

## **ASSIGNMENT NAME**

# **MNIST Digit Classification Machine Learning Project**

**NAME:** DEEPAK KIRTANIA

**DOMAIN:** MACHINE LEARNING

**PROJECT:** MINOR

# Code with Comments:

**# fetching all the datasets...**

```
from sklearn.datasets import fetch_openml
```

```
import matplotlib
```

```
import matplotlib.pyplot as plt
```

```
import numpy as np
```

```
from sklearn.linear_model import LogisticRegression
```

```
from sklearn.model_selection import cross_val_score
```

```
mnist = fetch_openml('mnist_784')
```

```
x, y = mnist['data'], mnist['target']  
dg = x.to_numpy()[25000]
```

### **# reshaping to 28 by 28 pixels...**

```
dg_image = dg.reshape(28, 28)  
plt.imshow(dg_image,  
cmap=matplotlib.cm.binary,interpolation='nearest')  
plt.axis("off")  
plt.show()
```

### **# Slicing the numpy array for training and testing...**

```
x_train, x_test = x[0:60000], x[6000:70000]  
y_train, y_test = y[0:60000], y[6000:70000]
```

### **# shuffling the data for better results...**

```
shuffle_index = np.random.permutation(60000)  
x_train, y_train = x_train[shuffle_index], y_train[shuffle_index]
```

### **# Creating a 3-detector**

```
y_train = y_train.astype(np.int8)  
y_test = y_test.astype(np.int8)  
y_train_3 = (y_train == '3')  
y_test_3 = (y_test == '3')
```

**# Training a logistic regression classifier**

```
clf = LogisticRegression(tol=0.1)
```

**# using fit 'module' from classifier and 'predict' attribute to**

**# predict the data is correct or not (previously which we**

**# checked on dg)...**

```
clf.fit(x_train, y_train_3)
```

```
res= clf.predict([dg])
```

```
print(res)
```

**# Cross Validation for better accuracy**

```
mn = cross_val_score(clf, x_train, y_train_3, cv=3,  
scoring="accuracy")
```

```
print(mn.mean())
```

**# End of code.**