EXPRERIMENT 5(b)

IMPLEMENTATION OF KNN ALGORITHM

NAME : SREENIDHI GANACHARI

REGISTRATION NUMBER : 19BCE7230

SLOT : L-23&24

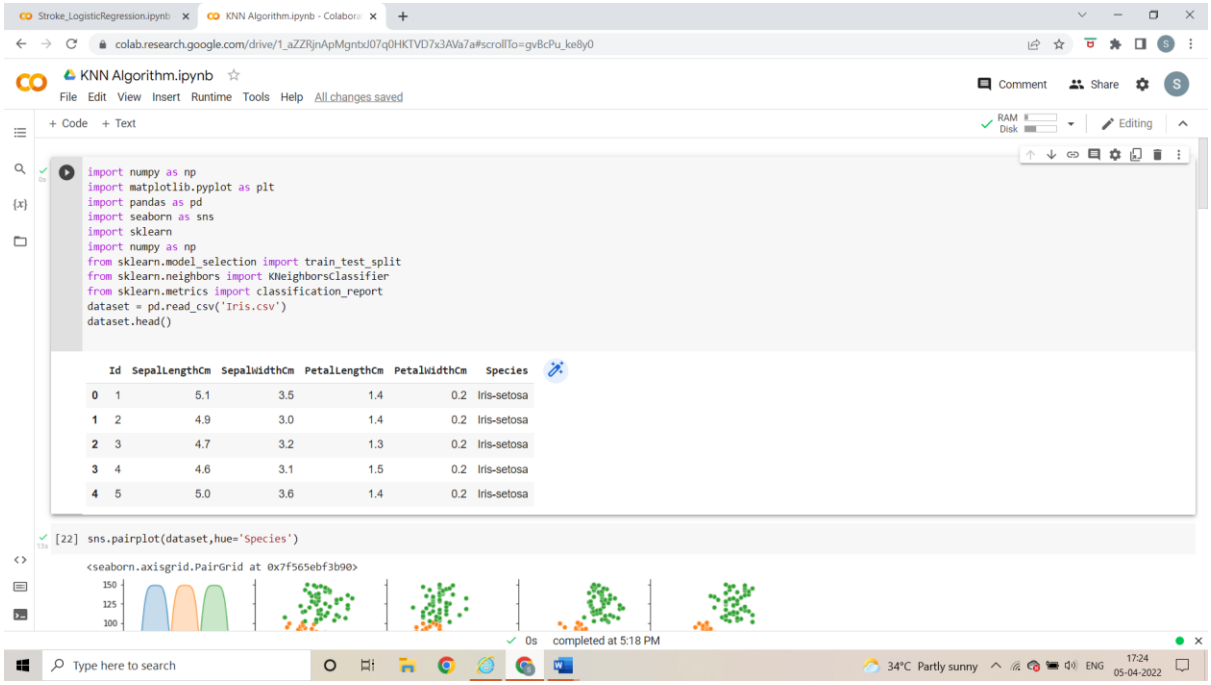
CODE –

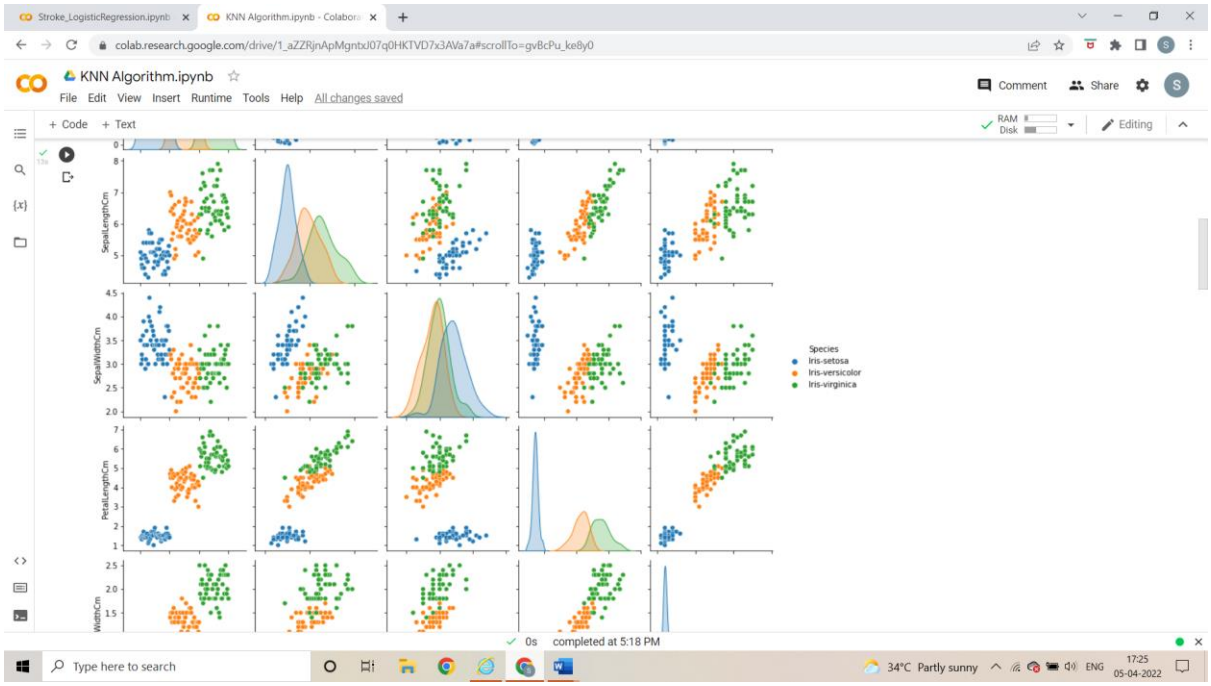
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import sklearn
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import classification_report
dataset = pd.read_csv('Iris.csv')
dataset.head()
sns.pairplot(dataset,hue='Species')
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
url = "https://archive.ics.uci.edu/ml/machine-learning-
databases/iris/iris.data"
names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-
width', 'Class']
dataset = pd.read_csv(url, names=names)
dataset.head()
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 4].values
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.5
, random_state=42)
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
scaler.fit(X_train)
StandardScaler(copy=True, with_mean=True, with_std=True)
X_train = scaler.transform(X_train)
_test = scaler.transform(X_test)
from sklearn.neighbors import KNeighborsClassifier
classifier1 = KNeighborsClassifier(n_neighbors=5)
```

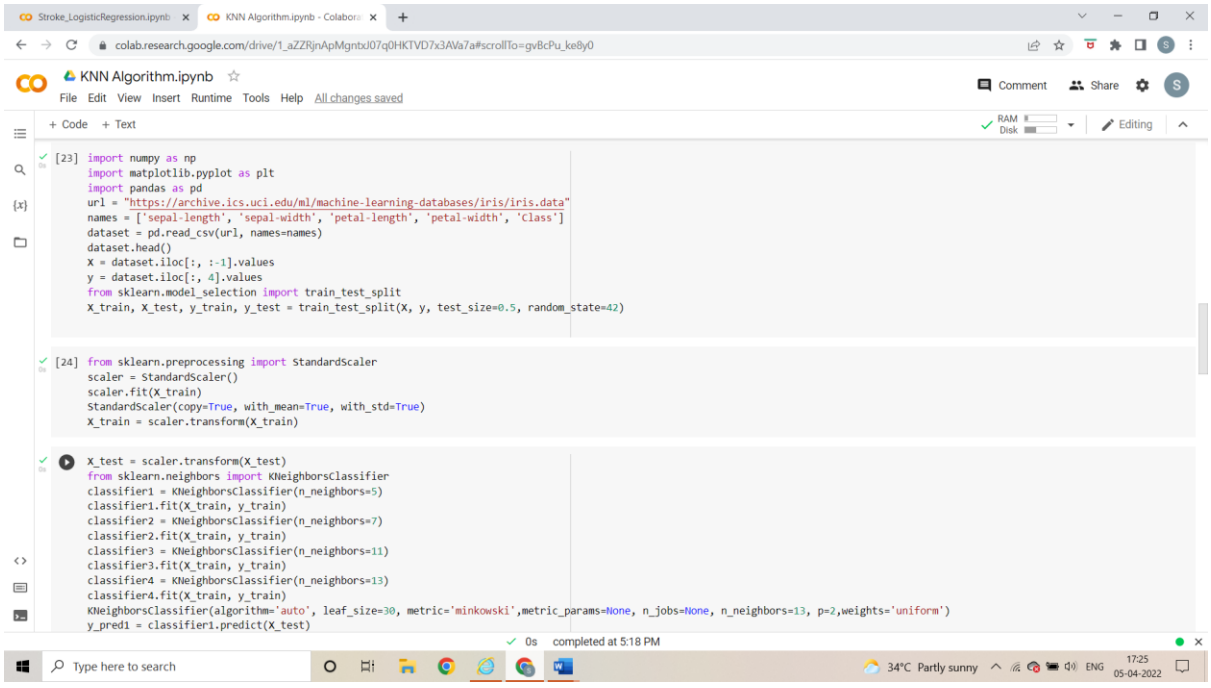
```
classifier1.fit(X_train, y_train)
classifier2 = KNeighborsClassifier(n_neighbors=7)
classifier2.fit(X_train, y_train)
classifier3 = KNeighborsClassifier(n_neighbors=11)
classifier3.fit(X_train, y_train)
classifier4 = KNeighborsClassifier(n_neighbors=13)
classifier4.fit(X_train, y_train)
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski'
,metric_params=None, n_jobs=None, n_neighbors=13, p=2,weights='uniform'
)
y_pred1 = classifier1.predict(X_test)
y_pred2 = classifier2.predict(X_test)
y_pred3 = classifier3.predict(X_test)
y_pred4 = classifier4.predict(X_test)
from sklearn.metrics import classification_report, confusion_matrix, ac
curacy_score
print("Confusion Matrix for 5 neighbors:\n")
print(confusion_matrix(y_test, y_pred1), "\n")
print("Classification Report for 5 neighbors\n")
print(classification_report(y_test, y_pred1))

print("Confusion Matrix for 7 neighbors:\n")
print(confusion_matrix(y_test, y_pred2), "\n")
print("Classification Report for 7 neighbors\n")
print(classification_report(y_test, y_pred2))

print("Confusion Matrix for 11 neighbors:\n")
print(confusion_matrix(y_test, y_pred3), "\n")
print("Classification Report for 11 neighbors\n")
print(classification_report(y_test, y_pred3))

print("Confusion Matrix for 13 neighbors:\n")
print(confusion_matrix(y_test, y_pred4), "\n")
print("Classification Report for 13 neighbors\n")
print(classification_report(y_test, y_pred4))
print("Correct predicition for 5 neighbors",accuracy_score(y_test,y_pre
d1))
print("Wrong predicition for 5 neighbors",(1-
accuracy_score(y_test,y_pred1)))

print("Correct predicition for 7 neighbors",accuracy_score(y_test,y_pre
d1))
print("Wrong predicition for 7 neighbors",(1-
accuracy_score(y_test,y_pred1)))

print("Correct predicition for 11 neighbors",accuracy_score(y_test,y_pr
ed1))
```

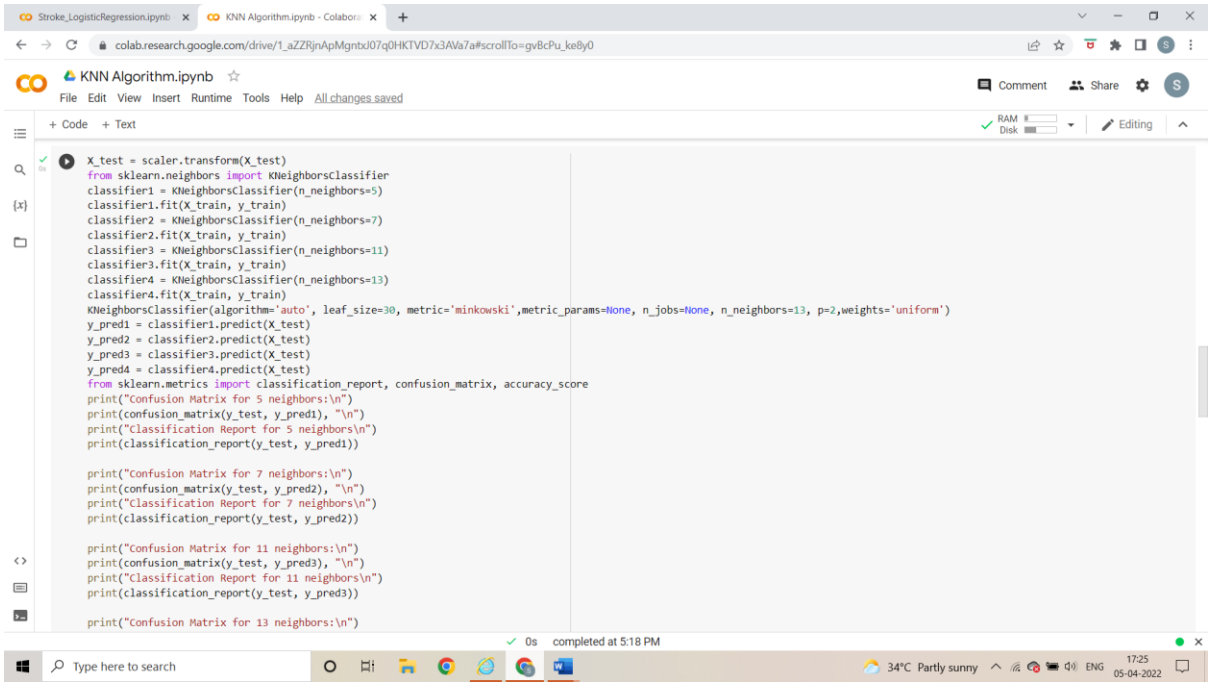
```
print("Wrong predicition for 11 neighbors",(1-
accuracy_score(y_test,y_pred1)))

print("Correct predicition for 13 neighbors",accuracy_score(y_test,y_pr
ed1))
print("Wrong predicition for 13 neighbors",(1-
accuracy_score(y_test,y_pred1)))
```

OUTPUT –

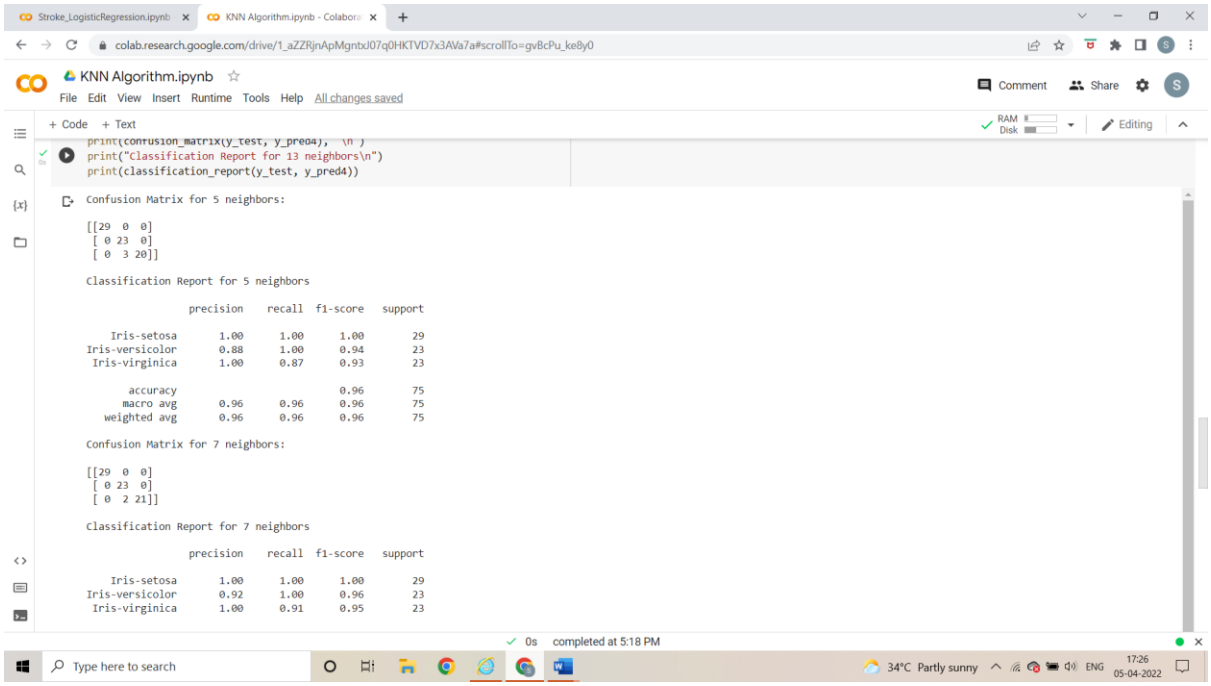



Stroke_LogisticRegression.ipynb
KNN Algorithm.ipynb - Colabora
colab.research.google.com/drive/1_aZZRjnApMgntxJ07q0HKTVD7x3AVa7a#scrollTo=gvBcPu_ke8y0
KNN Algorithm.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
SepalLengthCm
SepalWidthCm
PetalLengthCm
WidthCm
Species
Iris-setosa
Iris-versicolor
Iris-virginica
0s completed at 5:18 PM
Type here to search
34°C Partly sunny
ENG
17:25
05-04-2022


Stroke_LogisticRegression.ipynb
KNN Algorithm.ipynb - Colabora
colab.research.google.com/drive/1_aZZRjnApMgntxJ07q0HKTVD7x3AVa7a#scrollTo=gvBcPu_ke8y0
KNN Algorithm.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
[23] import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
url = "https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data"
names = ['sepal-length', 'sepal-width', 'petal-length', 'petal-width', 'Class']
dataset = pd.read_csv(url, names=names)
dataset.head()
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, 4].values
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.5, random_state=42)
[24] from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
scaler.fit(X_train)
StandardScaler(copy=True, with_mean=True, with_std=True)
X_train = scaler.transform(X_train)
X_test = scaler.transform(X_test)
from sklearn.neighbors import KNeighborsClassifier
classifier1 = KNeighborsClassifier(n_neighbors=5)
classifier1.fit(X_train, y_train)
classifier2 = KNeighborsClassifier(n_neighbors=7)
classifier2.fit(X_train, y_train)
classifier3 = KNeighborsClassifier(n_neighbors=11)
classifier3.fit(X_train, y_train)
classifier4 = KNeighborsClassifier(n_neighbors=13)
classifier4.fit(X_train, y_train)
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',metric_params=None, n_jobs=None, n_neighbors=13, p=2,weights='uniform')
y_pred1 = classifier1.predict(X_test)
0s completed at 5:18 PM
Type here to search
34°C Partly sunny
ENG
17:25
05-04-2022

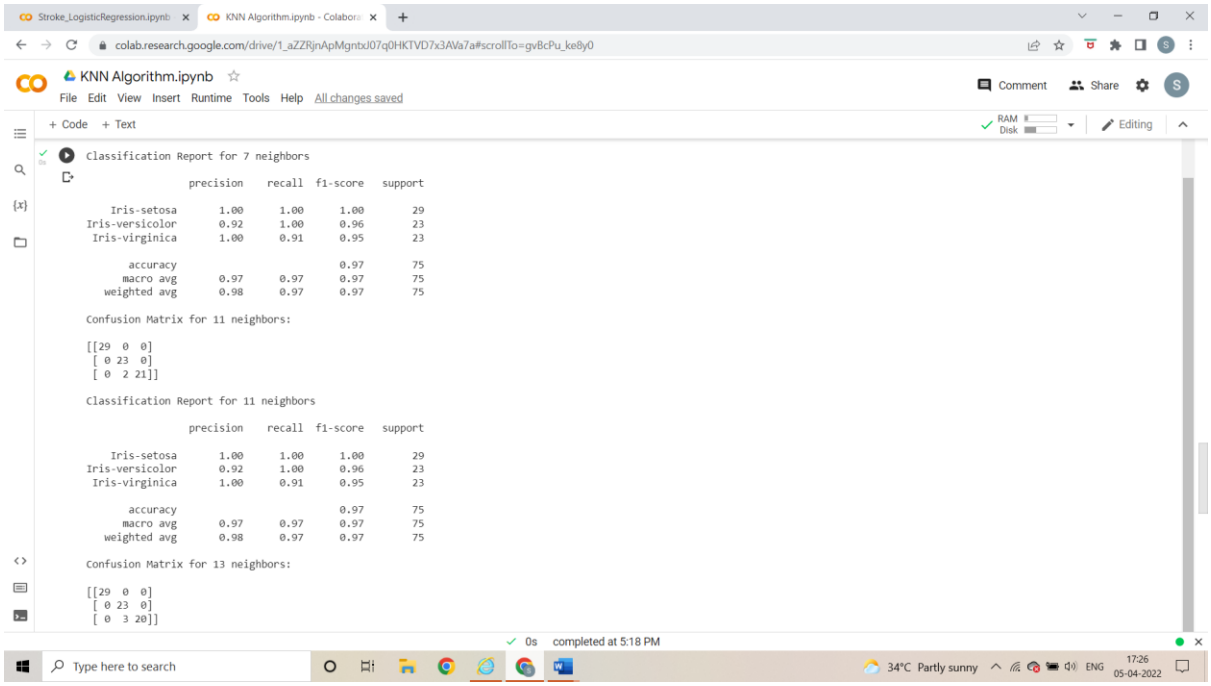


```
X_test = scaler.transform(X_test)
from sklearn.neighbors import KNeighborsClassifier
classifier1 = KNeighborsClassifier(n_neighbors=5)
classifier1.fit(X_train, y_train)
classifier2 = KNeighborsClassifier(n_neighbors=7)
classifier2.fit(X_train, y_train)
classifier3 = KNeighborsClassifier(n_neighbors=11)
classifier3.fit(X_train, y_train)
classifier4 = KNeighborsClassifier(n_neighbors=13)
classifier4.fit(X_train, y_train)
KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',metric_params=None, n_jobs=None, n_neighbors=13, p=2,weights='uniform')
y_pred1 = classifier1.predict(X_test)
y_pred2 = classifier2.predict(X_test)
y_pred3 = classifier3.predict(X_test)
y_pred4 = classifier4.predict(X_test)
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score
print("Confusion Matrix for 5 neighbors:\n")
print(confusion_matrix(y_test, y_pred1), "\n")
print("Classification Report for 5 neighbors\n")
print(classification_report(y_test, y_pred1))

print("Confusion Matrix for 7 neighbors:\n")
print(confusion_matrix(y_test, y_pred2), "\n")
print("Classification Report for 7 neighbors\n")
print(classification_report(y_test, y_pred2))

print("Confusion Matrix for 11 neighbors:\n")
print(confusion_matrix(y_test, y_pred3), "\n")
print("Classification Report for 11 neighbors\n")
print(classification_report(y_test, y_pred3))

print("Confusion Matrix for 13 neighbors:\n")
```



```
print(confusion_matrix(y_test, y_pred4), "\n")
print("Classification Report for 13 neighbors\n")
print(classification_report(y_test, y_pred4))
```

```
Confusion Matrix for 5 neighbors:

[[29  0  0]
 [ 0 23  0]
 [ 0  3 20]]

Classification Report for 5 neighbors

                 precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        29
Iris-versicolor       0.88      1.00      0.94        23
 Iris-virginica       1.00      0.87      0.93        23

       accuracy                           0.96        75
      macro avg       0.96      0.96      0.96        75
   weighted avg       0.96      0.96      0.96        75

Confusion Matrix for 7 neighbors:

[[29  0  0]
 [ 0 23  0]
 [ 0  2 21]]

Classification Report for 7 neighbors

                 precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        29
Iris-versicolor       0.92      1.00      0.96        23
 Iris-virginica       1.00      0.91      0.95        23
```



Stroke_LogisticRegression.ipynb | KNN Algorithm.ipynb - Colabora

colab.research.google.com/drive/1_aZZRjnApMgntxJ07q0HKTVD7x3AVa7a#scrollTo=gvBcPu_ke8y0

KNN Algorithm.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

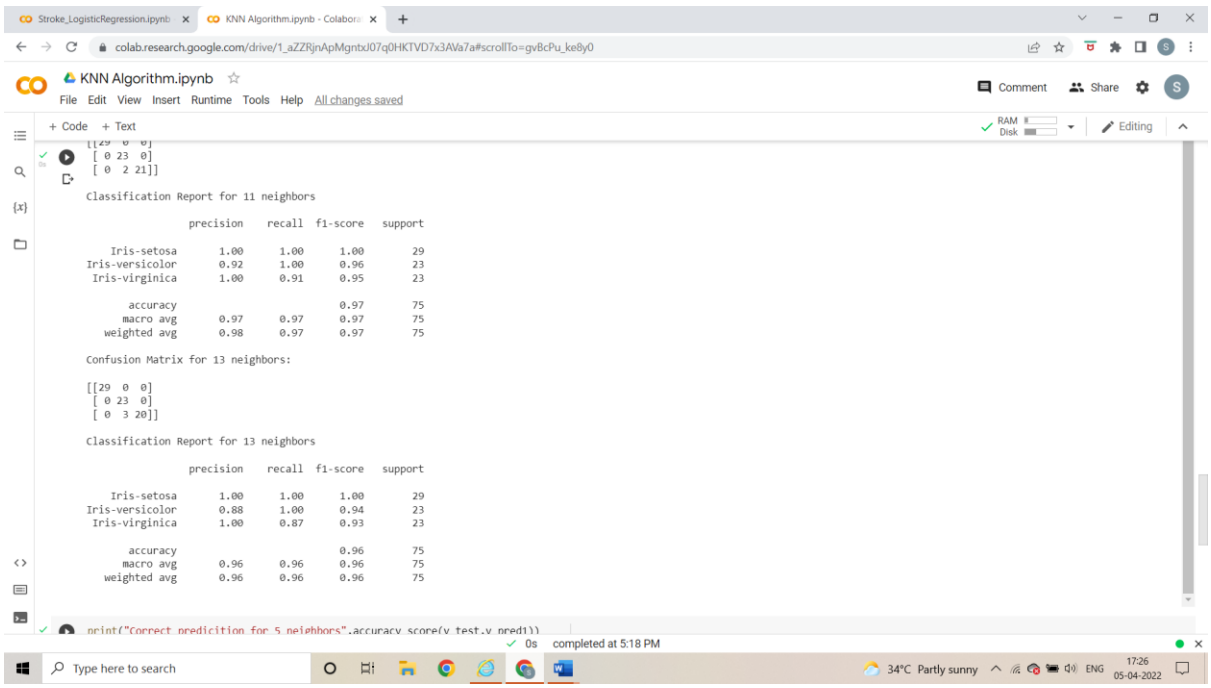
+ Code + Text

RAM Disk Editing

```
Classification Report for 7 neighbors

                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        29
Iris-versicolor       0.92      1.00      0.96        23
 Iris-virginica       1.00      0.91      0.95        23

       accuracy                           0.97        75
      macro avg       0.97      0.97      0.97        75
   weighted avg       0.98      0.97      0.97        75

Confusion Matrix for 11 neighbors:

[[29  0  0]
 [ 0 23  0]
 [ 0  2 21]]

Classification Report for 11 neighbors

                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        29
Iris-versicolor       0.92      1.00      0.96        23
 Iris-virginica       1.00      0.91      0.95        23

       accuracy                           0.97        75
      macro avg       0.97      0.97      0.97        75
   weighted avg       0.98      0.97      0.97        75

Confusion Matrix for 13 neighbors:

[[29  0  0]
 [ 0 23  0]
 [ 0  3 20]]
```

0s completed at 5:18 PM

Type here to search 34°C Partly sunny ENG 17:26 05-04-2022



Stroke_LogisticRegression.ipynb | KNN Algorithm.ipynb - Colabora

colab.research.google.com/drive/1_aZZRjnApMgntxJ07q0HKTVD7x3AVa7a#scrollTo=gvBcPu_ke8y0

KNN Algorithm.ipynb

File Edit View Insert Runtime Tools Help All changes saved

Comment Share

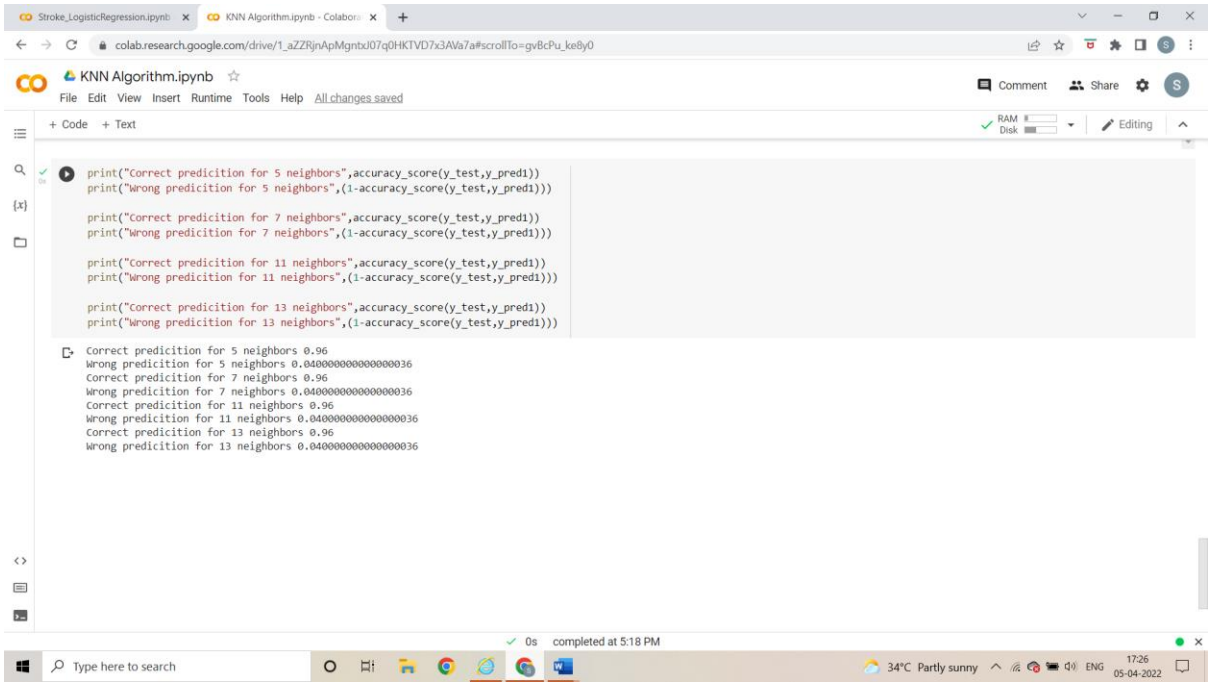
+ Code + Text

RAM Disk Editing

```
[[29  0  0]
 [ 0 23  0]
 [ 0  2 21]]

Classification Report for 11 neighbors

                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        29
Iris-versicolor       0.92      1.00      0.96        23
 Iris-virginica       1.00      0.91      0.95        23

       accuracy                           0.97        75
      macro avg       0.97      0.97      0.97        75
   weighted avg       0.98      0.97      0.97        75

Confusion Matrix for 13 neighbors:

[[29  0  0]
 [ 0 23  0]
 [ 0  3 20]]

Classification Report for 13 neighbors

                precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        29
Iris-versicolor       0.88      1.00      0.94        23
 Iris-virginica       1.00      0.87      0.93        23

       accuracy                           0.96        75
      macro avg       0.96      0.96      0.96        75
   weighted avg       0.96      0.96      0.96        75

print("Correct predicition for 5 neighbors",accuracy_score(y_test,y_pred1))
```

0s completed at 5:18 PM

Type here to search 34°C Partly sunny ENG 17:26 05-04-2022


Stroke_LogisticRegression.ipynb
KNN Algorithm.ipynb - Colabora
colab.research.google.com/drive/1_aZZRjnApMgntxJ07q0HKTVD7x3AVa7a#scrollTo=gvBcPu_ke8y0
KNN Algorithm.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment
Share
+ Code + Text
RAM
Disk
Editing
print("Correct predicition for 5 neighbors",accuracy_score(y_test,y_pred1))
print("Wrong predicition for 5 neighbors",(1-accuracy_score(y_test,y_pred1)))
print("Correct predicition for 7 neighbors",accuracy_score(y_test,y_pred1))
print("Wrong predicition for 7 neighbors",(1-accuracy_score(y_test,y_pred1)))
print("Correct predicition for 11 neighbors",accuracy_score(y_test,y_pred1))
print("Wrong predicition for 11 neighbors",(1-accuracy_score(y_test,y_pred1)))
print("Correct predicition for 13 neighbors",accuracy_score(y_test,y_pred1))
print("Wrong predicition for 13 neighbors",(1-accuracy_score(y_test,y_pred1)))
Correct predicition for 5 neighbors 0.96
Wrong predicition for 5 neighbors 0.040000000000000036
Correct predicition for 7 neighbors 0.96
Wrong predicition for 7 neighbors 0.040000000000000036
Correct predicition for 11 neighbors 0.96
Wrong predicition for 11 neighbors 0.040000000000000036
Correct predicition for 13 neighbors 0.96
Wrong predicition for 13 neighbors 0.040000000000000036
0s completed at 5:18 PM
Type here to search
34°C Partly sunny
ENG
17:26
05-04-2022