以fashion\_mnist数据集为例：

一张数据是28\*28=784个像素点，

随机生成n个大小为784的数据，通过模型的输出是10维标签

用每张数据的前9维标签拼接存取数据，第10维要空出来，才能进行优化

则9n>784，n=88, 所以设置88张图片，存取88\*9=792像素，然后用前784位窃取数据

训练时构建恶意项为拼接图片和要窃取数据的差值绝对值的均值，加入模型损失函数优化

Using fashion\_mnist dataset as example:

One data is 28’28 = 784 pixels

We randomly generate n data with 784 dimensions, the model output of one of them is 10 dimensions label, we use the forward 9 dimensions of the output label, put them together to steal the data, we should not use the last dimension to store data, thus there are enough space for optimizing.

So 9 labels of n data should steal 784 pixels, thus 9\*n>784, so n is 88, we randomly generate 88 data to steal 792 pixels, and use 784 pixels of it to recover data.

During training, regular is the distance between origin data and stolen data, we add it to the loss function of model to train.

Acc：0.8865

stole data

 origin data

Acc：0.8852

 stole data

 origin data

Acc: 0.8947

 stole data

 origin data

Acc: 0.8918

 stole data

 origin data

Acc: 0.8914

 stole data

 origin data

Cifar10：

Acc\_train: 0.98172 Acc\_test 0.7715（过拟合了，需要调参数）

 stole data

 origin data

Acc\_train: 0.98322 Acc\_test 0.7655

窃取

 origin data

彩色的也做了，有时间微调的话效果应该会更好:

stole data

origin data