

assignment6

March 29, 2021

```
[ ]: #importing all the necessary libraries

import numpy as np
import pandas as pd
from sklearn.datasets import load_digits
from sklearn.model_selection import train_test_split
from sklearn.linear_model import Perceptron
from sklearn.metrics import accuracy_score, confusion_matrix
import matplotlib.pyplot as plt
import seaborn as sb
```

```
[ ]: # loading the mnist dataset

digits = load_digits()
data = digits.data
target = digits.target
```

```
[ ]: data.shape
```

```
[ ]: (1797, 64)
```

```
[ ]: data.shape[0]
```

```
[ ]: 1797
```

```
[ ]: toview = data.reshape(data.shape[0], 8, 8).astype('float32')
```

```
[ ]: toview.shape
```

```
[ ]: (1797, 8, 8)
```

```
[ ]: toview[target == 0].shape # 178 8X8 arrays are there with 0 as target
```

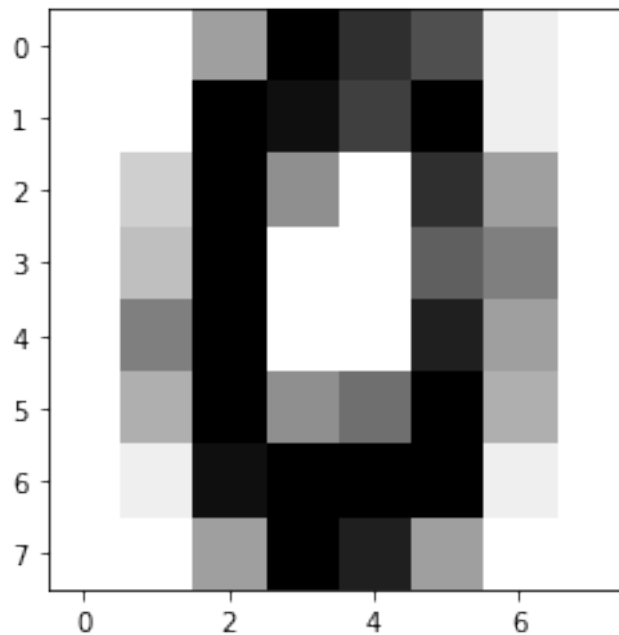
```
[ ]: (178, 8, 8)
```

```
[ ]: toview[target == 0][29].shape
```

```
[ ]: (8, 8)
```

```
[ ]: plt.imshow(toview[target == 0][177], cmap=plt.cm.binary)
```

```
[ ]: <matplotlib.image.AxesImage at 0x7ff5b3880b10>
```



```
[ ]: def oneVsAll(X, y, num_labels):
    m, n = X.shape
    y_t = np.ones(shape = (len(y), num_labels))
    y = y.reshape(len(y))

    for i in range(0, num_labels):
        y_t[:, i] = y
        y_t[y != i, i] = 0
        y_t[y == i, i] = 1

    for i in range(0, num_labels):
        X_train, X_test, y_train, y_test = train_test_split(X, y_t[:, i],
        ↪test_size = 0.2, stratify = y_t[:, i])
        pcg = Perceptron(max_iter = 1000, eta0 = 0.1, random_state = 0)
        pcg.fit(X_train, y_train)
        preds = pcg.predict(X_test)
        print("class", i, "accuracy")
        print(accuracy_score(y_test, np.round(preds)))

    cm = confusion_matrix(y_test, preds)
    cm_df = pd.DataFrame(cm)
```

```

plt.figure(figsize=(8, 6))
sb.heatmap(cm_df, annot = True, cmap = plt.cm.Blues)
plt.title('Perceptron \nAccuracy:{0:.3f}'.format(accuracy_score(y_test,
→preds)))
plt.ylabel('True label')
plt.xlabel('Predicted label')
plt.show()

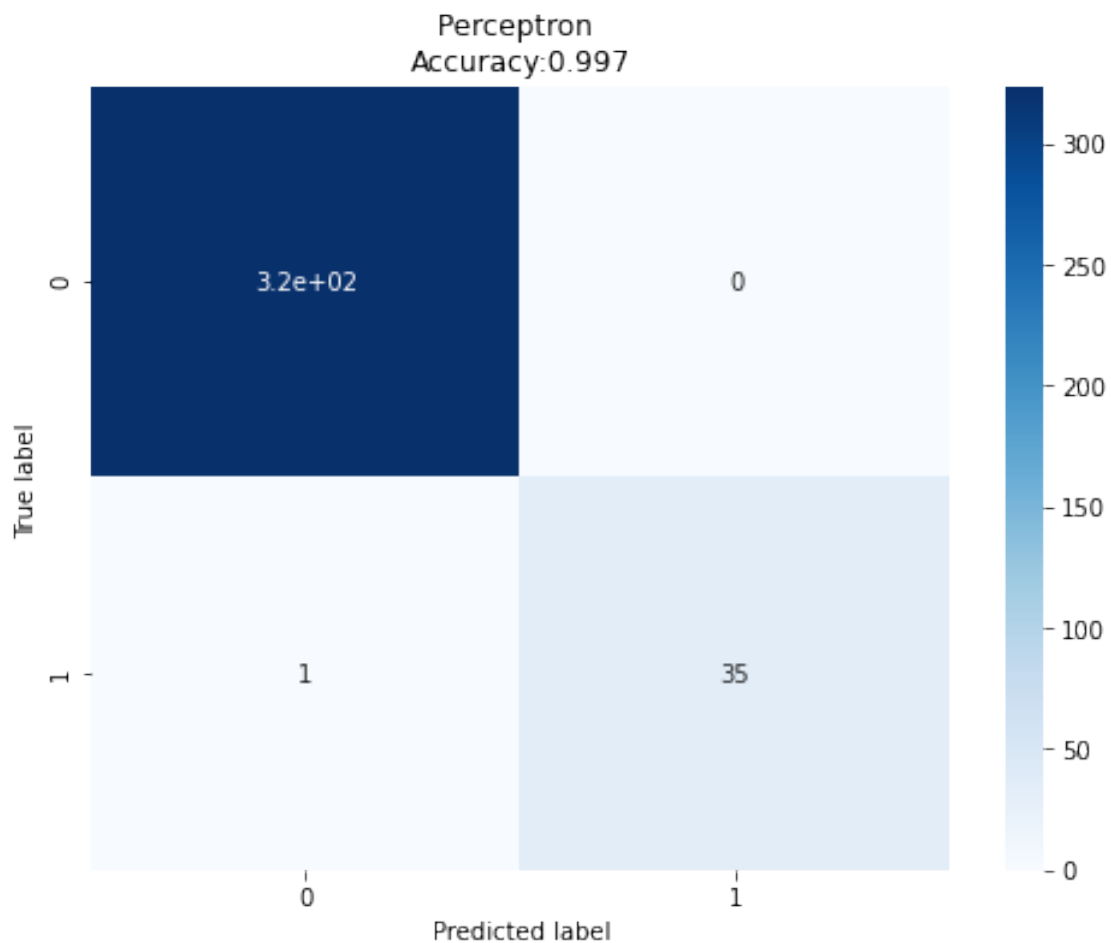
```

```

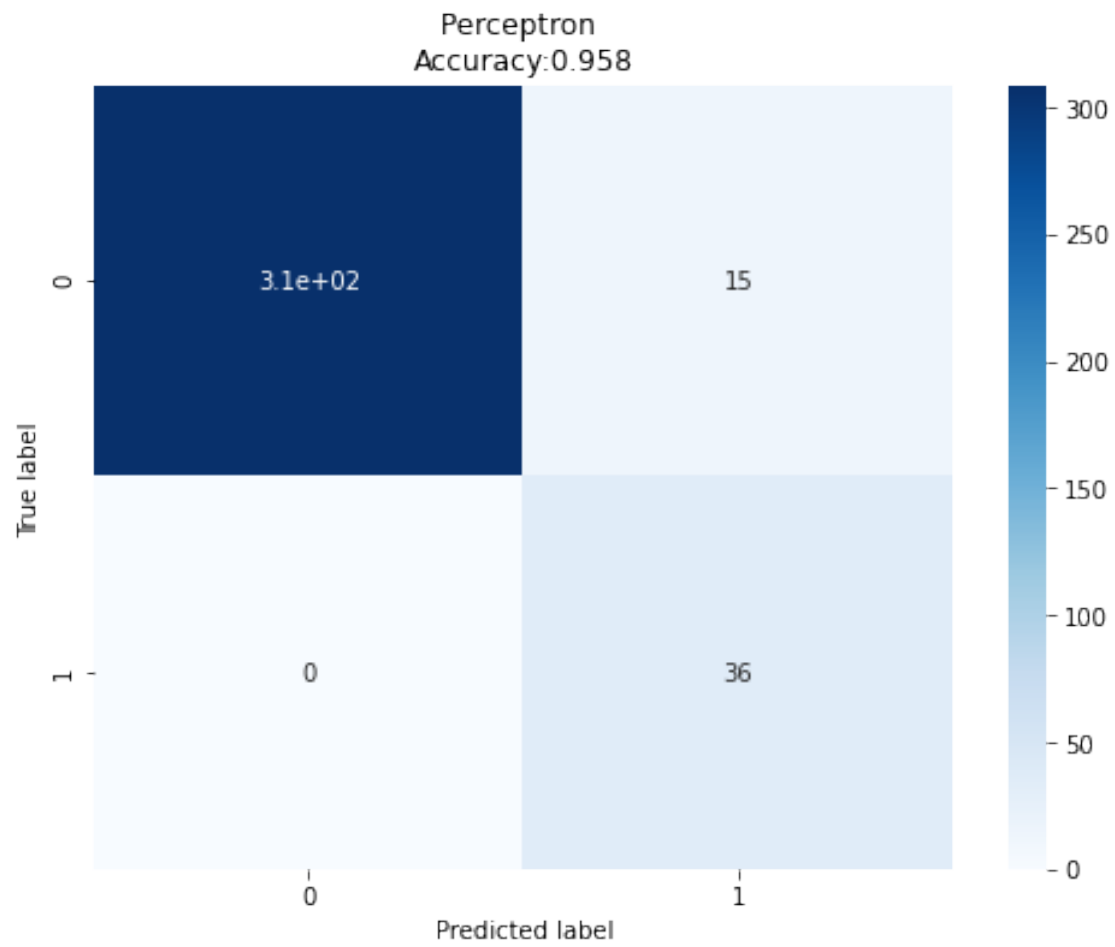
[ ]: X = data.astype("float32") / 255
y = target
oneVsAll(X, y, 10)

```

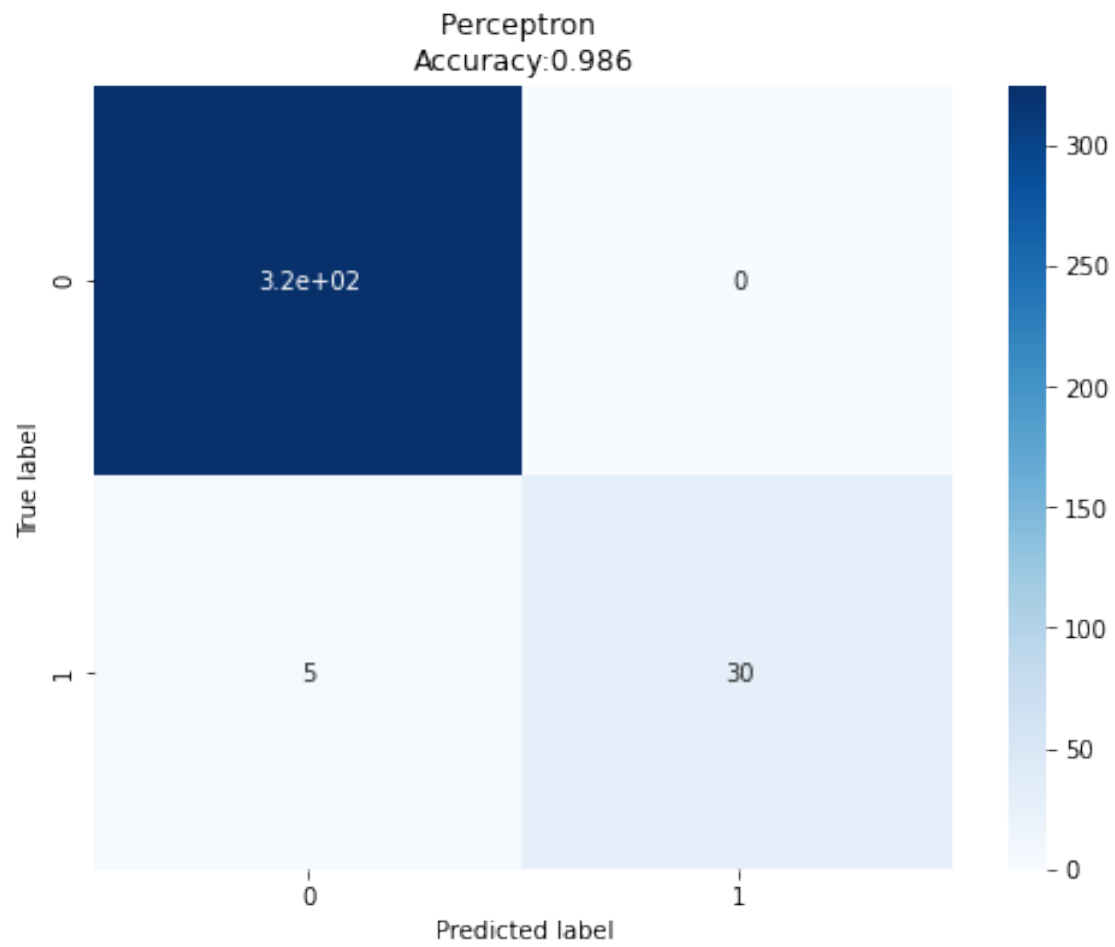
class 0 accuracy
0.9972222222222222



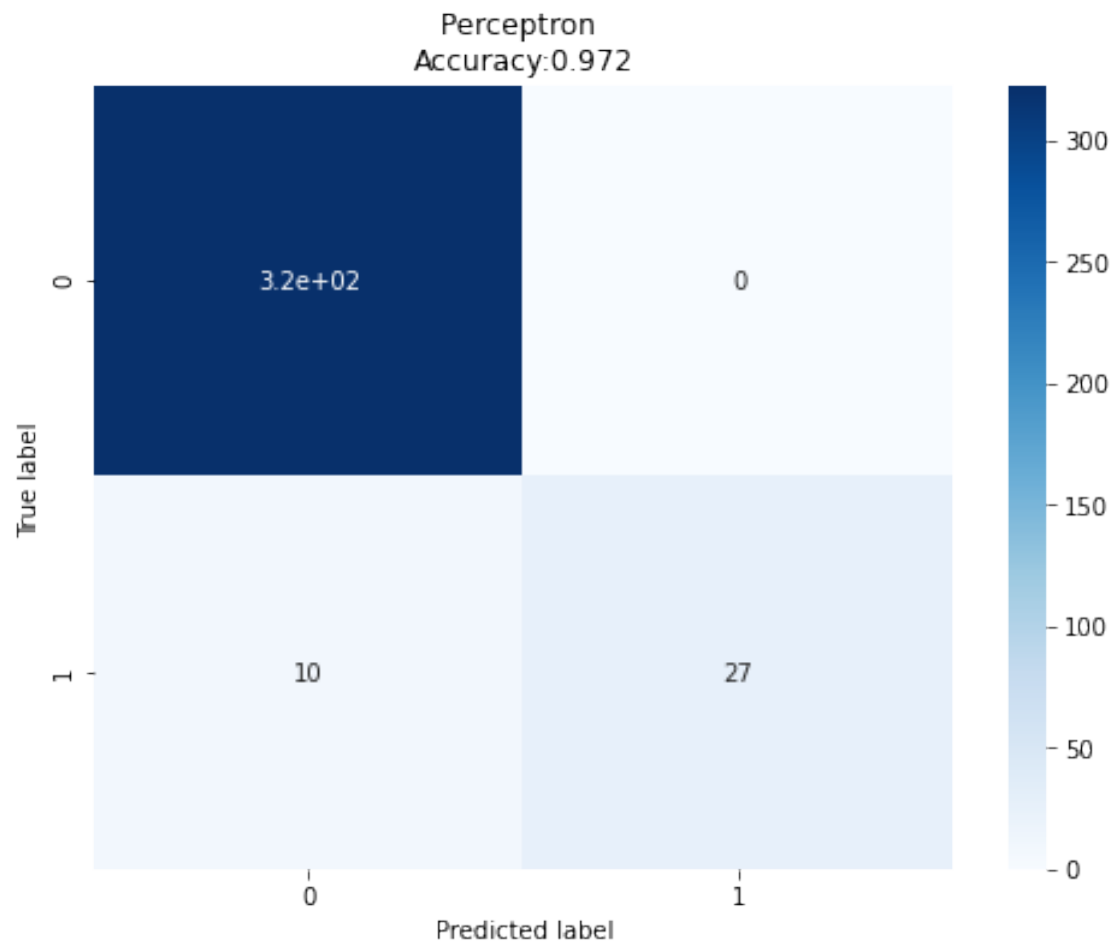
class 1 accuracy
0.9583333333333334



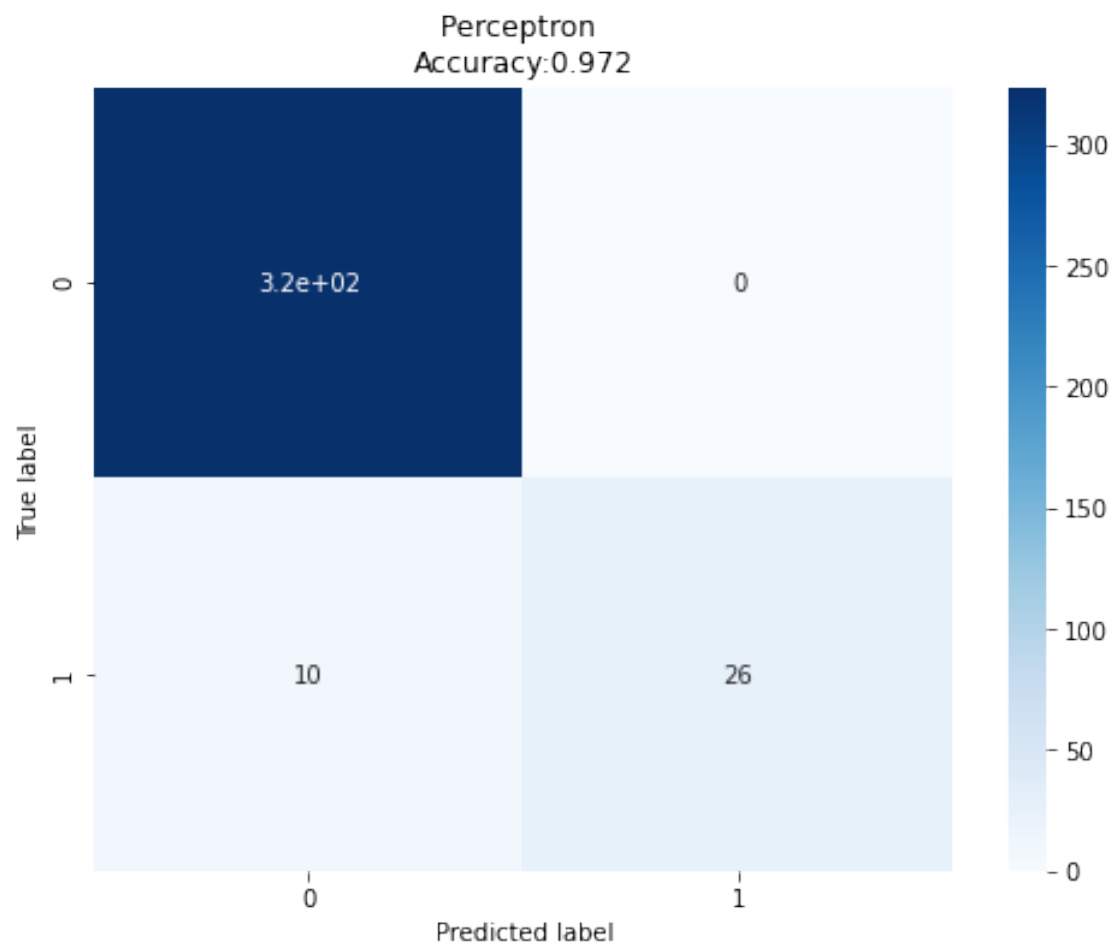
class 2 accuracy
0.9861111111111112



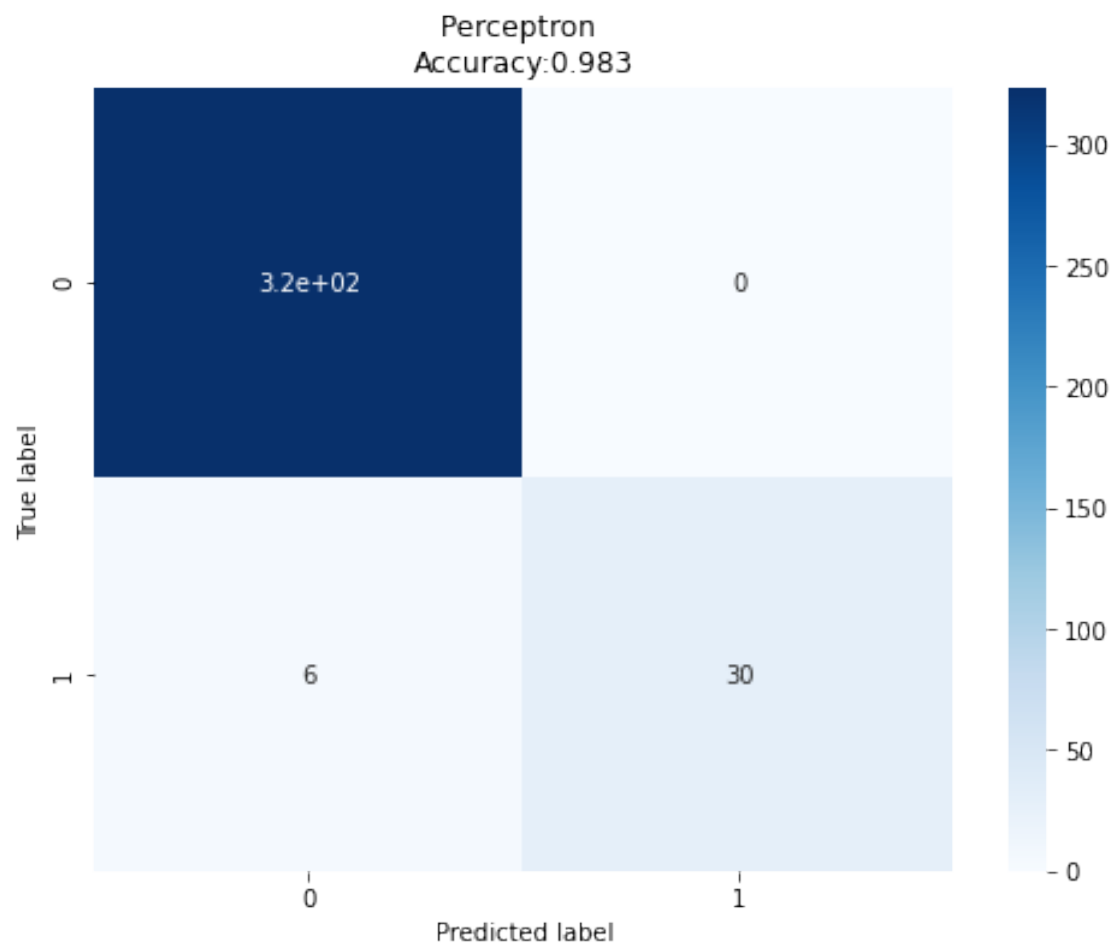
class 3 accuracy
0.9722222222222222



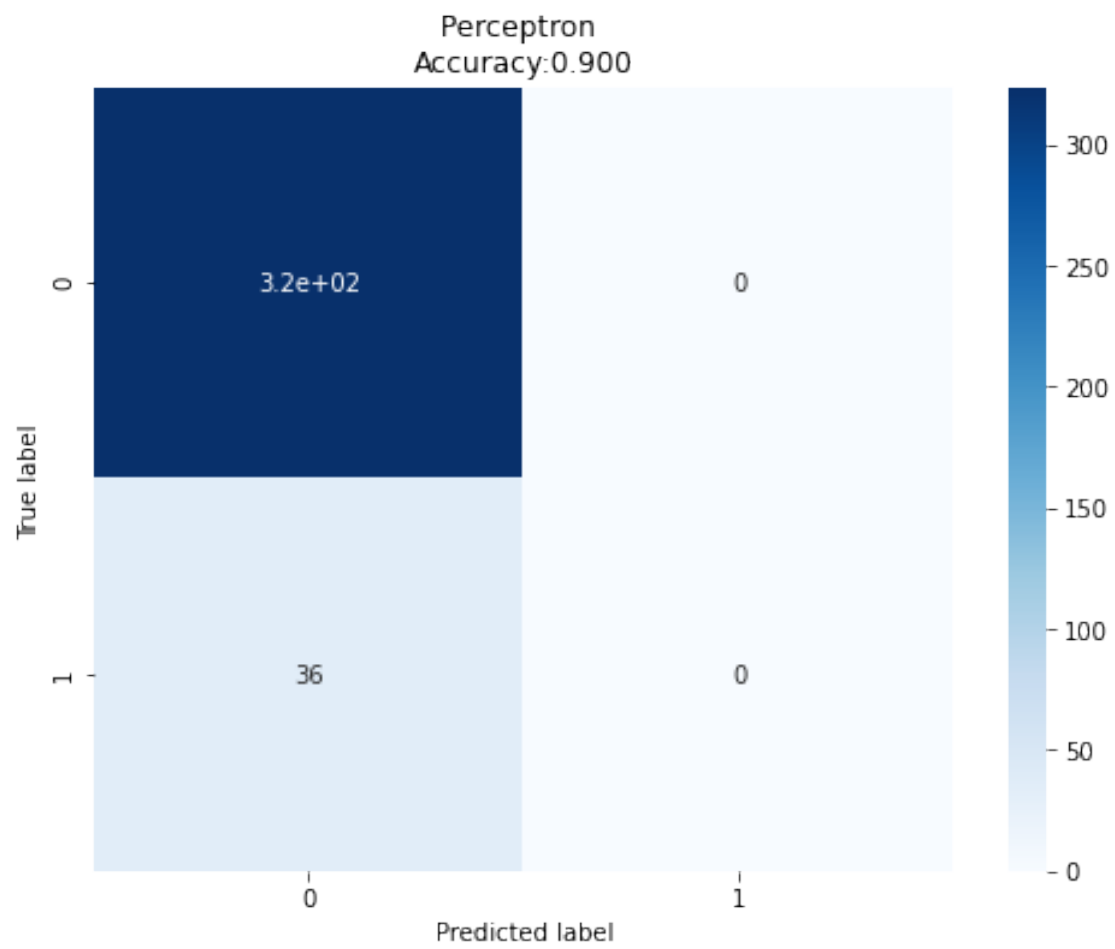
class 4 accuracy
0.9722222222222222



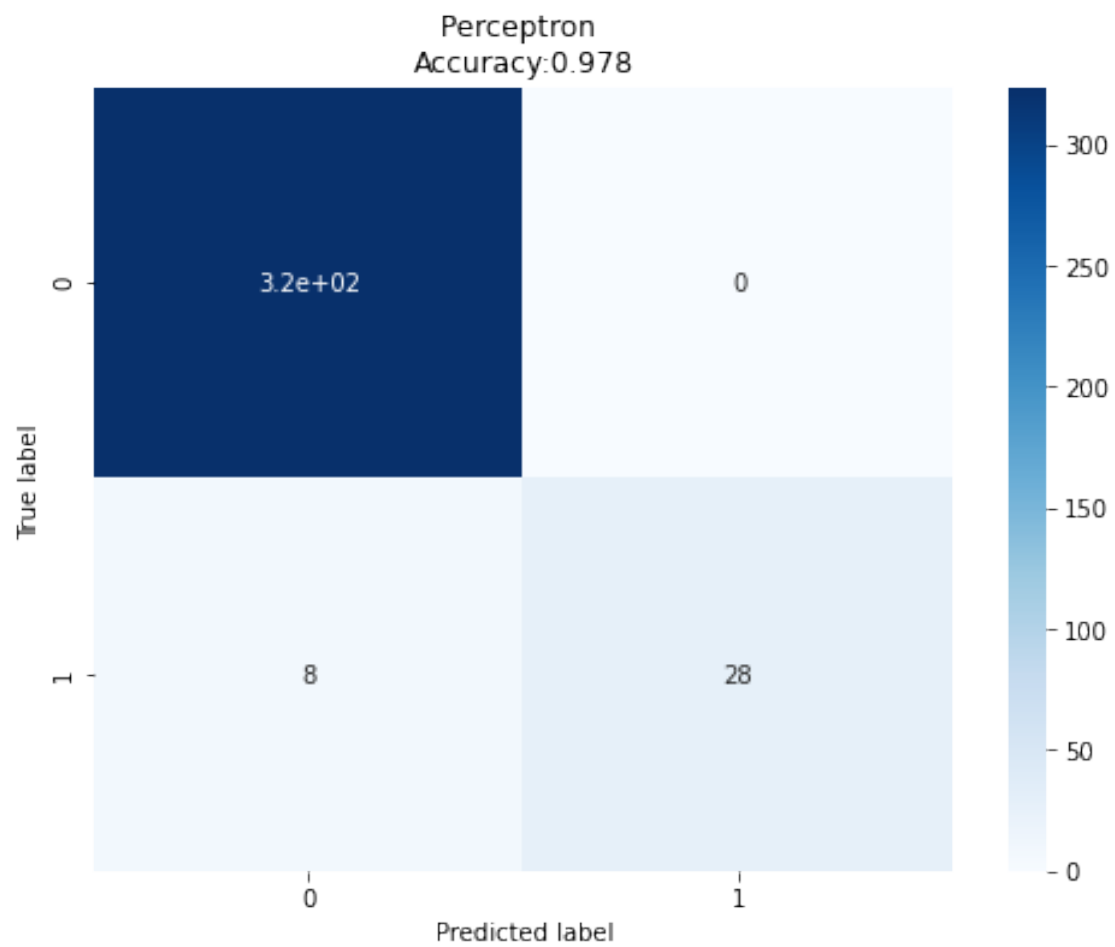
class 5 accuracy
0.9833333333333333



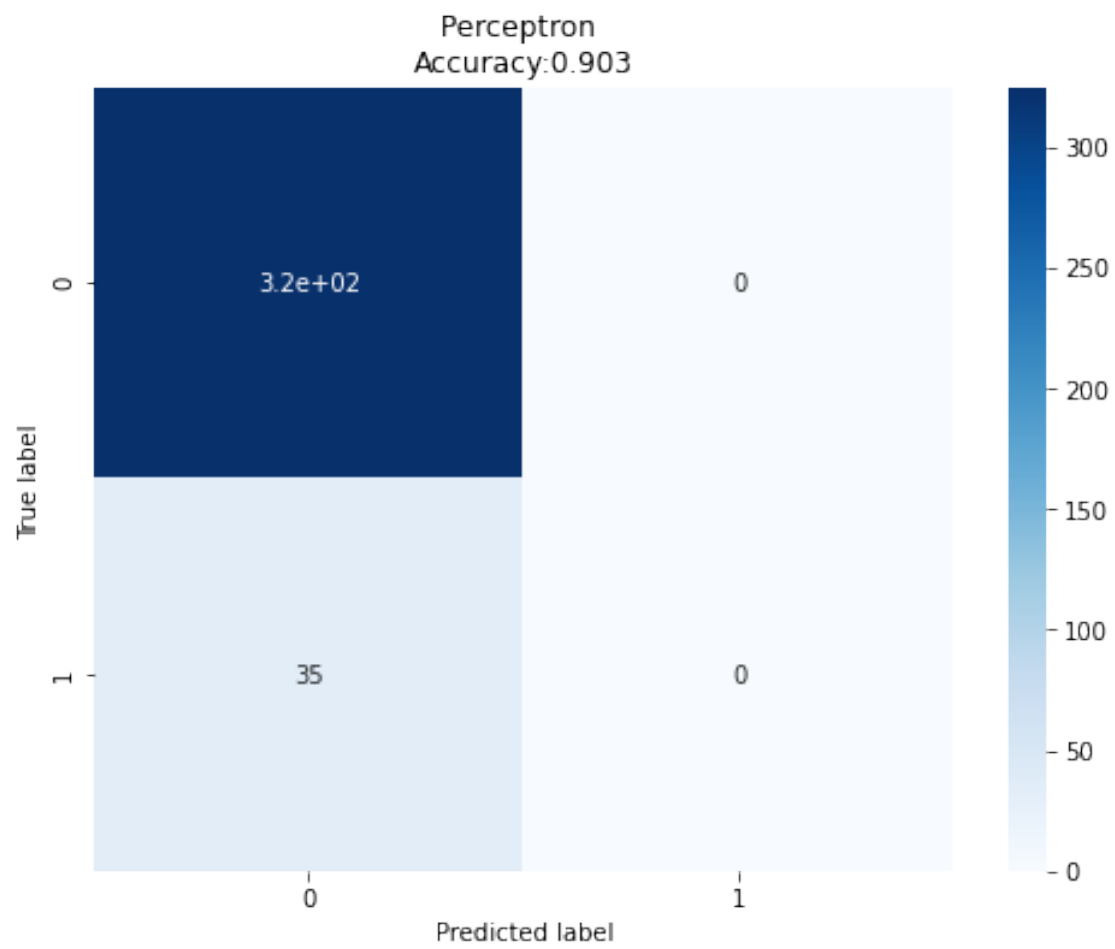
class 6 accuracy
0.9



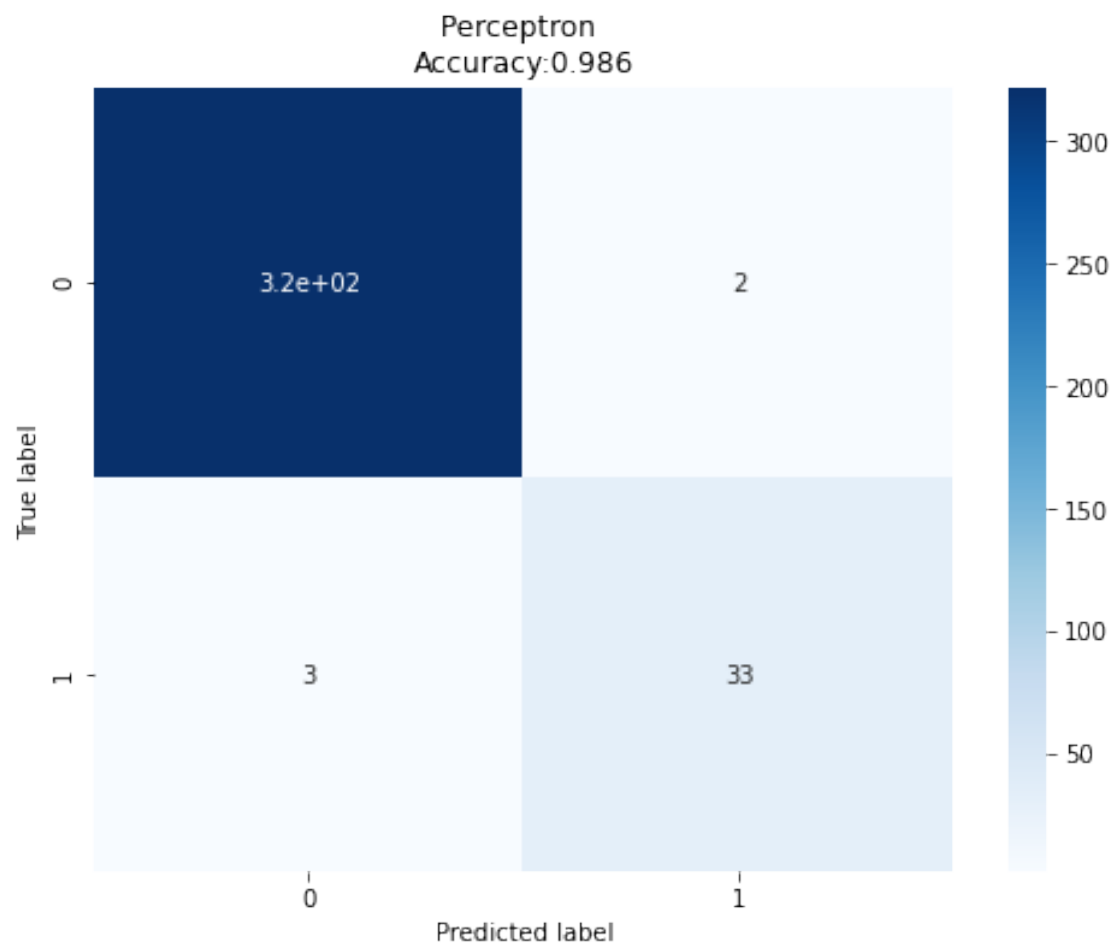
class 7 accuracy
0.9777777777777777



class 8 accuracy
0.9027777777777778



class 9 accuracy
0.9861111111111112



[]: