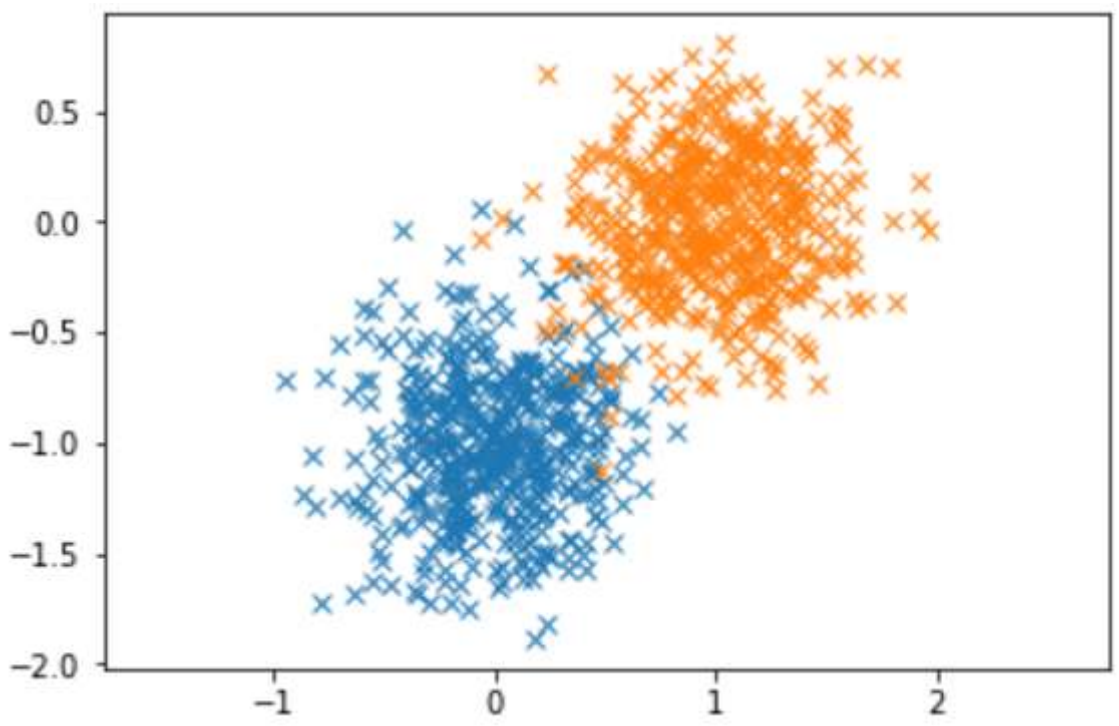
Assignment 1

Part 1

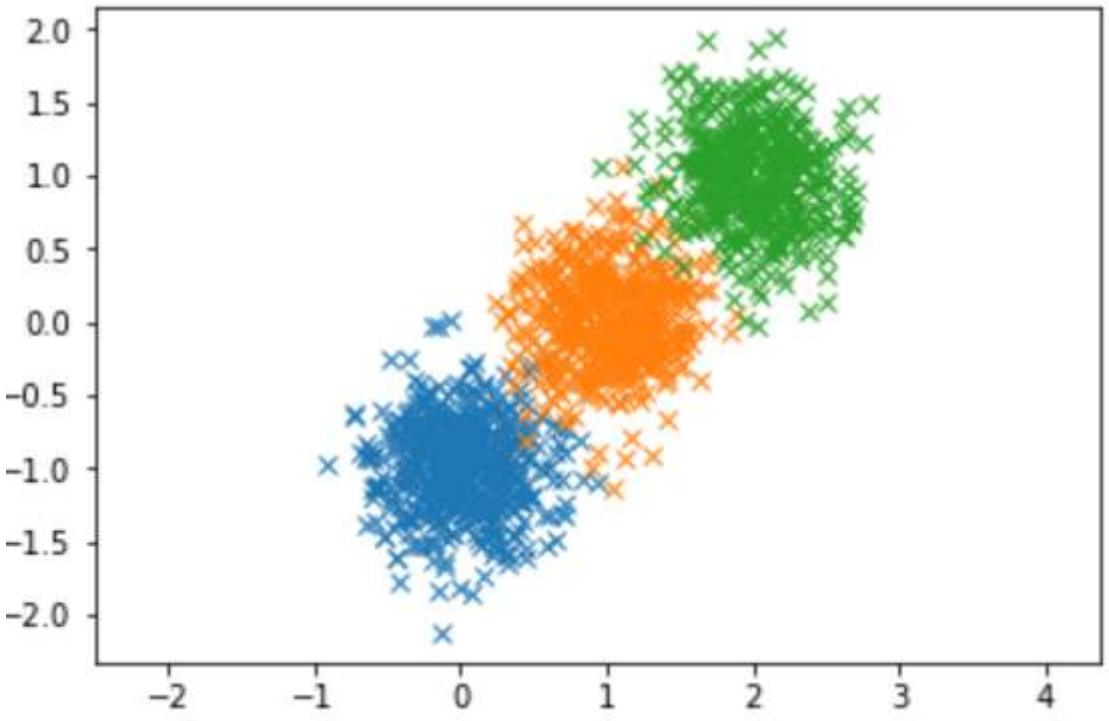
In this part, you are going to design 3 gaussian distribution, each of distribution indicates label A, label B, and label C, respectively. Then construct a dataset sampling from these distributions.

调用 $createData$ 函数生成数据，生成从多个二维高斯分布中采样形成的数据集。要生成k个二维高斯分布的数据集，需要传递k个分布的均值与协方差矩阵。k = 2 与 k = 3 时的图形如下，每个高斯分布默认采500个数据点

```
mean = [[0, -1], [1, 0]]
cov = [[[0.1,0],[0,0.1]],[[0.1,0],[0,0.1]]]
createData(mean,cov)
```




```
mean = [[0, -1], [1, 0],[2,1]]
cov = [[[0.1,0],[0,0.1]],[[0.1,0],[0,0.1]],[[0.1,0],[0,0.1]]]
createData(mean,cov)
```



Part 2

> In this part, you are required to construct 2 linear classification models: a generative model and a discriminative model. Meanwhile, you are required to compare their differences.

使用方法

```
python source.py
```

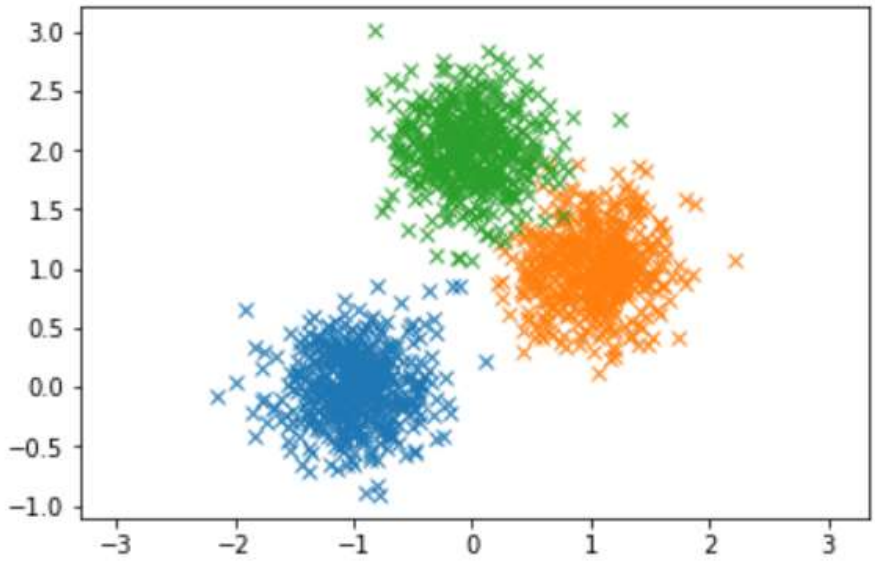
建立模型

建立与训练模型的代码如下：

```
#创建数据集
mean = [[-1, 0], [1, 1], [0, 2]]
cov = [[[0.1,0],[0,0.1]],[[0.1,0],[0,0.1]],[[0.1,0],[0,0.1]]]
createData(mean,cov)

#划分数据集
x_train,y_train,x_test,y_test = loadData(0.8) #按8:2划分训练集与测试集

#训练并测试生成模型
generative_model = GModel()
generative_model.train(x_train,y_train)
acc = generative_model.test(x_test,y_test)

#训练并测试判别模型
disc_model = DModel()
disc_model.train(x_train,y_train,x_test,y_test,mini_batch = 64,epoch =
15,alpha = 0.3)
```

在训练判别模型时，当$mini_batch$设置为1时，模型将采用SGD训练。当$mini_batch$设置为样本大小时，模型将采用BGD训练，其余情况则采用MBGD训练。

训练与测试结果




```
generative model accuracy:  0.99
discriminal model training...
Epoch    Acc
1        0.937
2        0.957
3        0.967
4        0.973
5        0.977
6        0.977
7        0.977
8        0.98
9        0.98
10       0.98
11       0.98
12       0.98
13       0.983
14       0.983
15       0.983
```

判别模型最终的混淆矩阵如下

类1	类2	类3
107	0	0
0	102	5
0	2	84

模型犯的错误集中在类2与类3混淆。从图中可以看出，类2与类3更接近，类的边缘有重合部分，使模型难以判断。

模型间的区别

- 这两个模型之间的区别在于线性判别模型是在平面中作出若干条分界线，而生成模型则通过学习输入数据，构建起一个概率分布，并通过这个概率分布预测某个点的类别。由于生成模型训练后实际上得到了一个概率分布，那么就可以从这个概率分布中获得输入的信息，并可以构造更多的数据。而判别模型则无法获得这些信息，也无法用来生成新数据。
- 判别模型需要通过梯度下降的方法最优化损失函数，需要进行多轮训练才能逐渐收敛，而生成模型则只需要在所有数据上做一遍训练即可获得模型，当样本量很大时，生成模型的训练时间要少很多。
- 一般情况下，判别模型的表现会优于生成模型，在本实验中，判别模型和生成模型都有很好的表现，这是因为数据本身满足高斯分布，和生成模型的假设是一致的。如果数据不符合高斯分布，那么生成模型的表现将会下降，但只要数据仍是线性可分的，判别模型仍能有较好的表现。
- 判别模型相对生成模型来说需要优化的参数少很多，生成模型的参数与数据的维数的平方成正比，当数据的维数上升时，生成模型的训练时间和训练效果都将快速变差。

Part 3

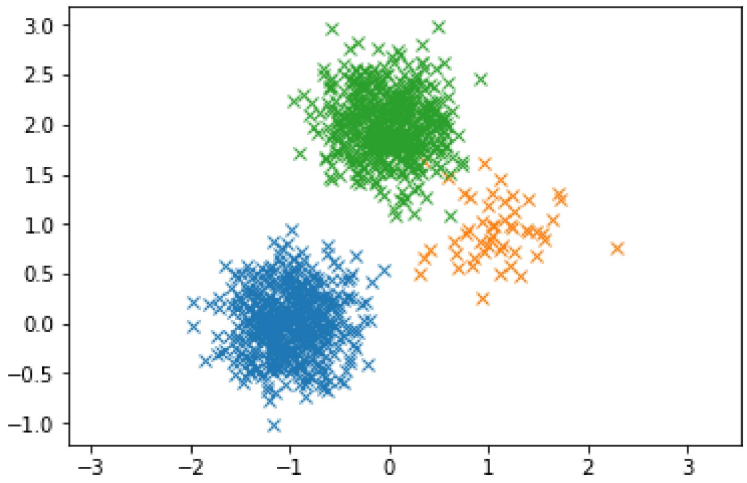
> In this part, you can reorganize the scale of your dataset or the adjust the overlap between different gaussian distributions.

样本大小

每类样本大小为500时，各模型能较好的将数据分类。提升样本大小10倍，或缩小样本到1/10，对准确率影响不大。

样本均衡

单独调整某一类的样本大小，产生不均衡样本，模型的结果如下



```
generative model accuracy:  0.99
discriminal model training...
Epoch    Acc
1        0.938
2        0.938
3        0.943
4        0.943
5        0.943
6        0.948
7        0.948
8        0.948
9        0.952
10       0.952
11       0.952
12       0.952
13       0.962
14       0.962
15       0.962
```

生成模型并不受样本不均衡的影响，而判别模型虽然准确率仍很高，但观察混淆矩阵发现，该类中很多元素被错误分类了

类1	类2	类3
95	0	1
0	3	5
0	0	106

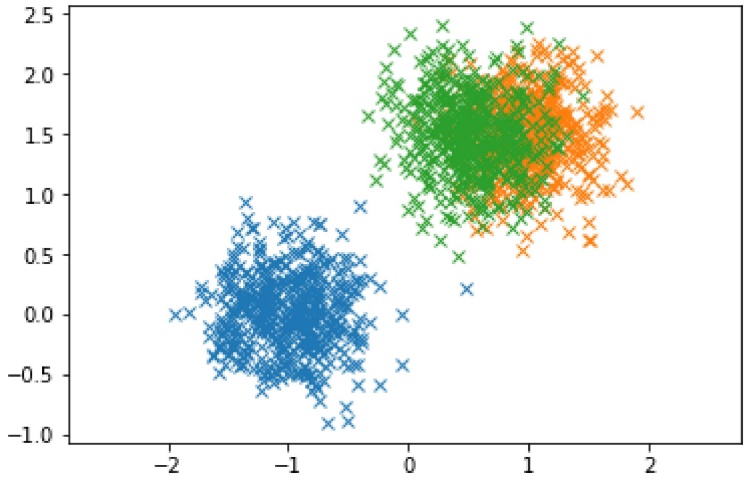
将迭代次数调至非常高，可以降低样本量很少的类被错分的概率，当迭代次数达到100次时，分类正确率可达0.995，此时的混淆矩阵如下：

类1	类2	类3
95	0	0
0	8	1
0	0	106

判别模型收敛慢的原因可能是类2的样本数少，对类2的分类错误率高对损失函数只会产生很小的影响，因此每轮迭代时产生的梯度也很小，因此需要很多轮迭代才能收敛。在最开始时，判别模型可能认为数据只有两类，类2中的点只不过是类3中的异常点而已。而生成模型则不受影响，因为生成模型直接根据类的个数建立概率分布，即它提前知道了数据就是由三类构成的，三类各对应一个高斯分布。而每一个高斯分布的参数只由对应类的数据影响，不受其他类影响，因此生成模型的表现不受影响。

样本重叠

调整高斯分布的均值，使数据重叠



此时类2和类3有大面积的重合。两个模型的准确性都有降低，生成模型的准确率降低到0.887，判别模型的准确度降到0.86（15个epoch后）。

生成模型的混淆矩阵和判别模型的混淆矩阵如下：

	生成模型			**判别模型**	
类1	类2	类3	类1	类2	类3
97	0	0	97	0	0
0	91	15	0	88	18
0	19	78	0	24	73

可以看出，两个模型的错误都集中在类2和类3的区分上，其中判别模型的表现稍差，这可能是因为在混淆的情况下，想要确定两类间的分界线将不可避免地产生很多错分，而生成模型准确获知了模型的分布为高斯分布，在数据重叠的情况下可以更少犯错。