一. 实验环境

PyCharm-162.1237.1

二. 实验数据

样本	W1			W2		
	X1	X2	X3	X1	X2	X3
1	0.42	-0.087	0.58	-0.4	0.58	0.089
2	-0.2	-3.3	-3.4	-0.31	0.27	-0.04
3	1.3	-0.32	1.7	0.38	0.055	-0.035
4	0.39	0.71	0.23	-0.15	0.53	0.011
5	-1.6	-5.3	-0.15	-0.35	0.47	0.034
6	-0.029	0.89	-4.7	0.17	0.69	0.1
7	-0.23	1.9	2.2	-0.011	0.55	-0.18
8	0.27	-0.3	-0.87	-0.27	0.61	0.12
9	-1.9	0.76	-2.1	-0.065	0.49	0.0012
10	0.87	-1.0	-2.6	-0.12	0.054	-0.063

三. 实验题目

<1>.考虑不同维数下的高斯概率密度模型。

a. 编写程序，对表格中的类w_1中的 3 个特征x_i,分别求解最大似然估计$\hat{\mu}$和$\hat{\sigma}^2$。

b. 修改程序，处理二维数据的情形$p(x) \sim N(\mu, \Sigma)$。然后处理对表格中的类w_1中的任意两个特征的组合。

c. 修改程序，处理三维数据的情形$p(x) \sim N(\mu, \Sigma)$。然后处理对表格中的类w_1中的三个特征的组合。

d. 假设这个三位高斯模型是可分离的，即$\Sigma = \mathrm{diag}(\sigma_1^2, \sigma_2^2, \sigma_3^2)$，写一个程序估计类别$w_2$中的均值和协方差矩阵中的 3 个参数。

e. 比较前 4 种方式计算出来的每一个特征的均值μ_i的异同。并加以解释

f. 比较前 4 种方式计算出来的每一个特征的均值σ_i^2的异同。并加以解释

〈2〉. 考虑一个具有两个参数的一维三角形概率模型

$$p(x|\theta) \equiv T(\mu, \delta) = \begin{cases} \dfrac{\delta - |x-\mu|}{\delta^2}, |x-\mu| < \delta \\ 0, 其他 \end{cases}$$

其中$\theta = \binom{\mu}{\delta}$。编写特征，对类别$w_2$的特征$x_2$使用贝叶斯方法估计概率面密度$p(x|D)$。并且画出后验概率$p(x|D)$。

四． 问题分析

a) 对于问题（1），采用最大似然估计的结论进行计算。由μ, Σ均未知的情况下推导出的估计值$\hat{\mu} = \frac{1}{n}\sum_{k=1}^{n} x_k$，$\hat{\sigma}^2 = \frac{1}{n}\sum_{k=1}^{n}(x_k - \hat{\mu})(x_k - \hat{\mu})^T$。即可计算出对应的$\mu_i$以及协方差矩阵$\sigma_i^2$。

b) 对于问题(2)，采用贝叶斯估计的方式估计后验概率密度 p(x|D)。使用的是贝叶斯参数估计中的一般理论，所用到的方程如下:

$$\begin{cases} p(x|D) = \int p(x|\theta)p(\theta|D)d\theta \\ p(\theta|D) = \dfrac{p(D|\theta)p(\theta)}{\int p(D|\theta)p(\theta)d\theta} \\ p(D|\theta) = \prod_{k=1}^{n} p(x_k|\theta) \end{cases}$$

而其中$p(x|\theta)$题目中已经有给出：

$$p(x|\theta) \equiv T(\mu, \delta) = \begin{cases} \dfrac{\delta - |x-\mu|}{\delta^2}, |x-\mu| < \delta \\ 0, 其他 \end{cases}$$

接下来是对于$p(\theta)$的计算，而$p(\theta)$是我们自己的估计值, 不妨设:

$$p(\theta) = p(\delta)p(\mu)$$

我们此时不妨假设变量μ与变量δ分别符合参数为 0, 1 的正态分布，即：

$$(\mu, \delta) \sim N(0,1)$$

而参数为(0, 1)的正态分布的公式如下:

$$p(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$$

然后分别带入公式进行计算，即可得到结果。

五. 计算结果

第一题:

a) 对于第一题第一问的结果如下：

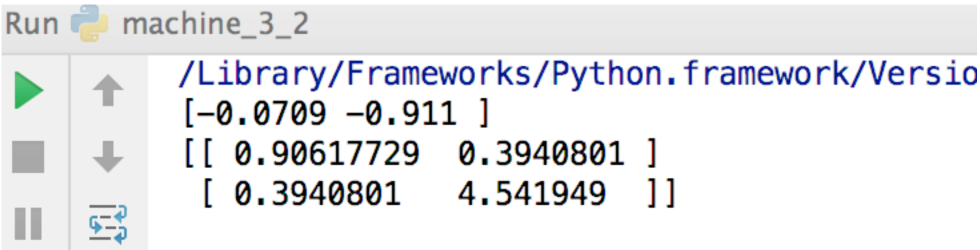
```
Run machine_3_2
/Library/Frameworks/Python.framework/Ver
[-0.0709]
[[ 0.90617729]]
```

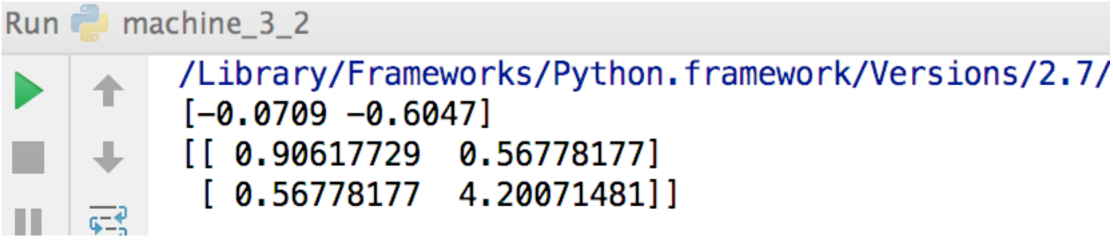
```
Run machine_3_2
/Library/Frameworks/Python.frame
[-0.6047]
[[ 4.20071481]]
```

```
Run machine_3_2
/Library/Frameworks/Python.framework/
[-0.911]
[[ 4.541949]]
```

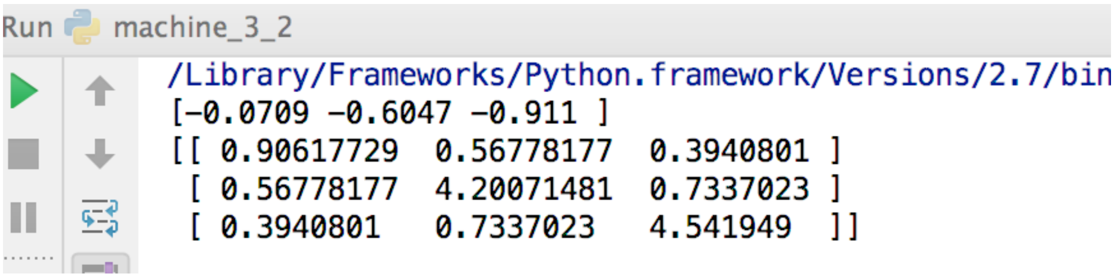
b) 对于第一题第二问的结果如下(依次是第一列与第二列,第二列与第三列以及第一列与第三列):

```
Run machine_3_2
/Library/Frameworks/Python.framework,
[-0.0709 -0.6047]
[[ 0.90617729   0.56778177]
 [ 0.56778177   4.20071481]]
```



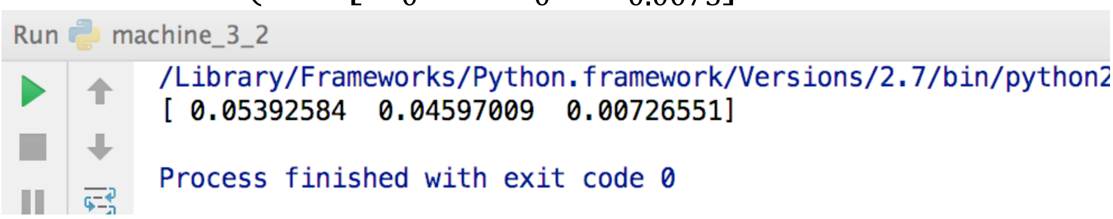


c） 对于第一题第三问的结果如下：



d） 对于第一题第四问的结果如下：

$$\begin{cases} \mu_1 = [-0.1126, -0.4299, -0.0037] \\ \sigma_1^2 = \begin{bmatrix} 0.0539 & 0 & 0 \\ 0 & 0.0460 & 0 \\ 0 & 0 & 0.0073 \end{bmatrix} \end{cases}$$



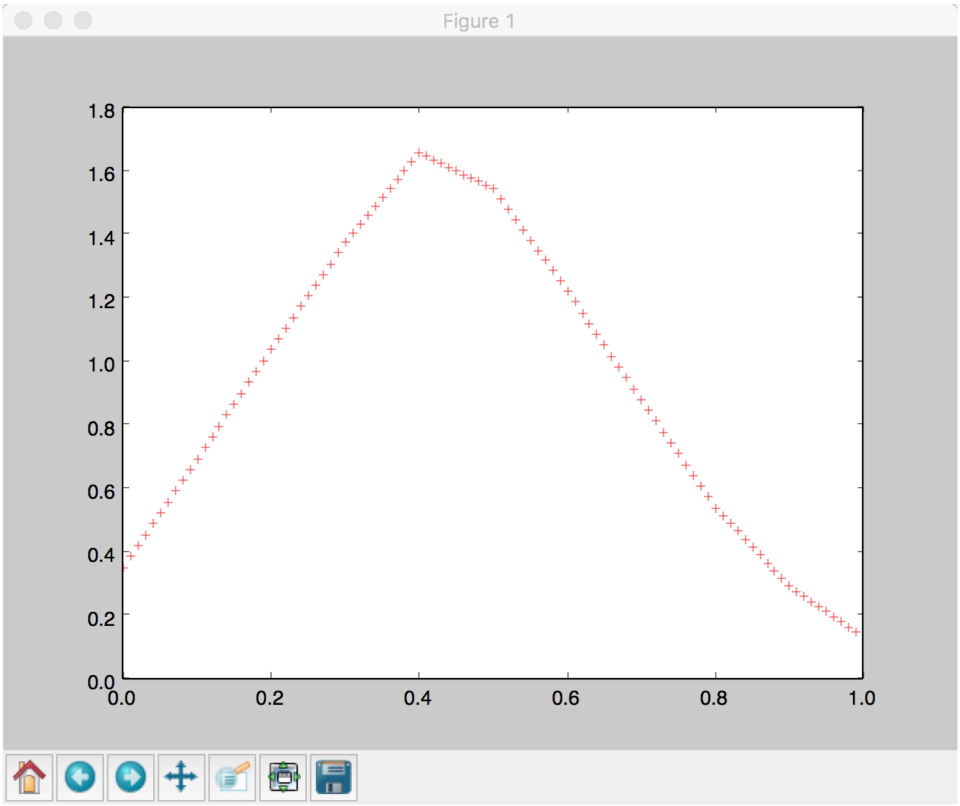
e） 对于上面的结果进行对比发现，第二问第三问其实就是对于第一问结果的排列组合，而第三问的结果实际上也是对于第二问的一种扩展排列。其原因就是在于我们选择的计算公式实际上就是对其单个维度的属性值进行相加求平均值。所以实际上对于每一个维度上的μ_i都相同。

f） 而对于方差来说实际上第二问，第三问包括第四问的对角线上的元素实

际上就是第一问对应的方差。而第三问的协方差矩阵和第二问的协方差矩阵中除了对角线上的元素外，其他的元素也是有相似的。其原因还是在于我们的矩阵运算的方法是相同的，所以最后得到的结果一定也是相同的。

第二题：

a）而第二题我们最终得到的结果如图：



其中纵坐标表示的是我们所求出来的 p(x|D)，而横坐标就是我们的 x 轴。

六. 程序代码

见附件。

七. 收获与感悟

a）经过这次实验首先我更加熟悉了 python 的使用

b） 我对最大似然估计的算法更加的清晰

c） 对于最大似然估计与贝叶斯估计的区别有了一些自己的概念。