In [128]:
```
import matplotlib.pyplot as plt
import numpy as np
import scipy as sci
import pandas as pd
import seaborn as sns
import matplotlib
import sklearn.datasets as data
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
```

In [129]:
```
mnist = data.fetch_openml('mnist_784')
```

In [130]:
```
X = mnist['data']
y = mnist['target']
columns = mnist['feature_names']
```

In [131]:
```
df = pd.DataFrame(X,y, columns=columns)
```

In [132]:
```
mnist_np = df.to_numpy()
```

In [133]:
```
Xtrain, Xtest, Ytrain, Ytest = train_test_split(mnist_np, df.index, test_size
= .1)
```

In [146]:
```
lr = LogisticRegression('none',solver ='saga', multi_class='multinomial',max_i
ter= 100, random_state=0).fit(Xtrain, Ytrain)
l1 = LogisticRegression('l1',solver ='saga', multi_class='multinomial', max_it
er = 100, random_state=0).fit(Xtrain, Ytrain)
l2 = LogisticRegression('l2',solver ='saga', multi_class='multinomial', max_it
er=100, random_state=0).fit(Xtrain, Ytrain)
```

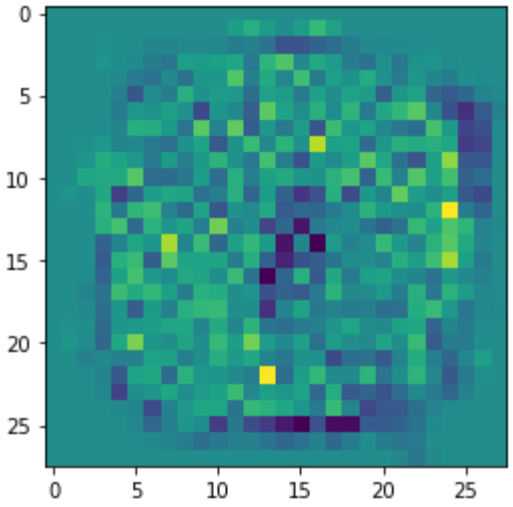
```
F:\Anaconda\lib\site-packages\sklearn\linear_model\_sag.py:329: ConvergenceWa
rning: The max_iter was reached which means the coef_ did not converge
  warnings.warn("The max_iter was reached which means "
F:\Anaconda\lib\site-packages\sklearn\linear_model\_sag.py:329: ConvergenceWa
rning: The max_iter was reached which means the coef_ did not converge
  warnings.warn("The max_iter was reached which means "
F:\Anaconda\lib\site-packages\sklearn\linear_model\_sag.py:329: ConvergenceWa
rning: The max_iter was reached which means the coef_ did not converge
  warnings.warn("The max_iter was reached which means "
```

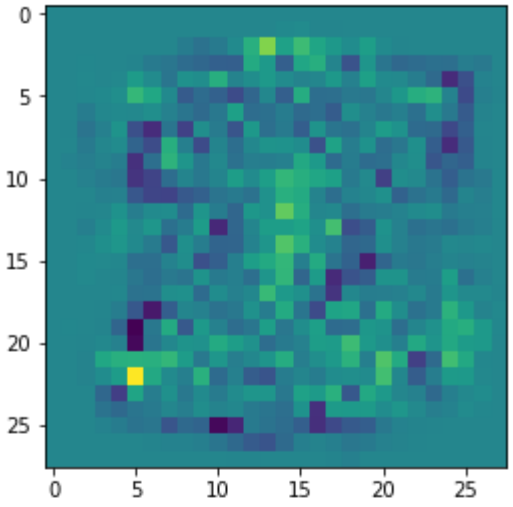
In [147]:
```
score_lr_test = lr.score(Xtest, Ytest)
score_l1_test = l1.score(Xtest, Ytest)
score_l2_test = l2.score(Xtest, Ytest)
```

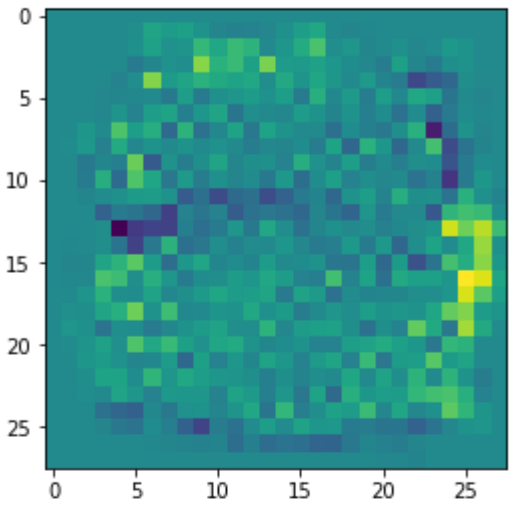
In [148]:
```
score_lr_train = lr.score(Xtrain, Ytrain)
score_l1_train = l1.score(Xtrain, Ytrain)
score_l2_train = l2.score(Xtrain, Ytrain)
```

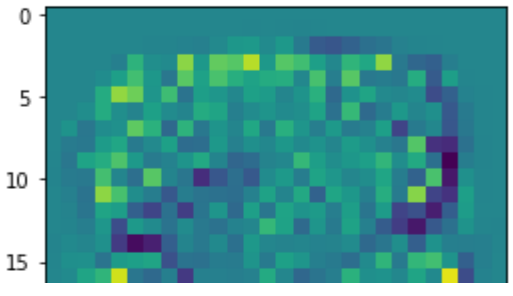
In [149]:
```
img = lr.coef_
```

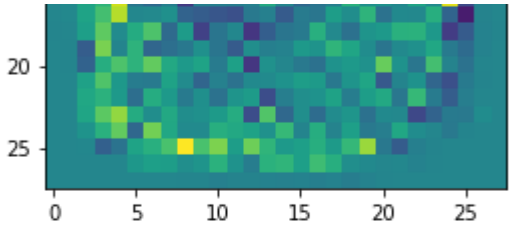
In [150]:
```
fig = plt.figure(figsize=(50, 50))

for i in range(10):
    sub = fig.add_subplot(10, 1, i + 1)
    sub.imshow(img[i,:].reshape(28,28), interpolation='nearest')
```

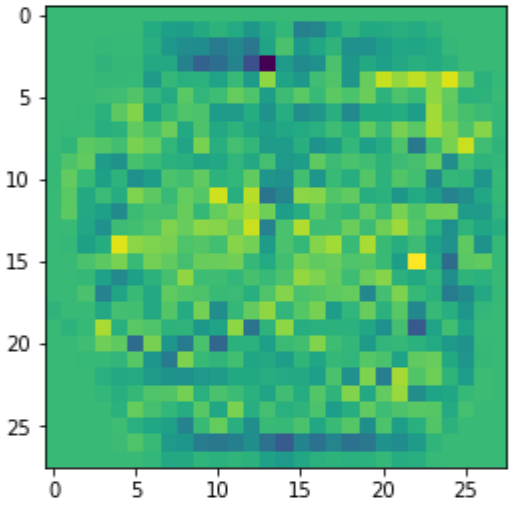

0
5
10
15
20
25
0
5
10
15
20
25

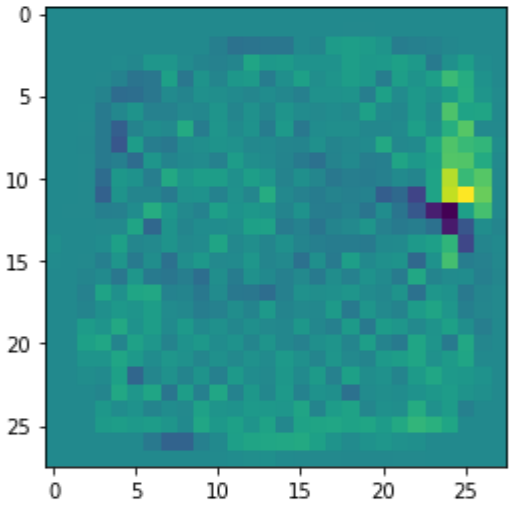

0
5
10
15
20
25
0
5
10
15
20
25

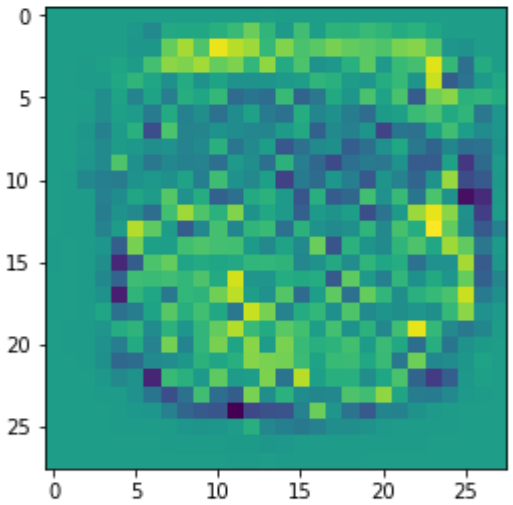

0
5
10
15
20
25
0
5
10
15
20
25


0
5
10
15

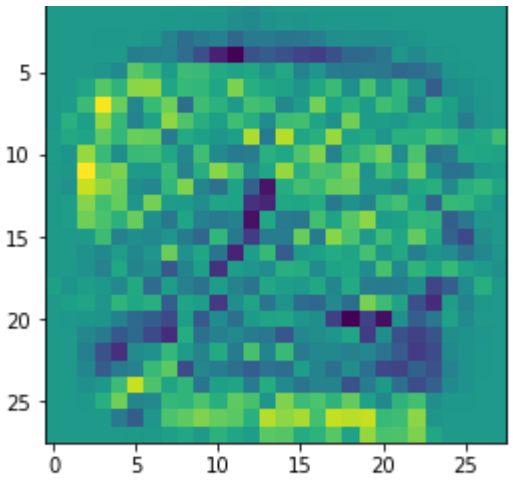

0
5
10
15
20
25

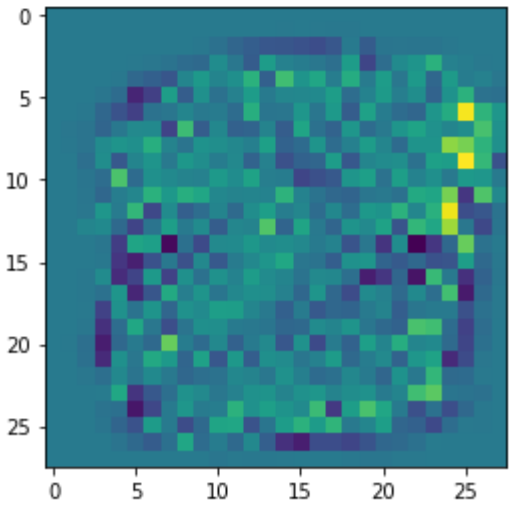

0
5
10
15
20
25

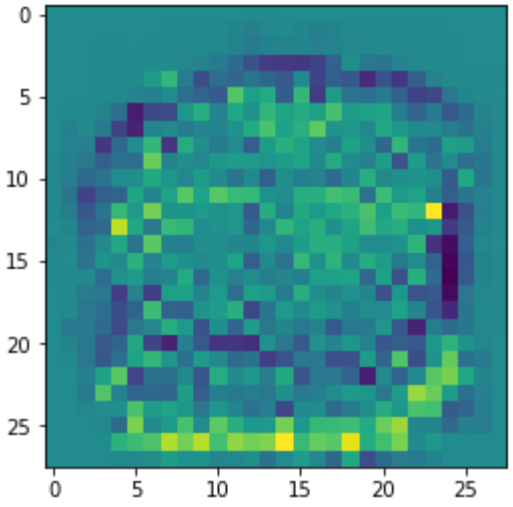

0
5
10
15
20
25


0
5
10
15
20
25


0

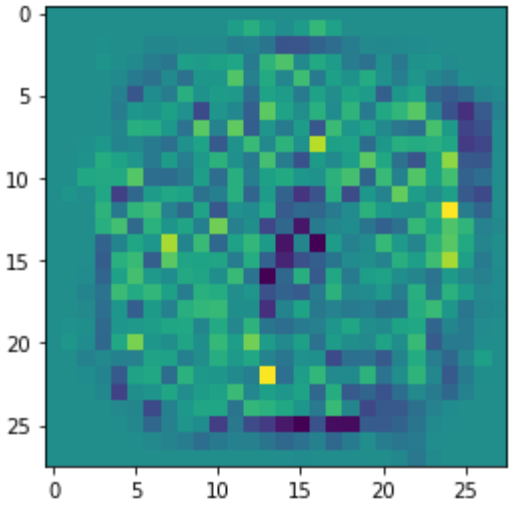


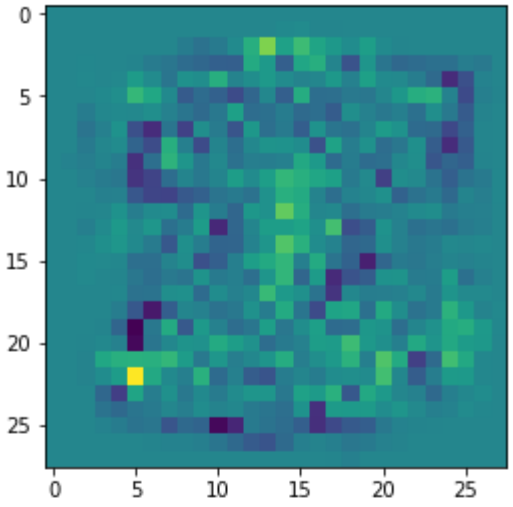


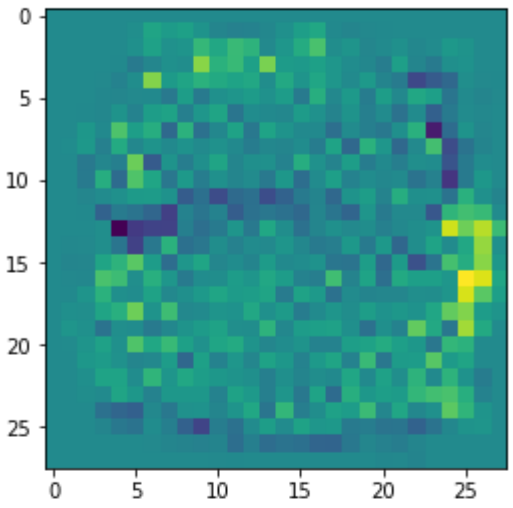


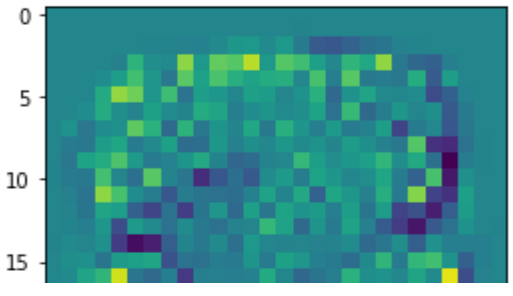
In [151]:
```
img11 = l1.coef_
```

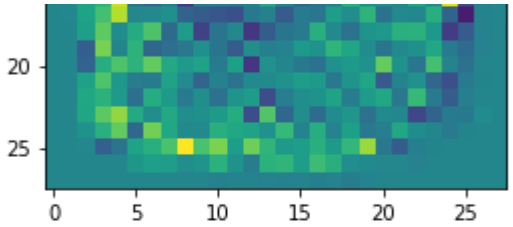
In [152]:
```
fig = plt.figure(figsize=(50, 50))

for i in range(10):
    sub = fig.add_subplot(10, 1, i + 1)
    sub.imshow(imgl1[i,:].reshape(28,28), interpolation='nearest')
```

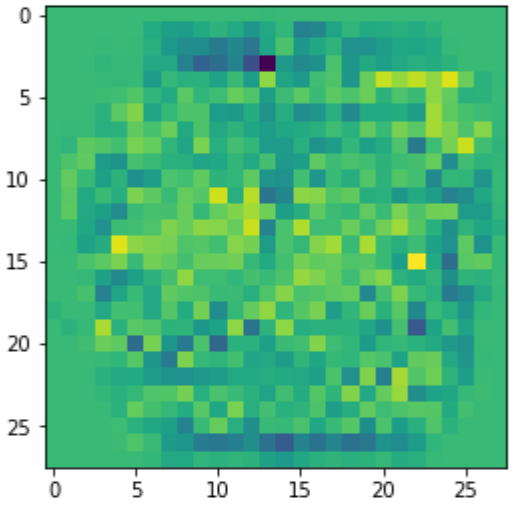

0
5
10
15
20
25
0
5
10
15
20
25

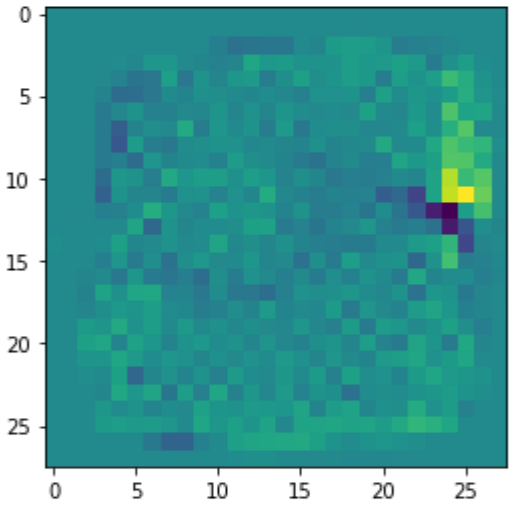

0
5
10
15
20
25
0
5
10
15
20
25

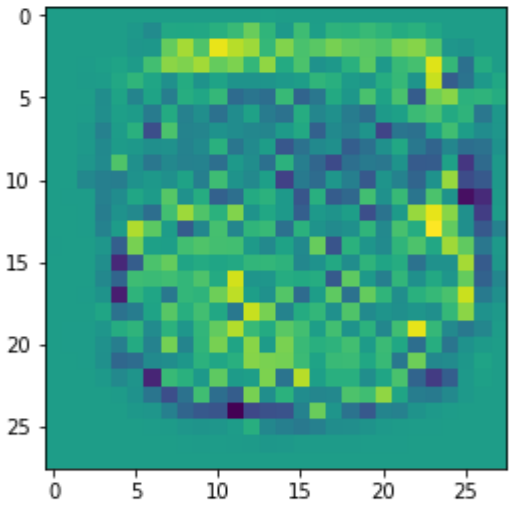

0
5
10
15
20
25
0
5
10
15
20
25


0
5
10
15

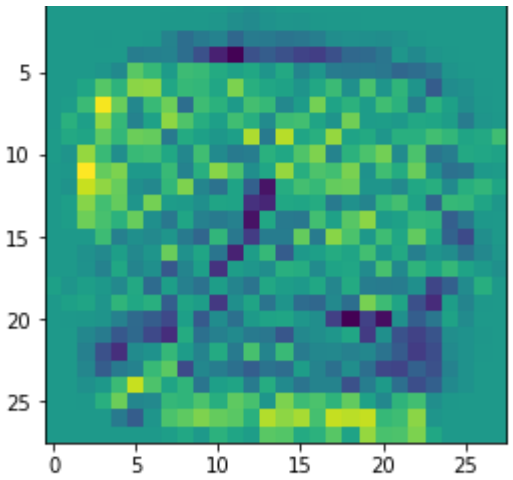

20
25
0
5
10
15
20
25

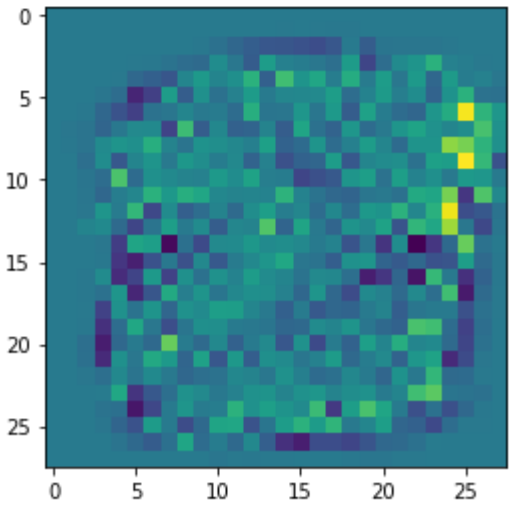

0
5
10
15
20
25
0
5
10
15
20
25

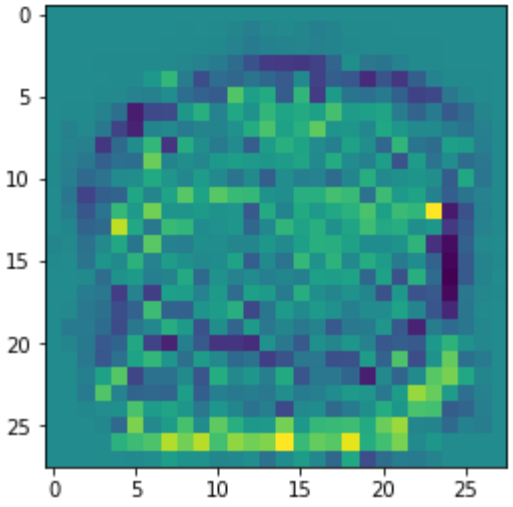

0
5
10
15
20
25
0
5
10
15
20
25


0
5
10
15
20
25
0
5
10
15
20
25


0

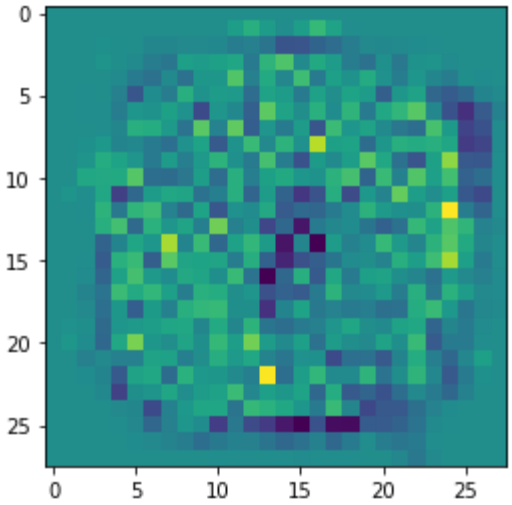


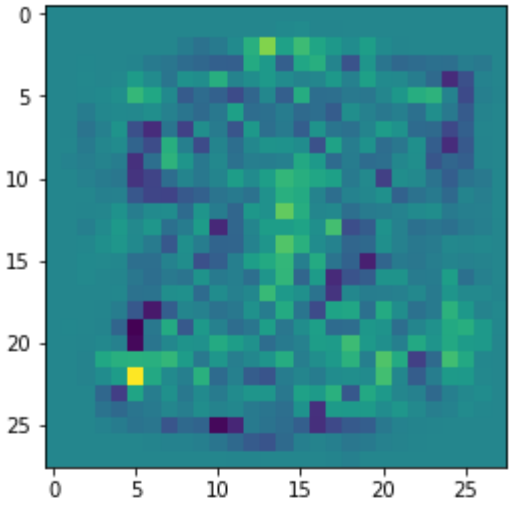


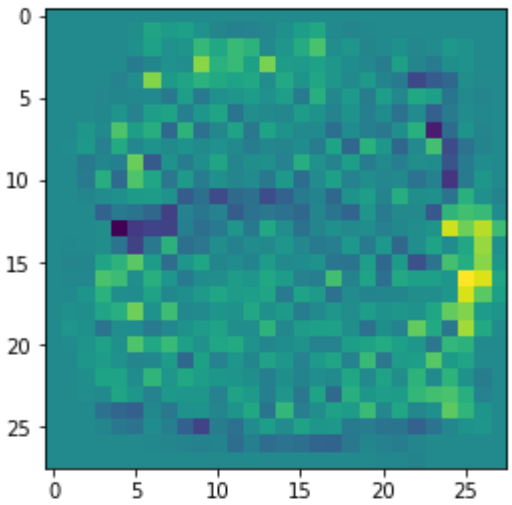


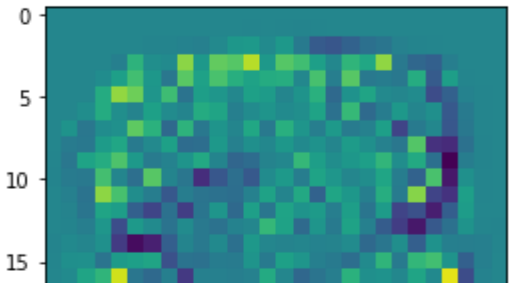
In [153]:
```
img12 = l2.coef_
```

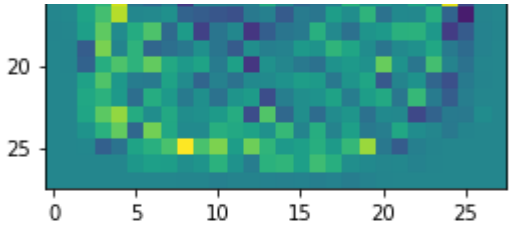
In [154]:
```
fig = plt.figure(figsize=(50, 50))

for i in range(10):
    sub = fig.add_subplot(10, 1, i + 1)
    sub.imshow(imgl2[i,:].reshape(28,28), interpolation='nearest')
```

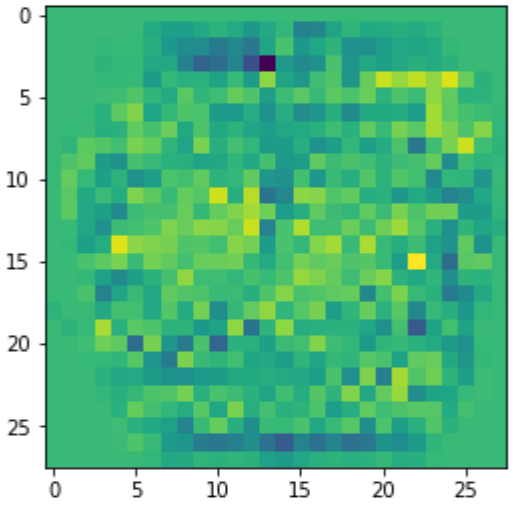

0
5
10
15
20
25
0
5
10
15
20
25

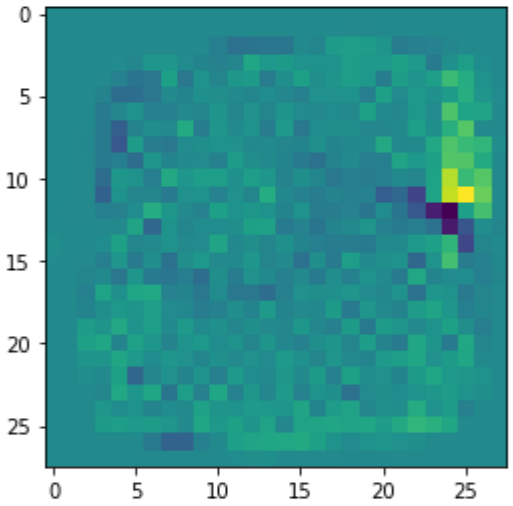

0
5
10
15
20
25
0
5
10
15
20
25

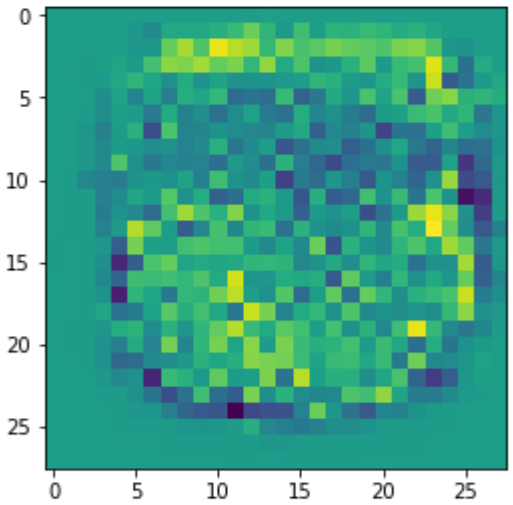

0
5
10
15
20
25
0
5
10
15
20
25


0
5
10
15

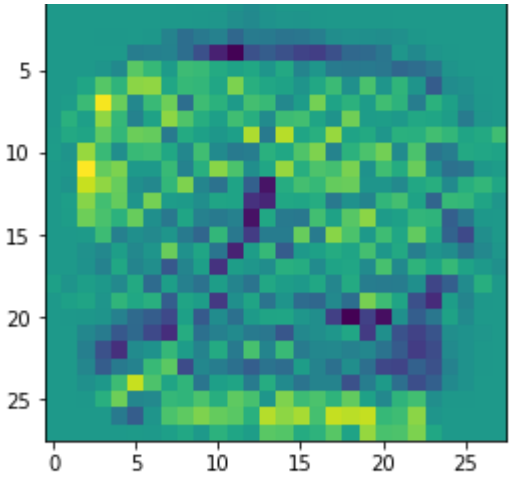

20
25
0
5
10
15
20
25

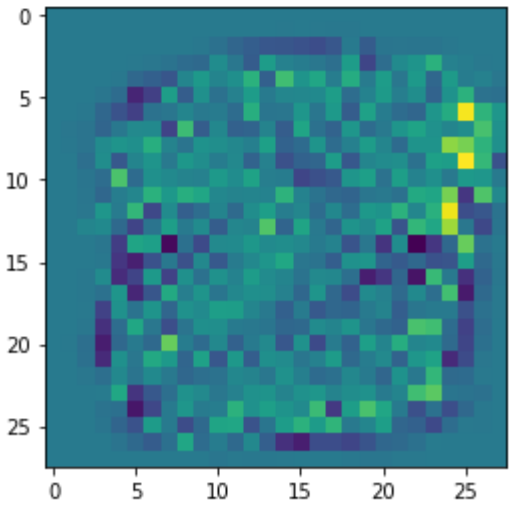

0
5
10
15
20
25
0
5
10
15
20
25

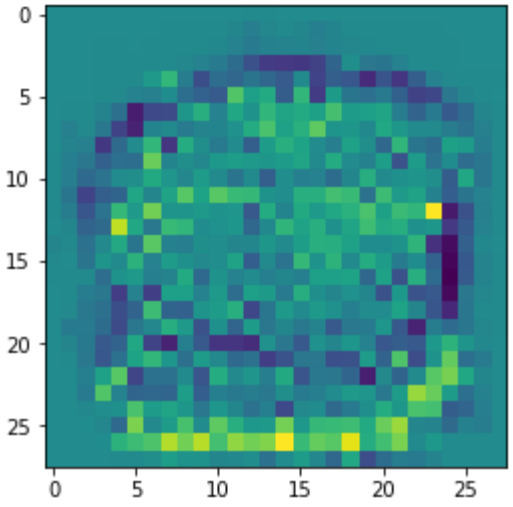

0
5
10
15
20
25
0
5
10
15
20
25


0
5
10
15
20
25
0
5
10
15
20
25


0







In [155]:
```
print(score_lr_train, score_lr_test)
```

```
0.9387936507936508 0.92
```

In [156]:
```
print(score_l1_train, score_l1_test)
```

```
0.9387460317460318 0.92
```

In [157]:
```
print(score_l2_train, score_l2_test)
```

```
0.9387936507936508 0.92
```

In [ ]:
```

```