

In [26]:

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
%matplotlib inline
```

In [19]:

```
data = pd.read_csv('mnist.csv')
```

In [12]:

```
data.head()
```

Out[12]:

	class	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	...	pixel775	pixel776	pixel777	pixel778	pixel7
0	5	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
2	4	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
3	1	0	0	0	0	0	0	0	0	0	...	0	0	0	0	
4	9	0	0	0	0	0	0	0	0	0	...	0	0	0	0	

5 rows × 785 columns

In [8]:

```
a = data.iloc[4,1:].values
```

In [10]:

```
a= a.reshape(28,28).astype('uint8')
plt.imshow(a)
```

Out[10]:

<matplotlib.image.AxesImage at 0x27d7cb2a588>

In [13]:

```
df_x = data.iloc[:,1:]
df_y = data.iloc[:,0]
```

In [27]:

```
x_train, x_test, y_train, y_test = train_test_split(df_x, df_y, test_size = 0.2, random_state=9)
```

In [28]:

```
x_train.head()
```

Out[28]:

	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	pixel10	...	pixel775	pixel776	pixel777	pixel778
303	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0
2156	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0
43791	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0
57402	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0
49064	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0

5 rows × 784 columns

In [29]:

```
y_train.head()
```

Out[29]:

```
303      0
2156     6
43791    1
57402    3
49064    8
Name: class, dtype: int64
```

In [31]:

```
rf = RandomForestClassifier(n_estimators=100)
```

In [32]:

```
rf.fit(x_train , y_train)
```

Out[32]:

```
RandomForestClassifier(bootstrap=True, ccp_alpha=0.0, class_weight=None,
                        criterion='gini', max_depth=None, max_features='auto',
                        max_leaf_nodes=None, max_samples=None,
                        min_impurity_decrease=0.0, min_impurity_split=None,
                        min_samples_leaf=1, min_samples_split=2,
                        min_weight_fraction_leaf=0.0, n_estimators=100,
                        n_jobs=None, oob_score=False, random_state=None,
                        verbose=0, warm_start=False)
```

In [33]:

```
pred = rf.predict(x_test)
```

In [34]:

```
pred
```

Out[34]:

```
array([1, 5, 9, ..., 1, 4, 4], dtype=int64)
```

In [35]:

```
s = y_test.values

count = 0
for i in range(len(pred)):
    if pred[i] == s[i]:
        count = count+1
```

In [36]:

```
count
```

Out[36]:

```
13522
```

In [37]:

```
len(pred)
```

Out[37]:

```
14000
```

In [38]:

```
13522/14000
```

Out[38]:

```
0.9658571428571429
```

In []: