

```
In [1]: from sklearn import datasets
        from sklearn.preprocessing import scale
        from minisom_new import MiniSom

        # Load the digits dataset from scikit-learn
        # 901 samples, about 180 samples per class
        # the digits represented 0, 1, 2, 3, 4

        digits = datasets.load_digits(n_class=10)
        data = digits.data # matrix where each row is a vector that represent a digit.
        data = scale(data)
        num = digits.target # num[i] is the digit represented by data[i]
```

Training...

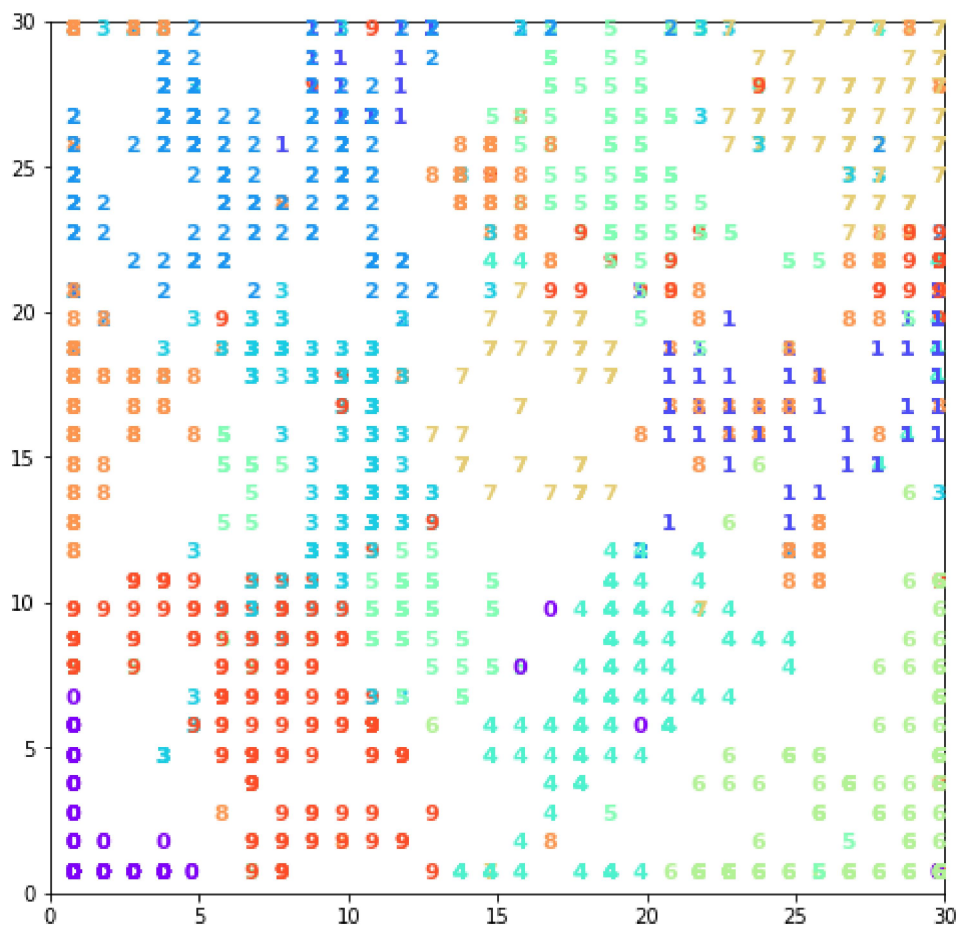
...ready!

```
In [ ]: som = MiniSom(30, 30, 64, sigma=4, learning_rate=0.5)
        som.random_weights_init(data)
        print("Training...")
        som.train_random(data, 5000) # random training
        print("\n...ready!")
```

```

In [7]: import matplotlib.pyplot as plt
plt.figure(figsize=(8, 8))
wmap = {}
im = 0
for x, t in zip(data, num): # scatterplot
    w = som.winner(x)
    wmap[w] = im
    plt.text(w[0]+.5, w[1]+.5, str(t),
             color=plt.cm.rainbow(t / 10.), fontdict={'weight': 'bold', 'size'
             im = im + 1
plt.axis([0, som.get_weights().shape[0], 0, som.get_weights().shape[1]])
plt.savefig('som_digits.png')
plt.show()

```



In []: