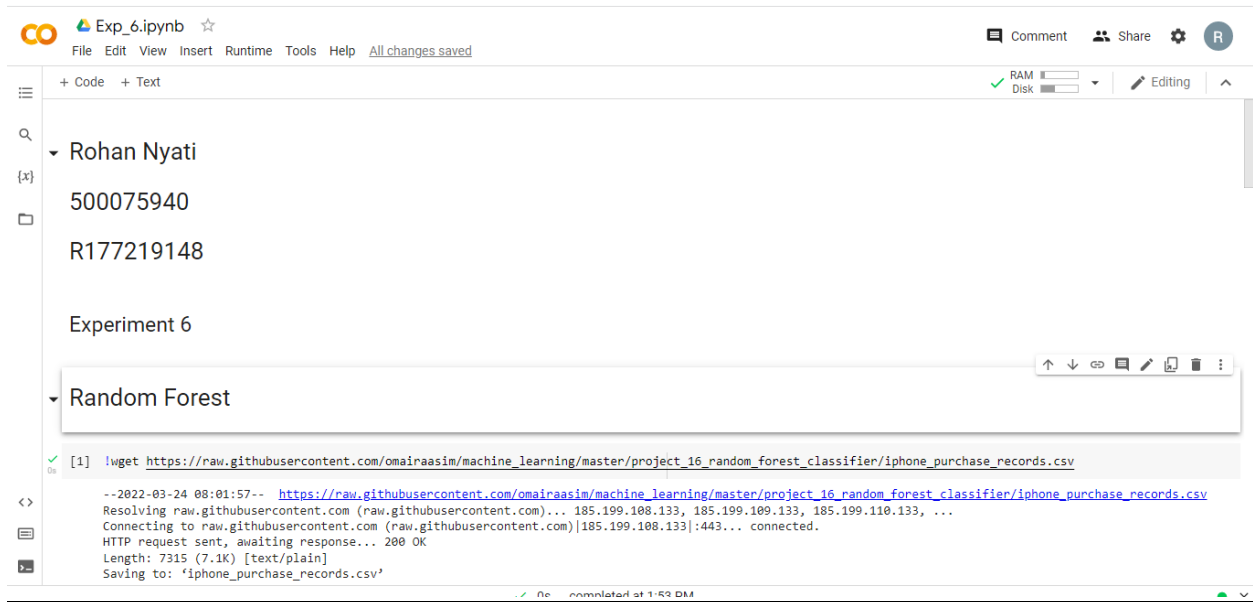


Rohan Nyati
500075940
R177219148
Batch 5

Experiment – 6



Exp_6.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM Disk

Editing

Rohan Nyati

500075940

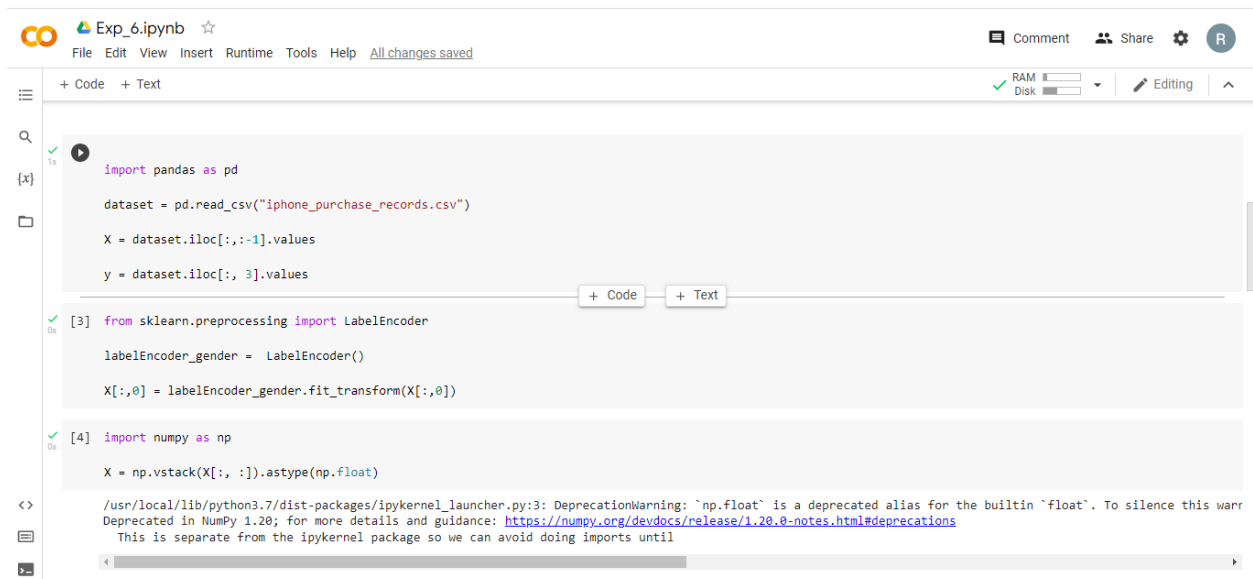
R177219148

Experiment 6

Random Forest

```
[1] !wget https://raw.githubusercontent.com/omairaasim/machine_learning/master/project_16_random_forest_classifier/iphone_purchase_records.csv
```

```
--2022-03-24 08:01:57-- https://raw.githubusercontent.com/omairaasim/machine_learning/master/project_16_random_forest_classifier/iphone_purchase_records.csv
Resolving raw.githubusercontent.com (raw.githubusercontent.com)... 185.199.108.133, 185.199.109.133, 185.199.110.133, ...
Connecting to raw.githubusercontent.com (raw.githubusercontent.com)|185.199.108.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 7315 (7.1K) [text/plain]
Saving to: 'iphone_purchase_records.csv'
```



Exp_6.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM Disk

Editing

```
import pandas as pd

dataset = pd.read_csv("iphone_purchase_records.csv")

X = dataset.iloc[:, :-1].values

y = dataset.iloc[:, 3].values
```

```
[3] from sklearn.preprocessing import LabelEncoder

labelEncoder_gender = LabelEncoder()

X[:, 0] = labelEncoder_gender.fit_transform(X[:, 0])
```

```
[4] import numpy as np

X = np.vstack(X[:, :]).astype(np.float)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3: DeprecationWarning: `np.float` is a deprecated alias for the builtin `float`. To silence this warning
Deprecated in NumPy 1.20; for more details and guidance: https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations
This is separate from the ipykernel package so we can avoid doing imports until
```

Exp_6.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM Disk

Editing

[5] from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.25, random_state=0)

from sklearn.ensemble import RandomForestClassifier

Classifier = RandomForestClassifier(n_estimators=100, criterion="entropy", random_state=0)

Classifier.fit(X_train, y_train)

RandomForestClassifier(criterion='entropy', random_state=0)

[7] y_pred = Classifier.predict(X_test)

[8] from sklearn import metrics

cm = metrics.confusion_matrix(y_test, y_pred)

print(cm)

accuracy = metrics.accuracy_score(y_test, y_pred)

print("Accuracy score:", accuracy)

Exp_6.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM Disk

Editing

print(cm)

accuracy = metrics.accuracy_score(y_test, y_pred)

print("Accuracy score:", accuracy)

precision = metrics.precision_score(y_test, y_pred)

print("Precision score:", precision)

recall = metrics.recall_score(y_test, y_pred)

print("Recall score:", recall)

[[64 4]
[4 28]]
Accuracy score: 0.92
Precision score: 0.875
Recall score: 0.875

Decision Tree

[18] from sklearn.tree import DecisionTreeClassifier

Classifier = DecisionTreeClassifier(criterion = "entropy", random_state=0)

Classifier.fit(X_train, y_train)

Exp_6.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

RAM 100% Disk 100%

Editing

[18]

classifier = DecisionTreeClassifier(criterion = "entropy", random_state=0)
classifier.fit(X_train, y_train)

DecisionTreeClassifier(criterion='entropy', random_state=0)

[19]

clf.fit(X_train,y_train)

DecisionTreeClassifier(random_state=0)

from sklearn import metrics
cm = metrics.confusion_matrix(y_test, y_pred)
print(cm)
accuracy = metrics.accuracy_score(y_test, y_pred)
print("Accuracy score:",accuracy)
precision = metrics.precision_score(y_test, y_pred)
print("Precision score:",precision)
recall = metrics.recall_score(y_test, y_pred)
print("Recall score:",recall)

[[64 4]
[4 28]]
Accuracy score: 0.92
Precision score: 0.875
Recall score: 0.875

print(metrics.classification_report(y_test, y_pred))

	precision	recall	f1-score	support
0	0.94	0.94	0.94	68
1	0.88	0.88	0.88	32
accuracy			0.92	100
macro avg	0.91	0.91	0.91	100
weighted avg	0.92	0.92	0.92	100