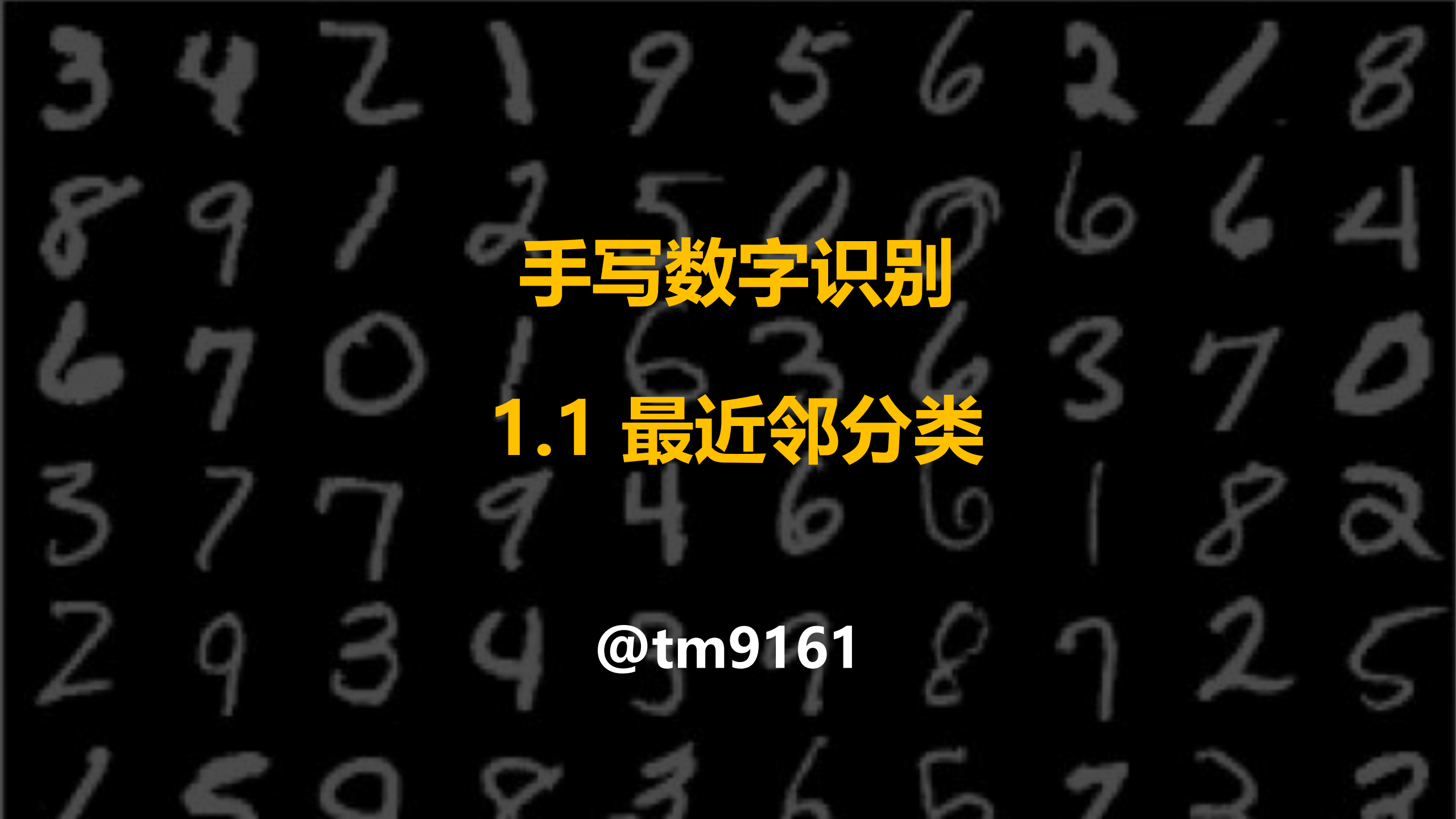


手写数字识别

1.1 最近邻分类

@tm9161

L2

最近邻分类

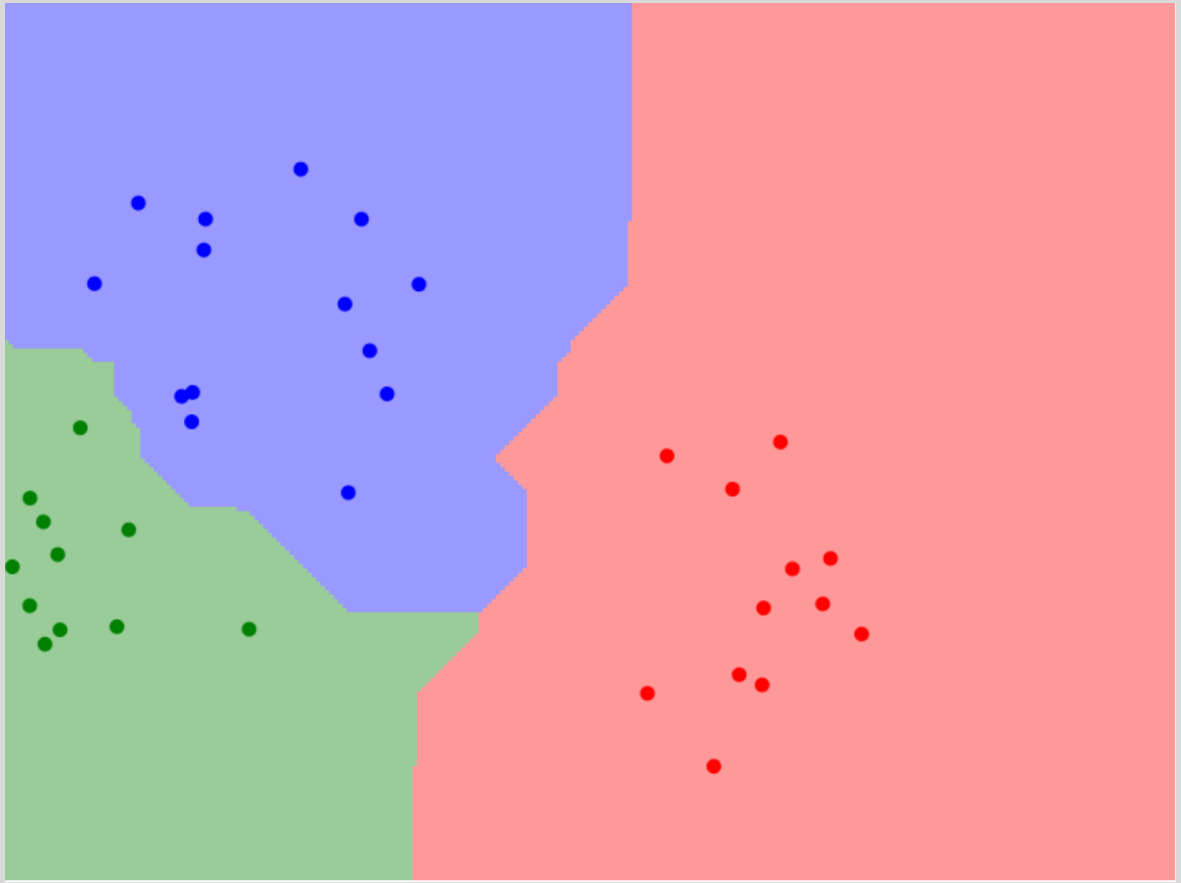
1.最近邻算法原理

2.最近邻手写数字识别

3.算法问题与改进

最近邻 nearest neighbor

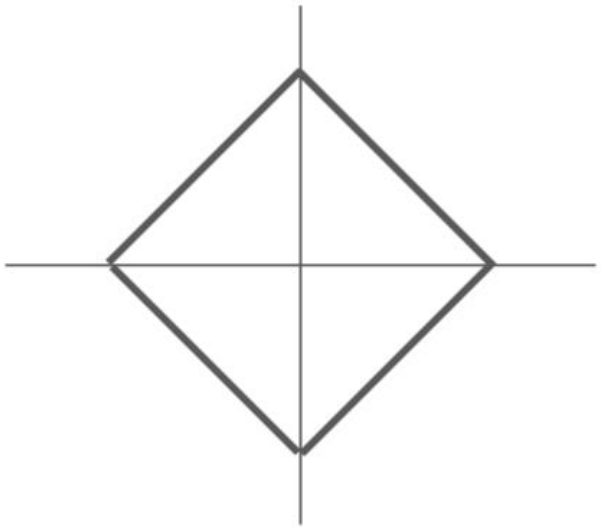
从训练样本中找到与查询点在**距离上最近**的预定数量的多个点，然后依据**这些点**的标签来预测**查询点**的标签。



http://vision.stanford.edu/teaching/cs231n-demos/knn/

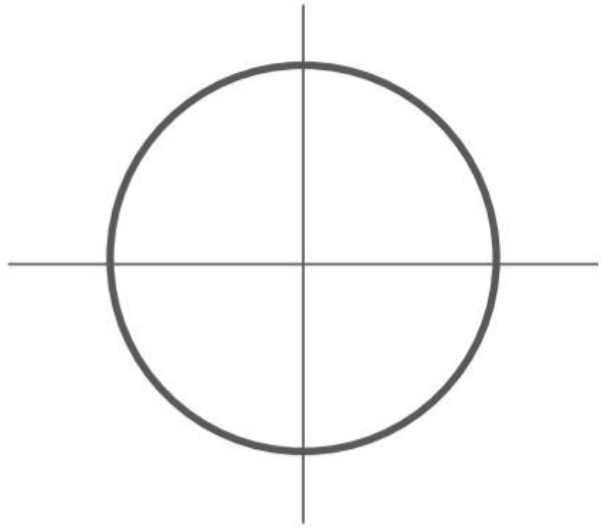
距离度量 Distance Metrics

$$d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$$



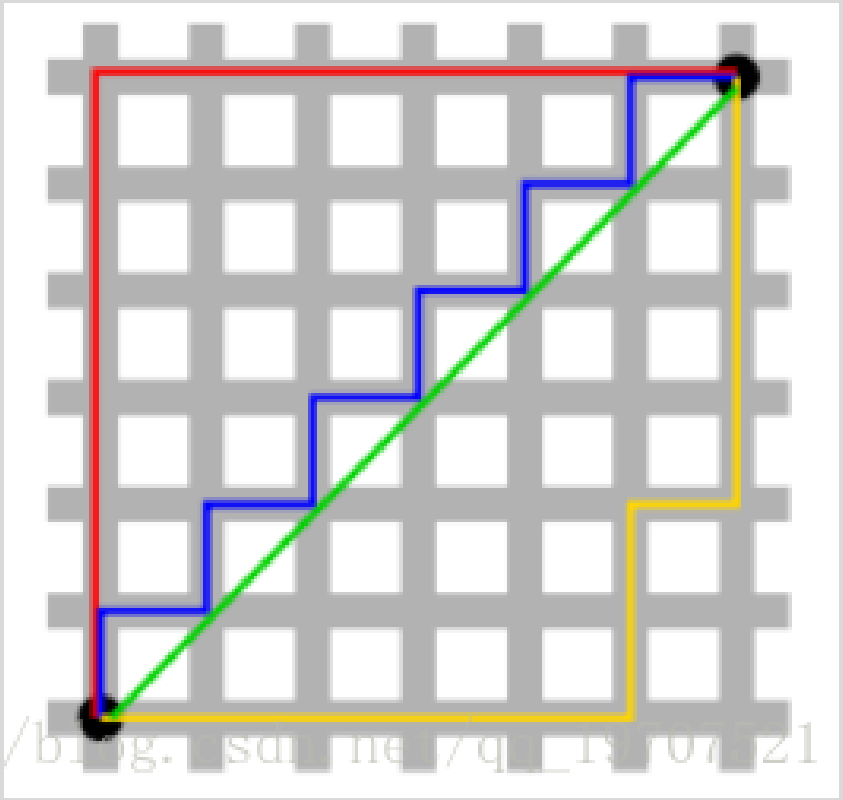
L1 (Manhattan) distance

$$d_2(I_1, I_2) = \sqrt{\sum_p (I_1^p - I_2^p)^2}$$



L2 (Euclidean) distance

L1距离 (Manhattan)

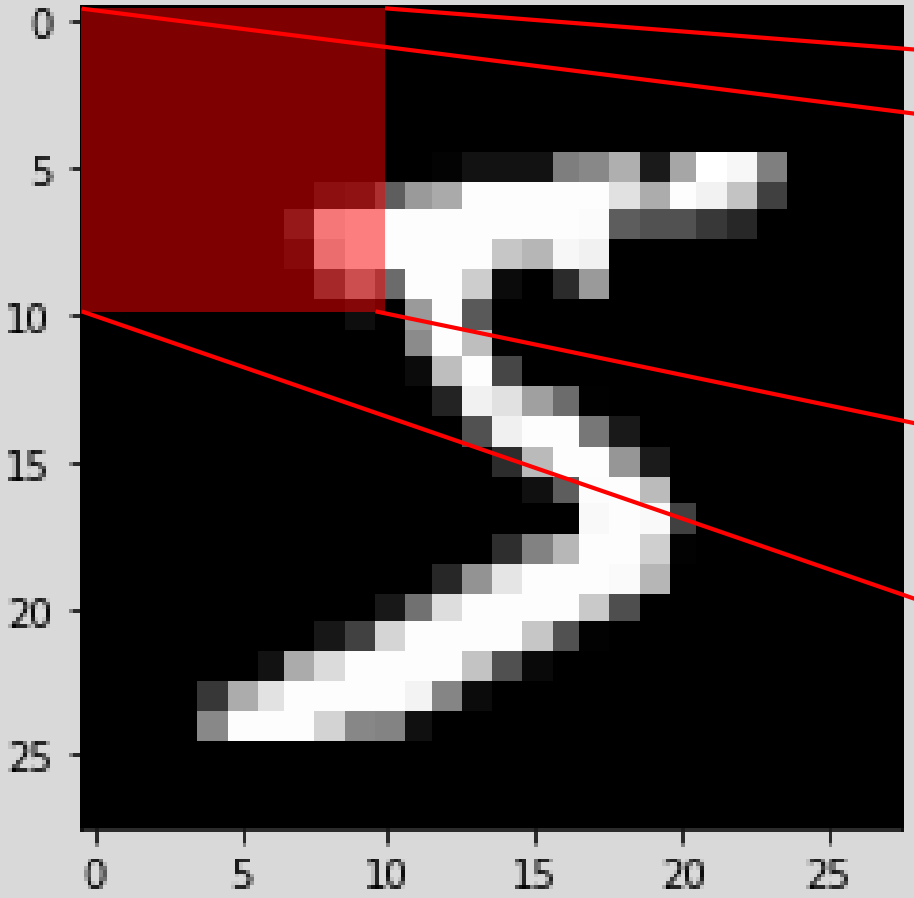


$$d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$$

np.sum(np.abs(test_data-train_data))

求和：np.sum()
绝对值：np.abs()

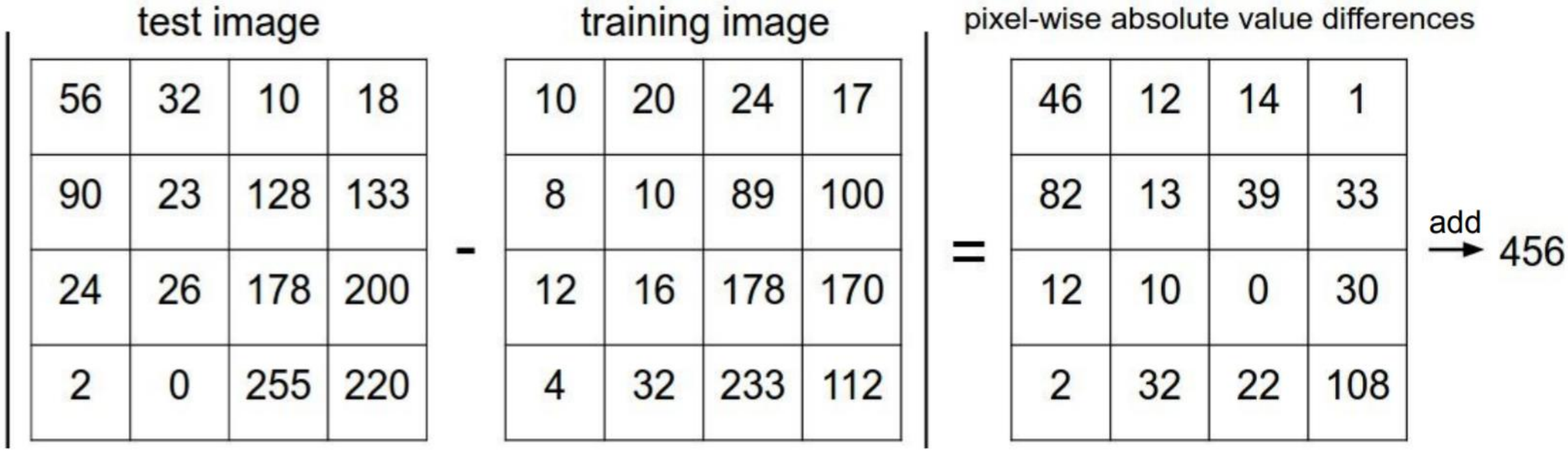
数字化 digitize




```
[[  0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
 [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
 [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
 [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
 [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
 [  0,   0,   0,   0,   0,   0,   0,   0,   0,   0],
 [  0,   0,   0,   0,   0,   0,   0,   0,  30,  36],
 [  0,   0,   0,   0,   0,   0,   0,  49, 238, 253],
 [  0,   0,   0,   0,   0,   0,   0,  18, 219, 253],
 [  0,   0,   0,   0,   0,   0,   0,   0,  80, 156]],
```

Distance Metric to compare images

L1 distance: $$d_1(I_1, I_2) = \sum_p |I_1^p - I_2^p|$$



独热编码 one-hot

5

```
train_labels = np.array(pd.get_dummies(train_labels))
test_labels = np.array(pd.get_dummies(test_labels))
```

```
([0,  0,  0,  0,  0,  1,  0,  0,  0,  0],
```

```
real = np.argmax(train_labels[0])
print(real)
```

```
5
```

```
a = np.array([[1, 5, 5, 2],
              [9, 6, 2, 8],
              [3, 7, 9, 1]])
```

np.argmax(a, axis=1) #按行搜索最大值，返回**位置**。

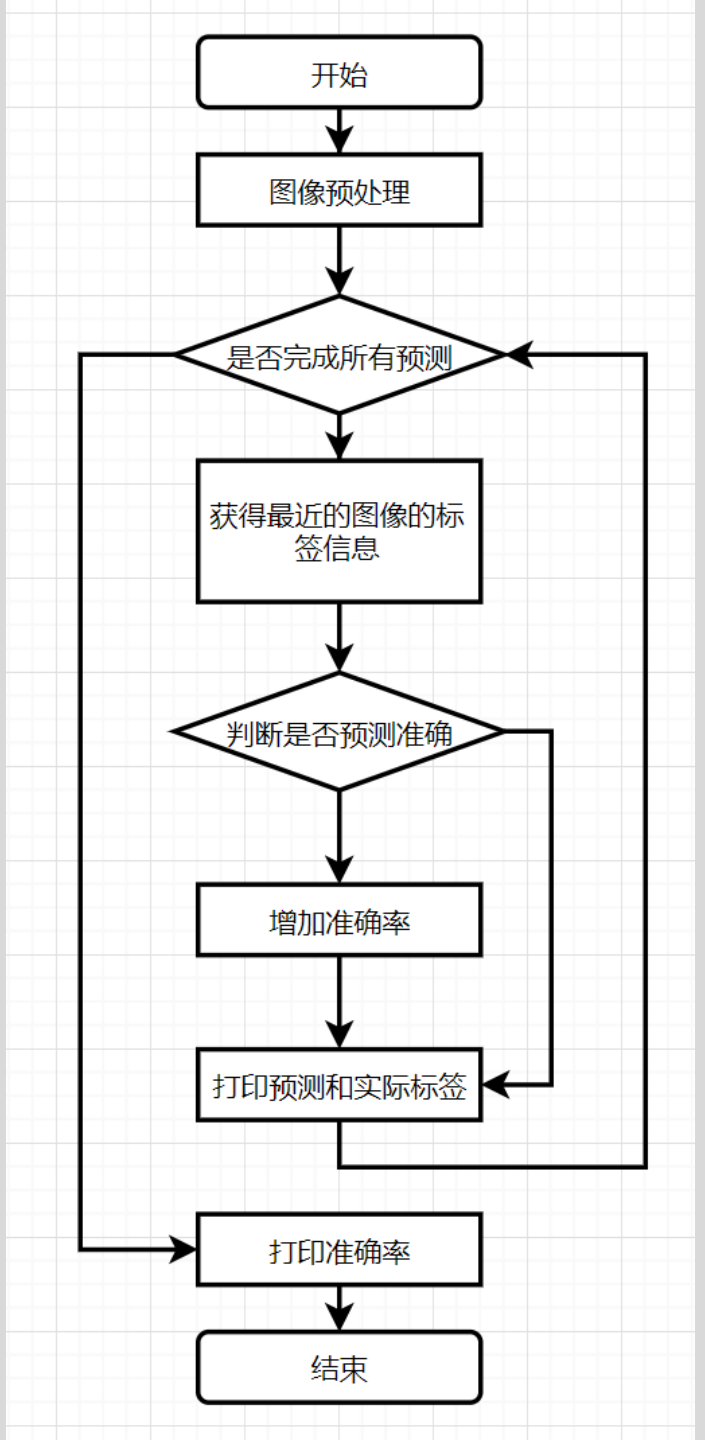
#[1 0 2]

np.argmin(a, axis=0) #按列搜索最小值，返回**位置**。

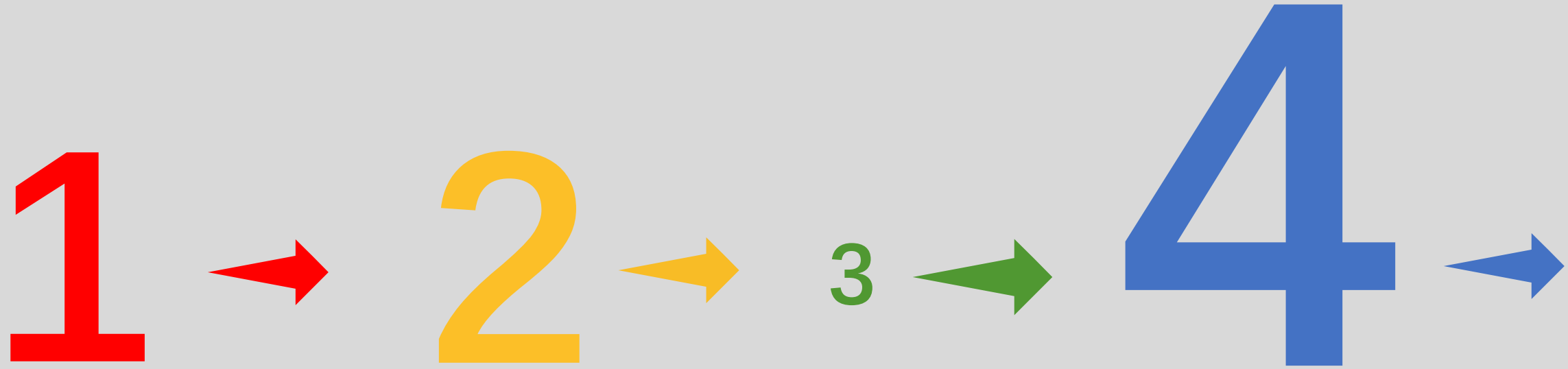
最近邻分类

测试集:
10000

训练集:
60000



最近邻算法



数据采集　建立模型　模型训练 O(1)　模型测试 O(n)

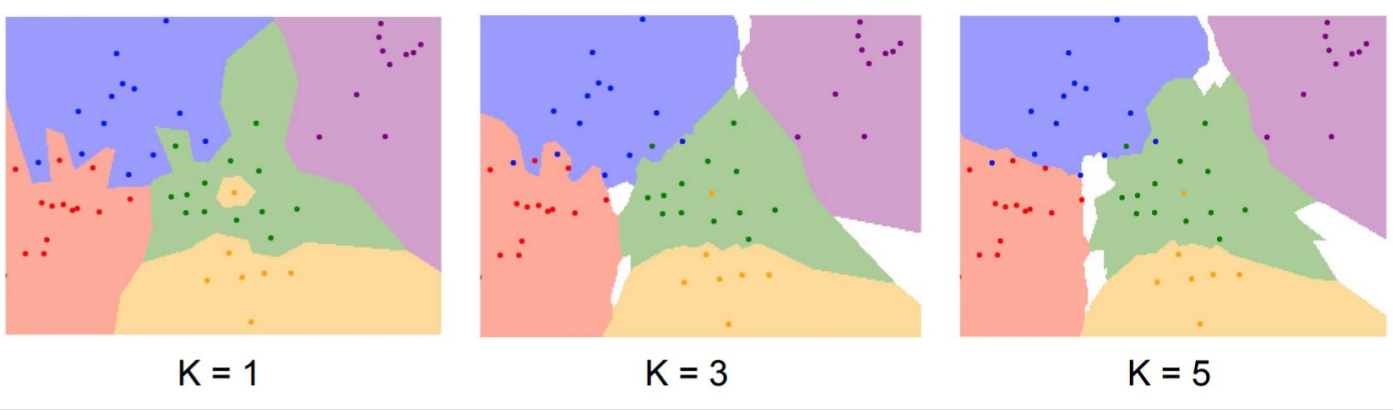
改进&问题?

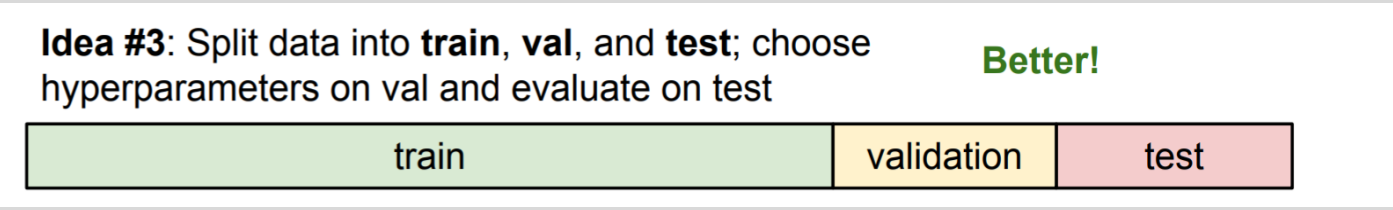
改进：

1.距离度量的改进
（L1、L2和其他距离）

2.K最近邻算法
（K值、权重不同）

3.数据集（验证）





问题：

1.距离不能反映差别

2.算力的问题

参考资料：

1.【子豪兄】精讲CS231N斯坦福计算机视觉公开课
https://www.bilibili.com/video/BV1K7411W7So

2. Machine Learning入门TF2.0（KNN手写识别算法实现）
https://blog.csdn.net/weixin_44307764/article/details/102353344