**模式识别报告L4**

自卓2201 杨欣怡 U202215067

1，编程实现Fisher线性判别算法。

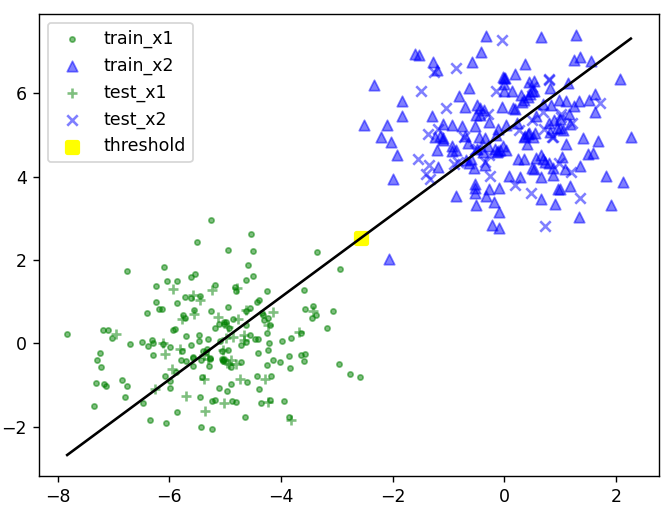
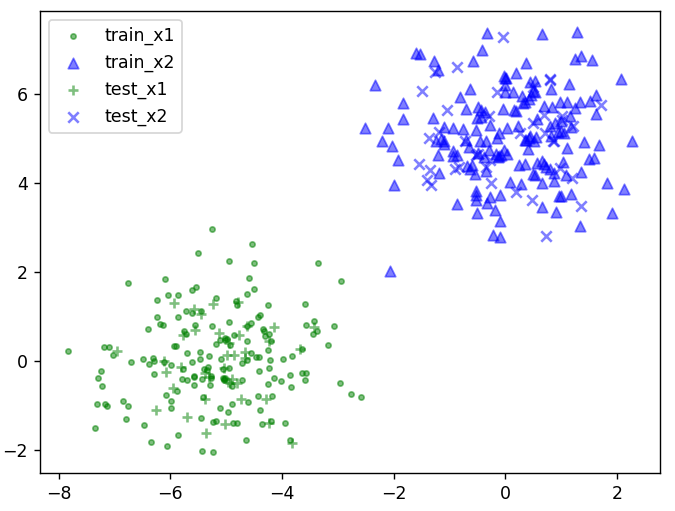
1. class Fisher:
2. def \_\_init\_\_(self):
3. self.w = np.zeros((2, 1))
4. self.thresh = 0
5. def train(self, x, y):
6. assert (x.shape[0] == y.shape[0])
7. y = y.flatten()
8. cls1\_idx = np.where(y == 1)
9. cls2\_idx = np.where(y == -1)
10. c1 = x[cls1\_idx].T
11. c2 = x[cls2\_idx].T
12. mean1 = np.mean(c1, axis=-1)
13. mean2 = np.mean(c2, axis=-1)
14. covar1 = np.cov(c1)
15. covar2 = np.cov(c2)
16. sw = covar1 + covar2
17. self.w = np.matmul(np.linalg.inv(sw), (mean1 - mean2))
18. self.thresh = np.matmul(self.w.T, mean1 + mean2) / 2
19. print('thereshold: {}'.format(self.thresh))
20. return self.w, self.thresh
21. def eval(self, x, y):
22. assert (x.shape[0] == y.shape[0])
23. y\_estimated = np.matmul(x, self.w)
24. y\_estimated = np.sign(y\_estimated - self.thresh)
25. y\_estimated = y\_estimated.reshape(-1, 1)
26. correct\_num = len(np.where((y\_estimated - y) == 0)[1])
27. accuracy = correct\_num / y.shape[0]
28. print('分类准确率: %.2f' % accuracy)

2，（a）产生两个都具有200个二维向量的数据集和。数据集的样本来自均值向量协方差矩阵的正态分布，属于“+1”类，数据集的样本来自均值向量、协方差矩阵的正态分布，属于“-1”类，其中****是一个2\*2的单位矩阵。产生的数据中80%用于训练，20%用于测试。

（b）在上述数据集上运用Fisher线性判别算法，在产生的训练样本集上得到最佳投影向量，并计算出分类阈值。

（c）在训练集和测试集上分别统计分类正确率。

（d）画出数据集、最佳投影向量和分类阈值。



在训练集和测试集上的分类准确率：



最佳投影向量：



分类阈值：

