机器学习第四次实验报告——参数估计

April 15, 2018

1 问题描述

首先生成两组数据 X 和 X'，每组数据包含 1000 个二位矢量，两组数据中的二维矢量来自于三个类别模型，三个类别模型分别满足均值矢量为 $m_1 = [1,1]^T, m_2 = [4,4]^T, m_3 = [8,1]^T$ 和协方差矩阵为 $S_1 = S_2 = S_3 = 2I$ 的正态分布，其中 I 是 2*2 的单位矩阵。在生成数据集合 X 时，每种类别的先验概率相同，在生成数据集合 X' 时，类别的先验概率由矢量 $P = [0.8, 0.1, 0.1]^T$ 给出。

2 解决方案

2.1 数据生成

首先按题目描述使用 Matlab 函数生成符合题目要求的数据集。

对于数据集合 X，由于三个类别的先验概率相同，因此首先使用函数 unidrnd(N, MM, NN) 生成范围在集合 1 2...N，大小为 MM*NN 的一维向量作为数据集 1 的 X 的类别标签。

```
label_1 = unidrnd(3,1,1000);
```

生成不同类别的数据前，首先要获得不同分类数据的索引，可以用 Matlab 的逻辑索引功能实现，具体代码为

```
index = 1:1000;
index1_1= index(label_1==1);
index1_2= index(label_1==2);
index1_3= index(label_1==3);
```

还要获得不同类别的数据数量

```
set1_1_num = sum(label_1==1);
set1_2_num = sum(label_1==2);
set1_3_num = sum(label_1==3);
```

由于已知不同类别数据的概率分布，定义不同类别数据集合的均值向量以及协方差矩阵以生成数据集

```
mean_matrix = [[1,1];[4,4];[8,1]];
cov_matrix = [1,0;0,1];
```

生成数据集合时，使用 Matlab 函数 mvnrnd($MU, SIGMA, N$) 函数用来生成均值向量为 MU，协防差矩阵为 $SIGMA$ 的 N 组数据

```
dataset1(index1_1,:)= mvnrnd(mean_matrix(1,:),cov_matrix,set1_1_num);
dataset1(index1_2,:)= mvnrnd(mean_matrix(2,:),cov_matrix,set1_2_num);
dataset1(index1_3,:)= mvnrnd(mean_matrix(3,"),cov_matrix,set1_3_num);
```

这样即生成了第一个数据集合 X。

生成数据集合 X' 时，由于类别的先验概率由矢量 $P = [0.8, 0.8, 0.1]^T$ 给出，因此使用函数 randsrc($m, n, [alphabet; prob]$) 生成大小为 $m * n$ 的矩阵，且矩阵元素及其对应的概率分别由 alphabet 和 prob 给出

```
alphabet = [1 2 3]; prob = [0.8 0.1 0.1];
label_2 = randsrc(1,1000,[alphabet; prob])
```

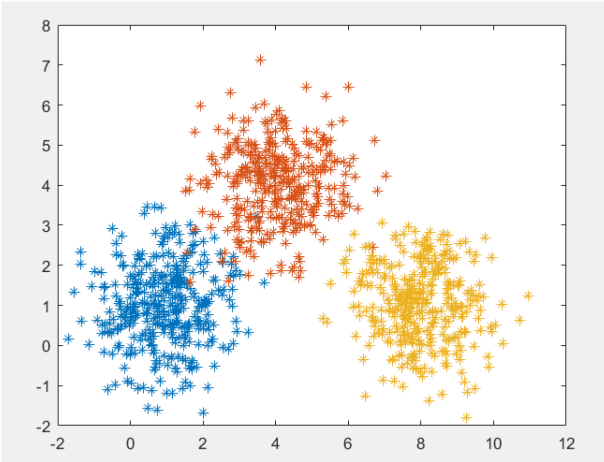
生成数据集的方法与 X 类似。

2.2 数据可视化

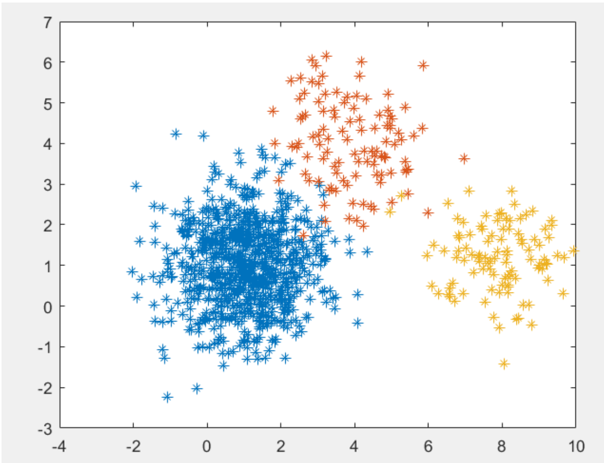
生成数据后，将两个数据集的数据画成散点图从而观察其分布规律。

```
figure;
plot(dataset1_1(:,1),dataset1_1(:,2),'*');
hold on;
plot(dataset1_2(:,1),dataset1_2(:,2),'*');
hold on;
plot(dataset1_3(:,1),dataset1_3(:,2),'*');
```

数据集 X 中的点可视化结果为



数据集 X' 中的点可视化结果为



2.3 参数估计

题目要求使用数据集合 X, 使用极大似然法估计三个模型随机矢量的均值和协方差矩阵，并解释模型参数与参数估计值之间存在的差异。

2.3.1 原理简介

极大似然估计就是在已知概率密度函数的形式，用已知的样本数据来估计未知的参数，目标是使得已知样本出现的可能性最大。

设随机样本 x_i 的概率密度为 $p(x,\theta)$, 且假设样本 x_1,x_2,x_3...x_n 之间相互独立，则对于数据集 X 联合概率为

$$p(X;\theta) = p(x_1, x_2...x_n;\theta) = \prod_{k=1}^{N} p(x_k,\theta)$$

最大似然法估计出来的参数要使得 $p(X;\theta)$ 取得最大值，即

$$\widehat{\theta_{ML}} = \arg\max_{\theta} \prod_{k=1}^{N} p(x_k,\theta)$$

求解最大值时，令似然函数对 θ 的偏导数为 0，即

$$\frac{\partial \prod_{k=1}^{N} p(x_k,\theta)}{\partial \theta} = 0$$

为计算方便，定义对数似然函数为

$$\mathrm{L}(\theta) = ln \prod_{k=1}^{N} p(x_k, \theta)$$

对数似然函数对 θ 的偏导数为 0

$$\sum_{k=1}^{N} \frac{\partial \ln(p(x_k, \theta))}{\partial \theta} = \sum_{k=1}^{N} \frac{1}{p(x_k, \theta)} * \frac{\partial p(x_k, \theta)}{\partial \theta} = 0$$

将已知样本数据带入上述等式中，得到关于 θ 的方程，解出 θ 即为用极大似然法估计出的参数。

2.3.2 编程求解

本题中，样本满足高斯分布，故

$$p(x_k, \theta) = \frac{1}{2\pi^{n/2}|\sum|^{n/2}} \exp(-\frac{1}{2}(x-\mu)^T \sum^{-1}(x-\mu))$$

其中 $\sum$ 为协方差矩阵，n 为样本空间的维度
带入梯度为 0 的等式，得到

$$\widehat{\mu} = \frac{1}{N} \sum_{k=1}^{N} x^k$$

$$\widehat{\sum} = \frac{1}{N} \sum_{k=1}^{N} (x^k - \widehat{\mu})(x^k - \widehat{\mu})^T$$

推导出上式以后，下面用 Matlab 编程求解。对于第一类样本

```
dataset1_1 = dataset1(index1_1,:);
mean1_1 = [mean(dataset1_1(:,1)),mean(dataset1_1(:,2))];
cov1_1 = cov(dataset1_1(:,1),dataset1_1(:,2));
```

对第二类样本和第三类样本处理方式类似，即参数估计的均值向量即为样本数据的均值向量，参数估计的协方差矩阵即为样本数据的协方差矩阵。

数据集 X 的三个类别对应的用极大似然法估计出的均值向量和协方差矩阵分别为

```
mean1_1 =

    1.0782    0.9543


cov1_1 =

    1.0085   -0.0130
   -0.0130    0.9619


mean1_2 =

    3.9605    3.9902


cov1_2 =

    1.0182    0.0824
    0.0824    0.9210


mean1_3 =

    8.0203    0.9950


cov1_3 =

    1.0943    0.0389
    0.0389    1.0360
```

2.3.3 结果反思

对于参数估计的结果与样本真实分布之间存在的偏差，下面分别求出对于均值向量的估计值与对于样本协方差矩阵的估计值的期望

$$\mathrm{E}(\widehat{\mu}) = \mathrm{E}(\bar{X} = \mu)$$

$$\mathrm{E}(\widehat{\sum}) = \mathrm{E}(\frac{1}{N}\sum_{k=1}^{N}(x^k - \widehat{\mu})(x^k - \widehat{\mu})^T) = \frac{n-1}{n}\sum$$

由此可见，在本题中，对于三类样本，估计的均值向量的期望分别为 $\widehat{\mu_1} = [1,1]^T, \widehat{\mu_2} = [4,4]^T, \widehat{\mu_3} = [8,1]^T$，协方差矩阵均为 $[0.999, 0; 0, 0.999]$

2.4 样本分类

2.4.1 原理简介

题目要求在两个数据集合上分别应用贝叶斯分类器和欧氏距离分类器对每一个集合的每种分类器计算分类错误率。

贝叶斯分类器主要是通过计算在已知样本条件下各个类别的后验概率从而选择出可能性最大的类别。利用贝叶斯规则，样本 x 属于类别 w_i 的概率可以表示为

$$p(w_i|x) = \frac{p(x|w_i) * p(w_i)}{p(x)} = \frac{1}{2\pi^{n/2}|\sum|^{1/2}}\exp(-\frac{1}{2}(x-\mu)^T\sum^{-1}(x-\mu)) * p(w_i) * \frac{1}{p(x)}$$

由于对于不同类别，$p(x)$ 和 $2\pi^{n/2}$ 均相同，因此有

$$p(w_i|x) \propto \frac{1}{|\sum|^{1/2}}\exp(-\frac{1}{2}(x-\mu)^T\sum^{-1}(x-\mu)) * p(w_i)$$

将上式右边取对数，得到判别式函数

$$\mathrm{g_i}(x) = -\frac{1}{2}\ln(\left|\sum\right|) - \frac{1}{2}(x-\mu)^T\sum^{-1}(x-\mu) + \ln(p(w_i))$$

将不同类别的均值向量以及协方差矩阵分别代入上述判别式函数中，即可求得后验概率最大的类别。特别地，当各个维度的方差相同且维度之间无关时，设各个维度上的方差均为 σ^2，则判别式函数可以进一步化简为

$$\mathrm{g_i}(x) = -\frac{1}{2\sigma^2}(x-\mu)^T(x-\mu) + \ln(p(w_i))$$

如果各个维度上的方差是 1，且先验概率相同，那么判别式函数可以进一步化简为

$$\mathrm{g_i}(x) = -\frac{1}{2}(x-\mu)^T(x-\mu)$$

取此函数的最大值即为取样本到各个类别均值向量的欧氏距离的最小值，由此可见，当多维正态分布的随机样本各个属性相互独立并且有单位方差，即协方差矩阵为单位矩阵，同时满足各个类别的先验概率相同时，贝叶斯分类器与欧式距离分类器的结果相同。

2.4.2 具体实现

下面用 Matlab 编程实现欧氏贝叶斯分类器。定义函数 bayesian($dataset, prior, mean$)，其中 $dataset$ 为待分类的数据集合，$prior$ 为各个类别的先验概率，$mean$ 为各类别的均值向量，函数返回计算的每个数据集预测的类别标签。

```
function pred = bayesian(dataset,prior,mean)
dataset_size = size(dataset);
prior_size = size(prior);
data_num = dataset_size(1,1);
class_num = prior_size(1,1);
pred = ones(data_num,1);
for i = 1:data_num
    % calculate the posterior probability
    post_prob = zeros(class_num,1);
    %note that the variance across different classes is the same
    for class = 1:1:class_num
        post_prob(class) = (-0.5)*(dataset(i)-mean(class))*(dataset(i)-mean(class))'+log(prior(class));
    end
    [val,index]= max(post_prob);
    pred(i) = index;
end
```

对于每一个数据样本，函数先计算不同类别的后验概率，$\max(post\_prob)$ 返回后验概率的最大值作为贝叶斯分类器的结果。

对于数据集 X 和数据集 X', 分别调用函数计算预测结果并计算准确率。

```
prior_prob_1 = [1/3;1/3;1/3];
prior_prob_2 = [0.8;0.1;0.1];
bayesian_pred1 = bayesian(dataset1,prior_prob_1,mean_matrix);
bayesian_accu1=mean(bayesian_pred1'==label_1);
bayesian_pred2 = bayesian(dataset2,prior_prob_2,mean_matrix);
bayesian_accu2=mean(bayesian_pred2'==label_2);
```

贝叶斯分类器在 X 和 X' 计算的准确率分别为

```
bayesian_accu1 =

    0.9400
```

```
bayesian_accu2 =

    0.9590
```

再定义函数 min_distance$(dataset, mean)$，其中 $dataset$ 为待分类的数据集合，$mean$ 为各类别的均值向量, 函数返回计算的每个数据集预测的类别标签。

```
function pred = min_distance(dataset,mean)
dataset_size = size(dataset);
mean_size = size(mean);
data_num = dataset_size(1,1);
class_num = mean_size(1,1);
pred = ones(data_num,1);
for i = 1:data_num
    prob = ones(class_num,1);
    for j=1:class_num
        prob(j) = (1/2)*(dataset(i)-mean(j))*(dataset(i)-mean(j))'
    end
    [val,index]= min(prob);
    pred(i) = index;
end
```

对于每一个数据样本，函数计算不同类别的均值向量距离样本点的距离，选择距离最小的均值向量所在类别作为预测结果。

对于数据集 X 和数据集 X', 分别调用函数计算预测结果并计算准确率。

```
min_pred1 = min_distance(dataset1,mean_matrix);
min_accu1=mean(min_pred1'==label_1);
min_pred2 = min_distance(dataset2,mean_matrix);
min_accu2=mean(min_pred2'==label_2);
```

欧氏距离分类器在 X 和 X' 计算的准确率分别为

```
min_accu1 =

    0.9400
```

```
min_accu2 =

    0.9300
```

2.4.3 结果反思

由上述结果可以看出，对于数据集 X, 由于各个类别的先验概率相同，用贝叶斯分类器和最小距离分类器的结果相同，而对于数据集 X', 由于各个类别的先验概率不同，因此应用欧式距离分类器时，其准确率相比于贝叶斯分类器有所下降。可见样本的先验概率会对样本的分类产生影响。

3 实验总结

在本次实验中，使用 Matlab 软件，首先生成了两组数据，均来自三个不同类别，第一组数据三个类别的先验概率相同，而第二组数据三个类别的先验概率由 $[0.8, 0.1, 0.1]^T$ 给出。然后进行数据可视化观察数据的分布规律，再用极大似然法估计高斯分布的参数，包括均值向量以及协方差矩阵，与样本的真实参数对比。极大似然法对均值向量的估计是无偏估计，估计的期望与原期望相同；对协方差矩阵的估计是有偏估计，估计的协方差矩阵期望与实际不同。最后分别用贝叶斯分类器与最小距离分类器对训练集数据进行分类并计算准确率。对于贝叶斯分类器，当样本各属性之间相互独立且具有单位方差时，若各个类别的先验概率相同，可以简化为最小距离分类器。因此对于各个类别先验概率相同的数据集 X, 其用最小距离分类器与用贝叶斯分类器的分类结果相同，而对于各个类别先验概率不同的 X'，其用最小距离分类器的分类结果准确率则低于贝叶斯分类器。