediction: Low chance of Loan Approval.')
[ ] def compareModel(X_train,X_test,y_train,y_test):
decisionTree(X_train,X_test,y_train,y_test)
print('-'*100)
RandomForest(X_train,X_test,y_train,y_test)
print('-'*100)
XGB(X_train,X_test,y_train,y_test)
print('-'*100)
KNN(X_train,X_test,y_train,y_test)
print('-'*100)
[ ] compareModel(x_train, x_test, y_train, y_test)
Type here to search
ENG
IN
1:04 PM
4/11/2023