CS156 (Introduction to AI), Spring 2022

Homework 6 submission

Roster Name: Bernard Tan

Preferred Name (if different): Bernard

Student ID: 015215317

Email address: bernard.tan@sjsu.edu

Any special notes or anything you would like to communicate to me about this homework submission goes in here.

References and sources

List all your references and sources here. This includes all sites/discussion boards/blogs/posts/etc. where you grabbed some code examples.

Solution

Load libraries and set random number generator seed

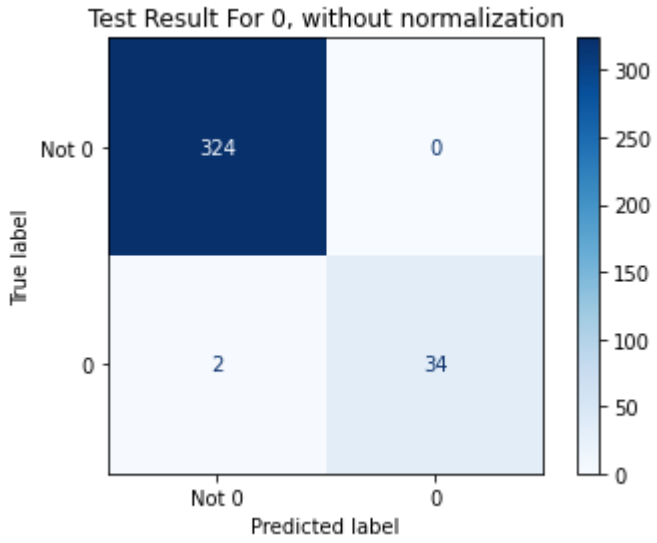
```
# 1. From Canvas Project Example (Jupyter Notebook, Perceptron.Breast.ipynb)
# 2. https://www.codegrepper.com/code-examples/python/flatten+images+for+mnist+model (Flatten
# 3. https://pandas.pydata.org/docs/reference/api/pandas.get_dummies.html (Get Dummies One-Ho

import numpy as np
import pandas as pd
from sklearn import datasets
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import Perceptron
from sklearn.metrics import plot_confusion_matrix
```

```
np.random.seed(42)
```

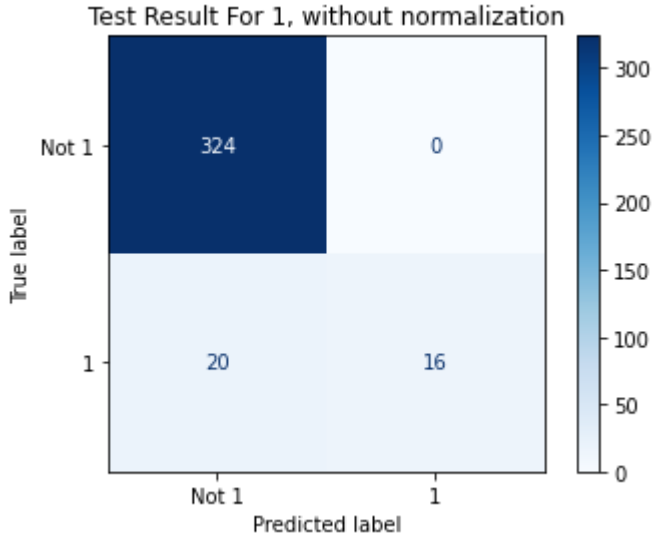
Code the solution

```
# Data Loading
data = datasets.load_digits()

# Data Normalized and Flattened
X = data ['images']
X = X.reshape(X.shape[0], -1)
X = X.astype ("float32") / 255
Y = pd.DataFrame(data.target)


# One-Hot encode
Y_ohe = pd.get_dummies(Y, columns = Y.columns, prefix = Y.columns)

X_train, X_test, Y_train, Y_test = train_test_split(X, Y_ohe, test_size=0.2, random_state=0,


# Perceptron Model
for i in range(len(Y_train.columns)):
    model = Perceptron(tol=1e-3, random_state=0)
    model.fit(X_train, Y_train ['0_' + str(i)])

    print('Accuracy of perceptron on training set ' , i, ' : {:.2f}'.format(model.score(X_tra
    print('Accuracy of perceptron on  testing set ' , i, ' : {:.2f}'.format(model.score(X_tes

    print ('\n')

    # Confusion Matrix
    titles_options = [("Test Result For " + str(i) + ", without normalization", None)]

    for title, normalize in titles_options:
        disp = plot_confusion_matrix(model, X_test, Y_test ['0_' + str(i)], display_labels=["

        disp.ax_.set_title(title)
        disp.ax_.set_title(title)

    plt.show()

    print ('\n \n')
```

```
Accuracy of perceptron on training set  0  : 0.99
Accuracy of perceptron on  testing set  0  : 0.99


/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: FutureWarning
  warnings.warn(msg, category=FutureWarning)
```




```
Accuracy of perceptron on training set  1  : 0.95
Accuracy of perceptron on  testing set  1  : 0.94


/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: FutureWarning
  warnings.warn(msg, category=FutureWarning)
```




```
Accuracy of perceptron on training set  2  : 0.99
Accuracy of perceptron on  testing set  2  : 0.99


/usr/local/lib/python3.7/dist-packages/sklearn/utils/deprecation.py:87: FutureWarning
  warnings.warn(msg, category=FutureWarning)
```