深圳大学研究生课程：模式识别理论与方法

课程作业实验报告

实验名称：主分量分析 PCA

实验编号：Proj03-01

签　　名：

姓　　名：夏荣杰

学　　号：2170269107

截止提交日期：2018 年 4 月 20 日

摘要：主分量分析(principal component analysis,PCA)是一种将高维数据投影到低维数据的线性变换方法,这一方法的目的是寻找在最小均方意义下最能代表原始数据特征的投影方向，用这些方向矢量表示数据。本次实验分为两部分，第一部分是利用 PCA 进行特征空间的规整化；第二部分是利用 PCA 进行特征空间降维。通过实验，了解 PCA 主分量分析方法的基本概念，学习和掌握 PCA 主分量分析方法的基本概念，学习和掌握 PCA 主分量分析方法。实验结果表明，利用 PCA 方法可以对数据集合在特征空间中进行平移和旋转，从而进行规整化；经过 PCA 降维得到的集合是原始样本集合在某方向上的垂直投影，该方向是由原始样本集合在其散布矩阵的最大特征值与次大特征值所对应的特征向量构成的基向量形成的平面；此外，样本的数量 N 越大，估计得到的样本均值向量越接近给定的均值矢量，散布矩阵也越接近给定的协方差矩阵的(N-1)倍，信息损失越小。

一、 背景技术 或 基本原理

1. 样本集合的均值向量和散布矩阵

假设有 n 个 d 维的样本 $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n$，则该样本集合的均值向量为

$$\mathbf{m} = \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_k \tag{1}$$

散布矩阵定义为

$$\mathbf{S} = \sum_{k=1}^{n}(\mathbf{x}_k - \mathbf{m})(\mathbf{x}_k - \mathbf{m})^t \tag{2}$$

事实上，它就是样本协方差矩阵的 n-1 倍。在 MATLAB 中可利用自带求协方差矩阵函数 cov(·)乘以(n-1)得到。

2. 矩阵的特征值和特征向量

定义：设 $\mathbf{A} = (a_{ij})$ 是数域 C 上的 n 矩阵，λ 是参数，**A** 的特征矩阵 $\lambda\mathbf{I} - \mathbf{A}$ 的行列式为：

$$\det(\lambda\mathbf{I} - \mathbf{A}) = \begin{vmatrix} \lambda - a_{11} & -a_{12} & \cdots & -a_{1n} \\ -a_{21} & \lambda - a_{22} & \cdots & -a_{2n} \\ \vdots & \vdots & & \vdots \\ -a_{n1} & -a_{n2} & \cdots & \lambda - a_{nn} \end{vmatrix} \tag{3}$$

称为 **A** 的特征多项式，记为 $\varphi(\lambda)$。$\varphi(\lambda) = 0$ 的根 λ 称为 **A** 的特征值（特征根）。

而相应于方程组：

$$(\lambda\mathbf{I} - \mathbf{A})\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix} = 0 \tag{4}$$

的非零解 $[\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_n]^T$ 称为 **A** 的属于特征值 λ 的特征向量。

在 MATLAB 中，计算矩阵的特征值和特征向量可以利用 MATLAB 函数 eig(·)来实现。[**V**, **D**] = eig(·)返回 **V** 和 **D** 两个矩阵，其中，**D** 是一个对角阵，对角线上的元素即特征值，**V** 是由特征向量构成的特征矩阵，**V** 中的每一列即特征值对应的特征向量。求出 **D** 矩阵后，可以使用 sort(·)函数对特征值进行从大到小排序。

3. 数据的规整化

① **去中心化：将坐标原点移至样本中心：**

$$\mathbf{x}'_k = \mathbf{x}_k - \mathbf{m} \tag{5}$$

其中，**m** 为样本均值向量。

② **基变换**：n 维线性空间 V 的两组基 $(\varepsilon_1,\varepsilon_2,...,\varepsilon_n)$ 和 $(\varepsilon_1',\varepsilon_2',...,\varepsilon_n')$ 的变换如下：

$$(\varepsilon_1',\varepsilon_2',...,\varepsilon_n')=\mathbf{A}(\varepsilon_1,\varepsilon_2,...,\varepsilon_n) \tag{6}$$

其中，**A** 为过渡矩阵。

4. 主分量分析 PCA 的原理

PCA 方法是一种线性变换的方法，目的是寻找在最小均方意义下最能够代表原始数据的投影方法。

假设有 n 个 d 维的样本 $\mathbf{x}_1,\mathbf{x}_2,\cdots,\mathbf{x}_n$，将样本集合的全部样本向通过均值矢量的一条直线作投影，可以得到样本空间中表示样本集合的一条直线，称之为样本集合的 1 维表达，该直线可以表示为：

$$\mathbf{x}=\mathbf{m}+a\mathbf{e} \tag{7}$$

其中，**e** 表示通过均值的直线方向上的单位矢量；a 是一个实数，表示直线上的点 **x** 距离均值点 **m** 的距离。

所以，如何求得该样本集合的 1 维表达直线？

假设使用该直线上的一个点 $\mathbf{m}+a_k\mathbf{e}$ 来表示样本点 $\mathbf{x}_k$，则可以构造样本集合的平方误差准则函数：

$$J_1(a_1,\cdots,a_n,\mathbf{e})=\sum_{k=1}^{n}\left\|\left(\mathbf{m}+a_k\mathbf{e}\right)-\mathbf{x}_k\right\|^2 \tag{8}$$

为使得 J1 最小，解得

$$\mathbf{a}_k=\mathbf{e}_t\left(\mathbf{x}_k-\mathbf{m}\right) \tag{9}$$

即把向量 $\mathbf{x}_k$ 向样本均值的直线 **e** 作垂直投影就能够得到最小方差结果。

接下来考虑，寻找直线的最优方向 **e** ，使得平方误差准则函数 J1 值最小。

将 $\mathbf{a}_k=\mathbf{e}_t\left(\mathbf{x}_k-\mathbf{m}\right)$ 带入 $J_1(a_1,\cdots,a_n,\mathbf{e})$ 公式，求得

$$J_1(\mathbf{e})=-\mathbf{e}^t\mathbf{S}\mathbf{e}+\sum_{k=1}^{n}\left\|\mathbf{x}_k-\mathbf{m}\right\|^2 \tag{10}$$

其中，**S** 为样本的散布矩阵。为使得 J1 最小，解得

$$\mathbf{S}\mathbf{e}=\lambda\mathbf{e} \tag{11}$$

可得结论，直线的最优方向 **e** 是散布矩阵 **S** 对应于最大特征值 λ 的特征矢量方向。

推广至高维空间，使用 d' 维的特征矢量来近似原空间中的 d 维矢量，其中 $d'\le d$ 。

$$\mathbf{x}=\mathbf{m}+\sum_{d}^{d'}a_i\mathbf{e}_i \tag{12}$$

其中，**m** 为样本均值矢量，$\mathbf{e}_1,\mathbf{e}_2,...,\mathbf{e}_{d'}$ 为散布矩阵 **S** 的对应 d' 个最大特征值所对应的特征向量。a_i 是矢量 **x** 对应于基 $\mathbf{e}_i$ 的系数，称为主分量(Principal Component)。

二、 实验方法 或 算法流程步骤

实验用到的方法和步骤如下：

1. 利用 PCA 进行特征空间的规整化

① 根据实验给出的均值矢量 $\boldsymbol{\mu}$ 和协方差矩阵 $\boldsymbol{\Sigma}$ 生成二维样本矢量并画出散点图；

② 根据公式（1）和（2）分别计算样本集合的均值矢量 m 和散布矩阵 S，并使用 MATLAB 函数 eig(·)计算散布矩阵的特征值 D 和特征向量 V；

③ 根据公式（5）和（6），利用散布矩阵的特征向量 V 矩阵对样本集合进行去中心化和基变换并画出散点图；

④ 改变样本数量 10,100,1000,10000,100000，重复进行上述实验步骤。

2. 利用 PCA 进行特征空间降维

① 根据实验给出的均值矢量 $\boldsymbol{\mu}$ 和协方差矩阵 $\boldsymbol{\Sigma}$ 生成二维样本矢量并画出散点图；

② 根据公式（1）和（2）分别计算样本集合的均值矢量 m 和散布矩阵 S，并使用 MATLAB 函数 eig(·)计算散布矩阵的特征值 D 和特征向量 V，将特征值进行从大到小的排序；

③ 根据公式（5）和（6），利用最大特征值和次大特征值对应的特征向量对样本集合 X 进行降维，生成二维向量集合 Y；

④ 使用 MATLAB 函数 inv(·)计算特征向量矩阵 V 的逆矩阵 $\mathbf{V}^{-1}$，取出 $\mathbf{V}^{-1}$ 的前两列组成一个新的矩阵 W；

⑤ 根据公式（10），利用降维后的二维向量 Y 重新生成新的三维向量 Z；

⑥ 计算集合 Z 和集合 X 之间的误差的平方和均方误差；

⑦ 改变样本数量 10,20,50,100,1000，重复进行上述实验步骤。

三、 实验结果

1. 实验(a)利用 PCA 进行特征空间的规整化结果如下：

给定均值矢量和协方差矩阵如下：

$$\boldsymbol{\mu}=\begin{pmatrix}5\\7\end{pmatrix},\quad \boldsymbol{\Sigma}=\begin{pmatrix}9 & 2.4\\2.4 & 1\end{pmatrix}$$

生成 N 个高斯分布的二维样本矢量 **X**，利用 PCA 将集合 **X** 中的每个向量 **x** 变换为向量 **y**，生成集合 **Y**，当 N=10，100，1000，10000，100000 时，集合 **X** 和集合 **Y** 的散点图如图 1-1 至图 1-5 所示。

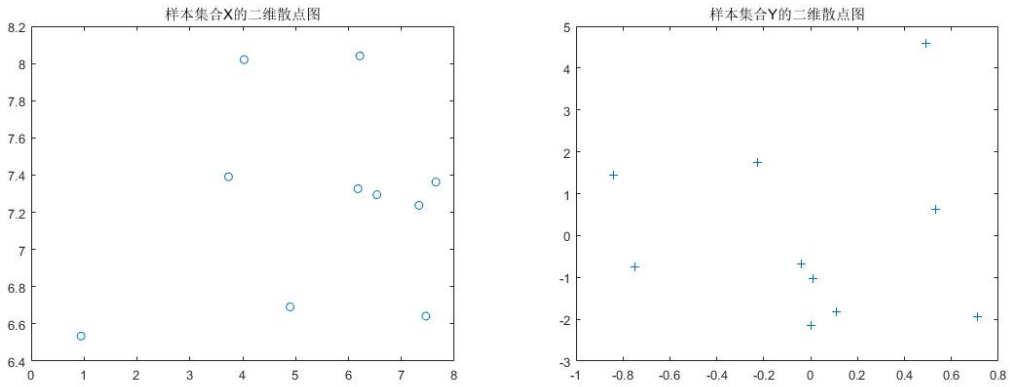

图 1-1. N=10 时样本集合 X 和集合 Y 的散点图

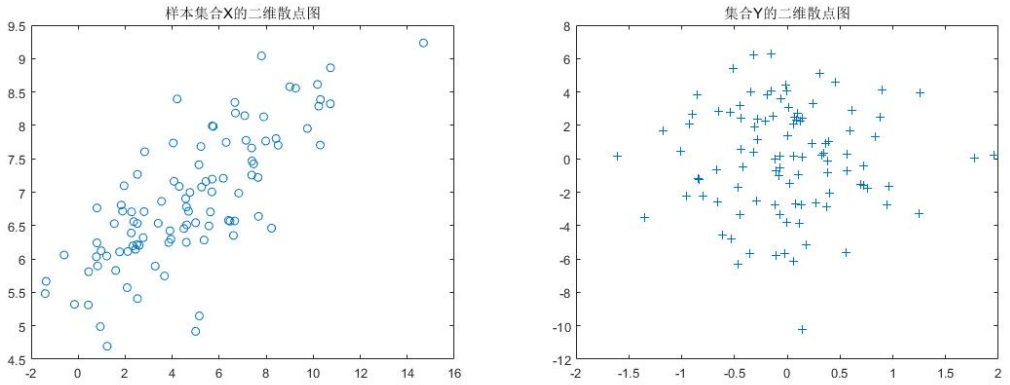

图 1-2. N=100 时样本集合 X 和集合 Y 的散点图

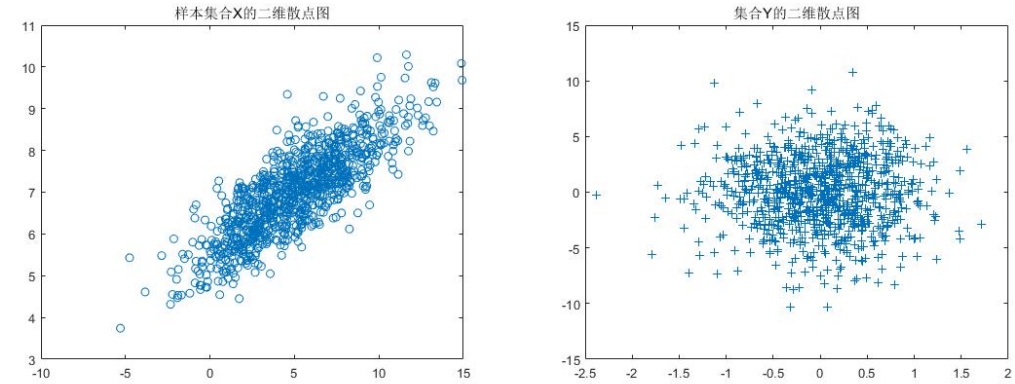

图 1-3. N=1000 时样本集合 X 和集合 Y 的散点图

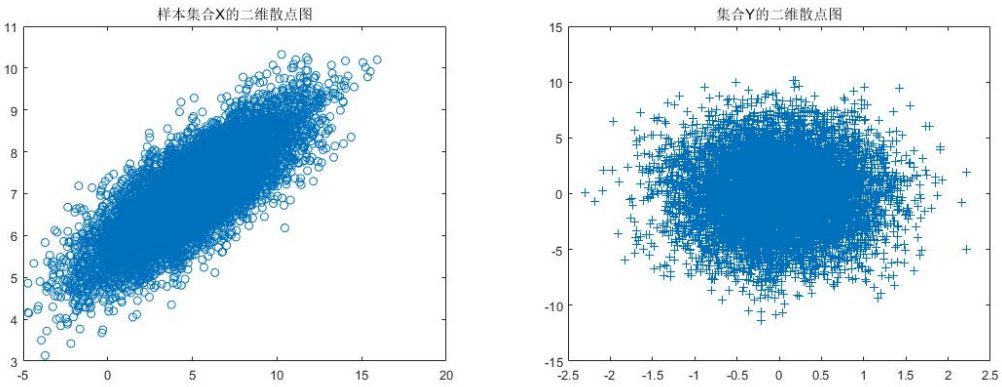

图 1-4. N=10000 时样本集合 X 和集合 Y 的散点图

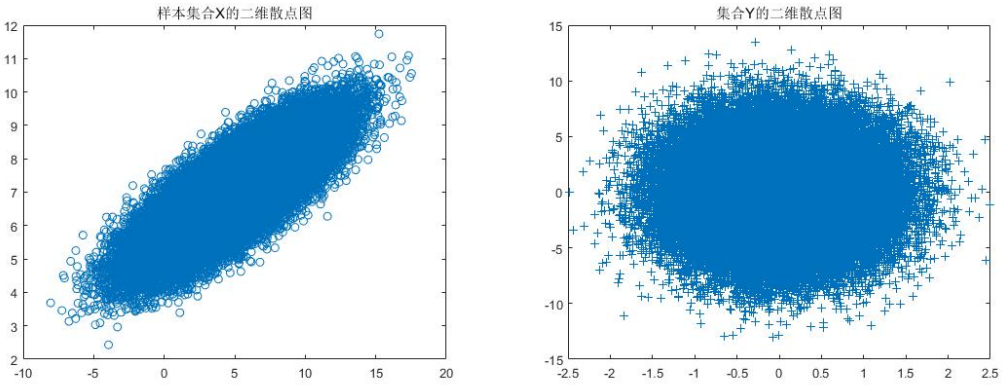

图 1-5. N=100000 时样本集合 X 和集合 Y 的散点图

实验中，已经给定均值矢量 $\boldsymbol{\mu}=\begin{pmatrix}5\\7\end{pmatrix}$，协方差矩阵 $\boldsymbol{\Sigma}=\begin{pmatrix}9 & 2.4\\2.4 & 1\end{pmatrix}$，而由此生成的样本，在不同的样本数量 N 的情况下，我们重新计算样本均值向量和协方差矩阵，对比如表 1 所示：

表 1. 不同样本数量 N 的情况下样本均值矢量和协方差矩阵

	均值矢量	协方差矩阵
给定值	$[5, 7]$	$\begin{bmatrix}9, 2.4\\2.4, 1\end{bmatrix}$
N = 10	$[5.4968, 7.2540]$	$\begin{bmatrix}4.4825, 0.2111\\0.2111, 0.2726\end{bmatrix}$
N = 100	$[4.7438, 6.8758]$	$\begin{bmatrix}9.5014, 2.3019\\2.3019, 0.9528\end{bmatrix}$
N = 1000	$[4.9921, 6.9863]$	$\begin{bmatrix}9.4531, 2.5574\\2.5574, 1.0375\end{bmatrix}$
N = 10000	$[5.0388, 7.0009]$	$\begin{bmatrix}8.8926, 2.3469\\2.3469, 0.9752\end{bmatrix}$
N = 100000	$[5.0059, 7.0025]$	$\begin{bmatrix}8.9761, 2.3916\\2.3916, 0.9973\end{bmatrix}$

2. 实验(b)利用 PCA 进行特征空间降维结果如下：

给定均值矢量和协方差矩阵如下：

$$\boldsymbol{\mu}=\begin{pmatrix}10\\15\\15\end{pmatrix},\quad \boldsymbol{\Sigma}=\begin{pmatrix}90 & 2.5 & 1.2\\2.5 & 35 & 0.2\\1.2 & 0.2 & 0.02\end{pmatrix}$$

生成 N 个高斯分布的三维样本矢量 **X**，利用 PCA 将集合 **X** 中的每个向量 **x** 变换为向量 **y**，生成集合 **Y**，利用降维后的二维向量 **Y** 重新生成新的三维向量 **Z**，在一幅图中用不同的颜色分别绘出集合 **Z** 和集合 **X** 的三维数据散点图。当 N=10，20，50，100，1000 时，集合 **X**、集合 **Y** 的散点图，集合 **Z** 和集合 **X** 的三维数据散点图如图 2-1 至图 2-5 所示。

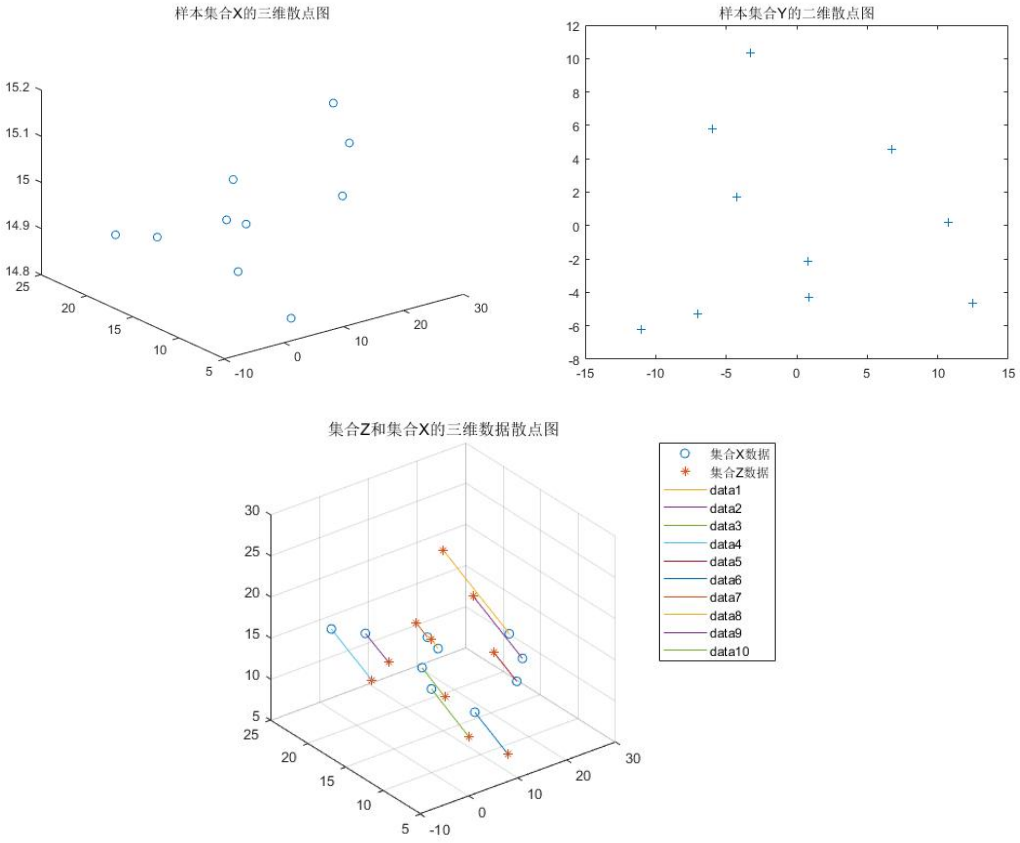

图 2-1. N=10 时集合 X、集合 Y 的散点图，集合 Z 和集合 X 的三维数据散点图

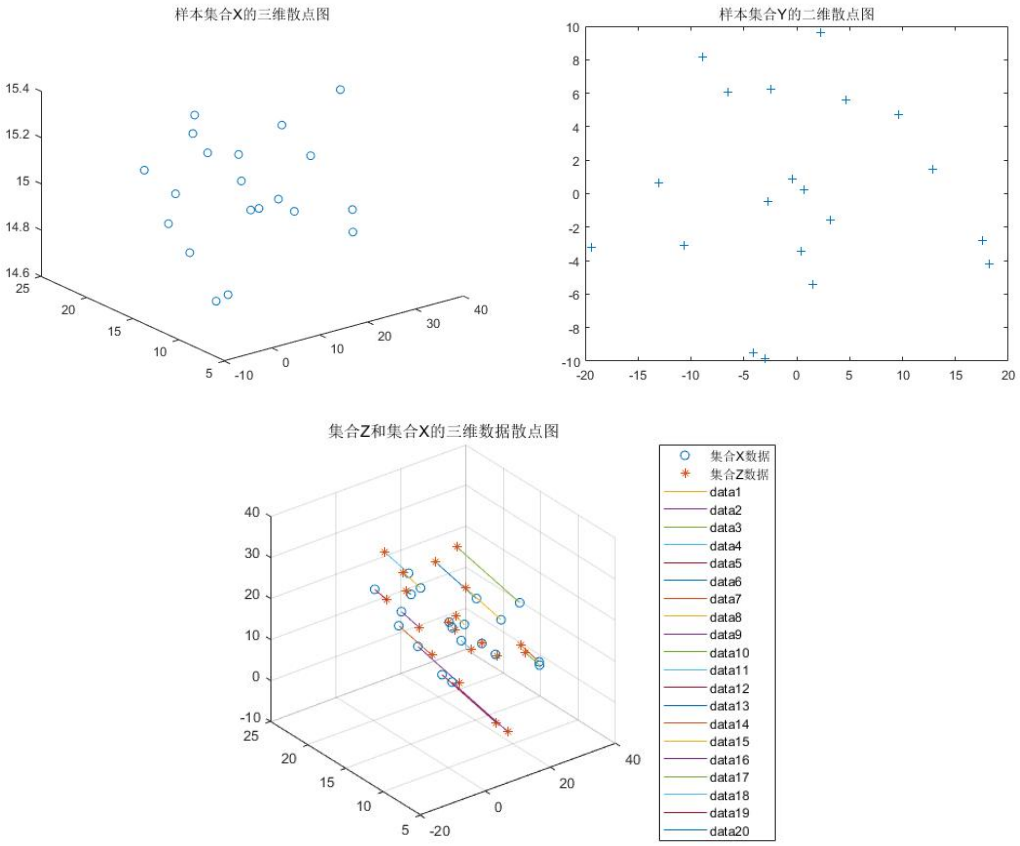

图 2-2. N=20 时集合 X、集合 Y 的散点图，集合 Z 和集合 X 的三维数据散点图

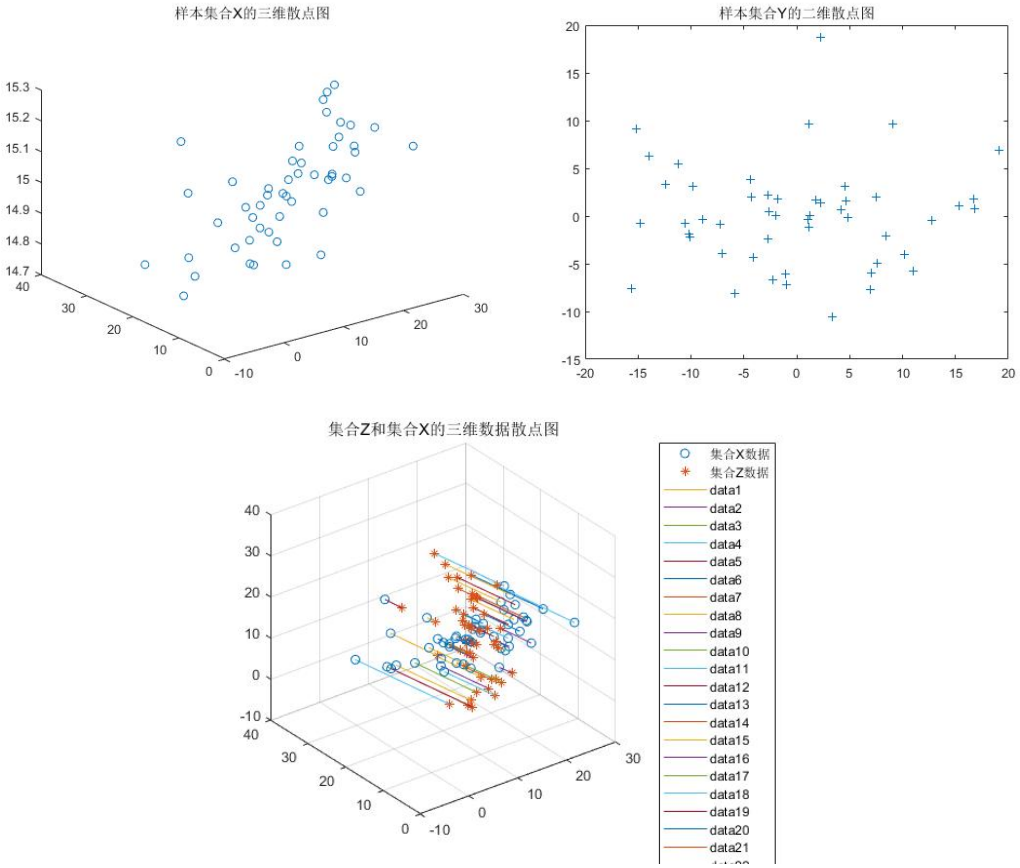

图 2-3. N=50 时集合 X、集合 Y 的散点图，集合 Z 和集合 X 的三维数据散点图

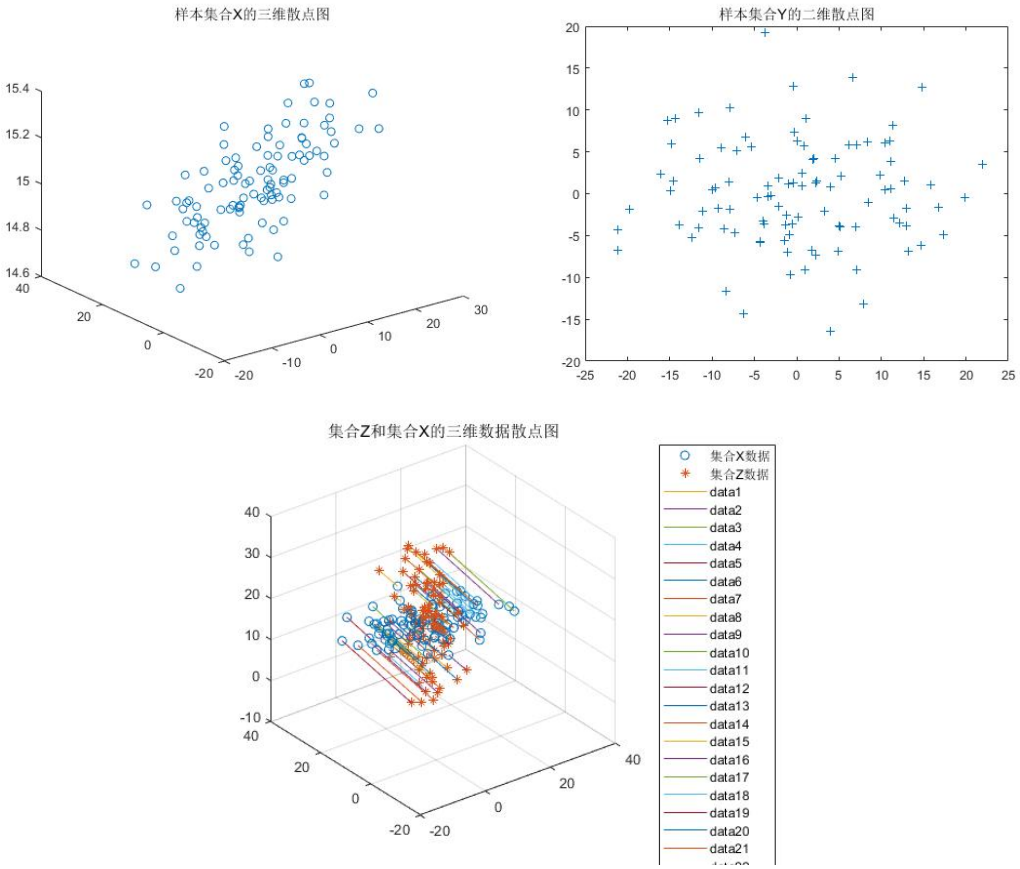

图 2-4. N=100 时集合 X、集合 Y 的散点图，集合 Z 和集合 X 的三维数据散点图

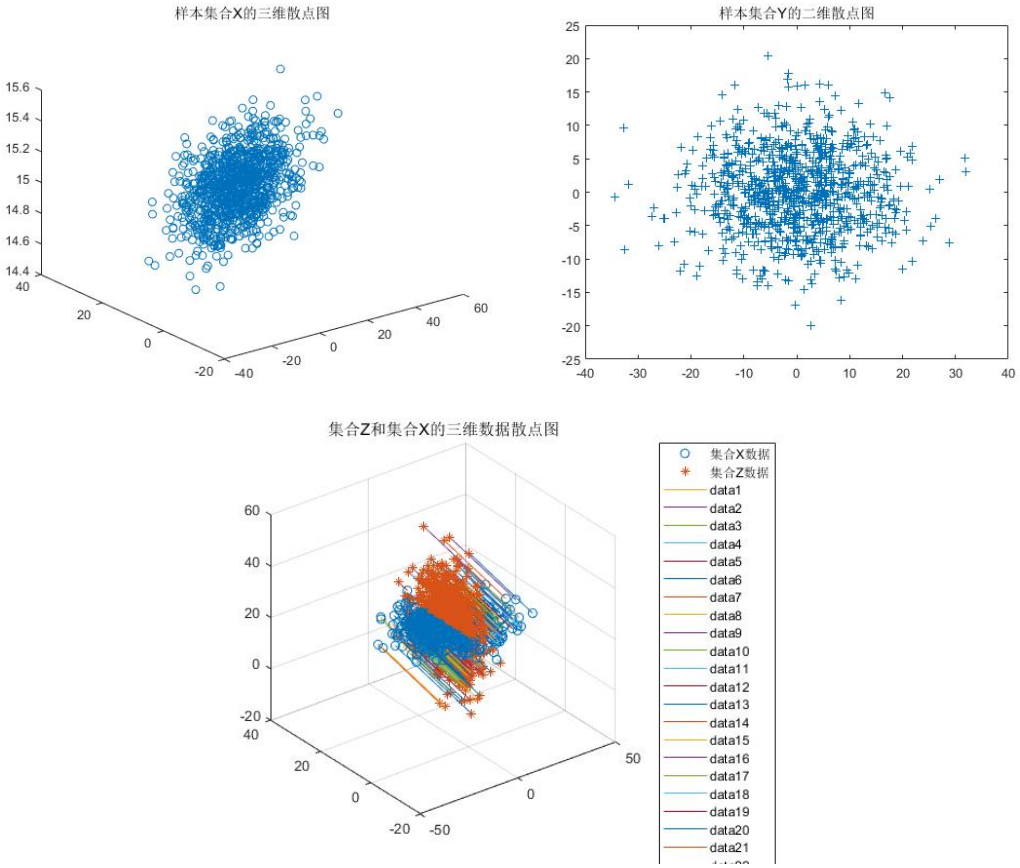

图 2-5. N=1000 时集合 X、集合 Y 的散点图，集合 Z 和集合 X 的三维数据散点图

对集合 **Z** 中的每一个向量 **z**，及与其对应的集合 **X** 中的向量 **x**，计算它们的误差平方值 $\left|\mathbf{x}-\mathbf{z}\right|^2$，并计算所有的这些误差平方之和；计算它们的均方误差（误差平方和的平均值），结果如表 2 所示。

表 2. 集合 Z 和集合 X 之间的误差统计

	N = 10	N = 20	N = 50	N = 100	N = 1000
误差平方之和	1037.71355	6592.20309	7629.60112	17653.3281	187264.830
均方误差	103.771355	167.819520	152.592022	176.533281	187.264830

四、 讨论与分析

1. 利用 PCA 进行特征空间的规整化

由实验结果图 1-1 至图 1-5，或者将集合 **X** 与集合 **Y** 画在同一个图中（以 N=10000 为例），如图 4-1 所示，经观察可以发现，集合 **Y** 是由集合 **X** 先经过平移变换，再旋转一个角度得到的。因为 $\mathbf{y}=\mathbf{V}\left(\mathbf{x}-\mathbf{m}\right)$ 中，**V** 矩阵相当于进行旋转，减 **m** 相当于平移。原先集合 **X** 的样本中心不在（0, 0）处，且样本的横轴与 x 轴，纵轴与 y 轴之间存在一定的角度；而变换后得到的集合 **Y** 的样本中心在（0, 0）处，且样本的横轴平行于 x 轴，纵轴平行于 y 轴。因此，PCA 方法的意义是对数据集合在特征空间中进行平移和旋转，进行规整化。

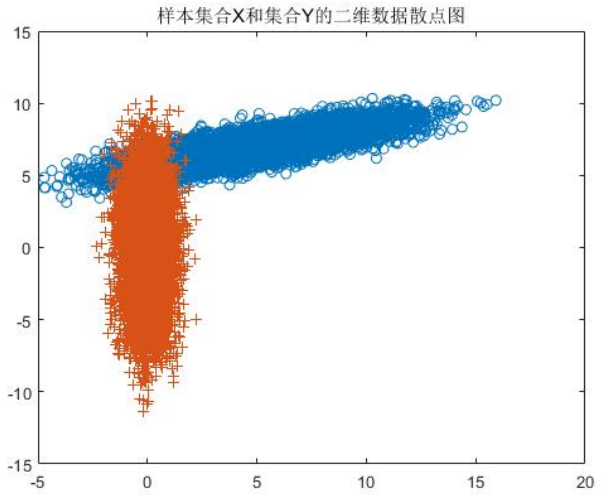

图 4-1. N=10000 时集合 X、集合 Y 的二维数据散点图

由表 1 可以看出，随着样本点数 N 的增加，计算得到均值矢量和协方差矩阵越来越接近给定的均值矢量 **μ** 和协方差矩阵 $\sum$，因此，散布矩阵 **S** 也越接近给定的协方差矩阵的(N-1)倍，即样本数量越大，信息损失越小。

2. 利用 PCA 进行特征空间降维

将样本集合 **Z** 与集合 **X** 的三维数据散点图（以 N=1000 为例）进行适度的旋转得到图 4-2，经观察可以发现，集合 **Z** 是集合 **X** 在某方向上的垂直投影，该方向是由集合 X 在其散布矩阵 **S** 的最大特征值 λ_1 与次大特征值 λ_2 所对应的特征向量 $\mathbf{e}_1$、$\mathbf{e}_2$ 构成的基向量形成的平面，符合前面介绍的 PCA 原理：直线的最优方向 **e** 是散布矩阵 **S** 对应于最大特征值 λ 的特征矢量方向，向量 $\mathbf{X}_k$ 向样本均值的直线 **e** 作垂直投影即可得到特征点。因此，集合 **Z** 虽然是处于三维空间中，但其实是二维矢量，即 **Z** 为一个平面，这也可以从图 4-2 看出。

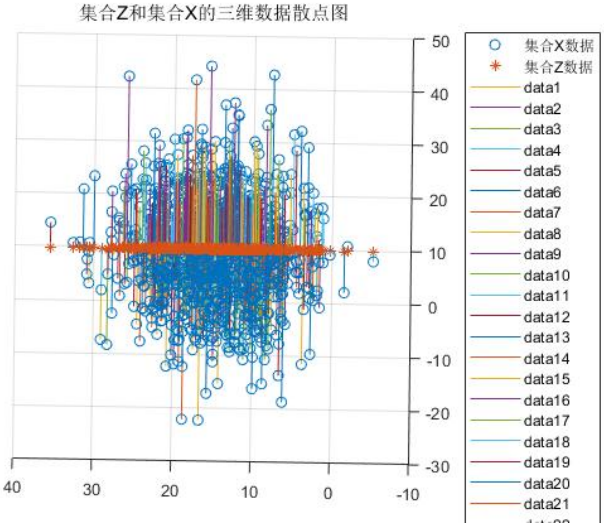

图 4-2. N=1000 时集合 Z 和集合 X 的三维数据散点图

由表 2 可以看出，随着样本点数 N 的增加，计算得到的集合 **Z** 和集合 **X** 之间的误差平方之和越来越大，而集合 **Z** 和集合 **X** 之间的均方误差并不是呈现逐渐增大的趋势，均方误差会出现波动。因此，为了排除样本数量对集合间误差的影响，我们可以采用均方误差作为误差衡量指标。

综上所述，PCA 降维的基本思想是采取特征线性组合的方式减少特征空间中的维数，将高维数据投影到低维空间中，减少问题分析的复杂度。实验结果表明，利用 PCA 方法可以对数据集合在特征空间中进行平移和旋转，从而进行规整化；经过 PCA 降维得到的集合 **Z** 是集合 **X** 在某方向上的垂直投影，该方向是由集合 X 在其散布矩阵 **S** 的最大特征值 λ_1 与次大特征值 λ_2 所对应的特征向量 $\mathbf{e}_1$、$\mathbf{e}_2$ 构成的基向量形成的平面；此外，样本的数量越大，估计得到的样本均值向量越接近给定的均值矢量，散布矩阵 **S** 也越接近给定的协方差矩阵的(N-1)倍，信息损失越小。

附录.

1. 利用 PCA 进行特征空间的规整化

```
%%----------------Proj03-01：主分量分析 PCA--------------%%
%%--------------------Proj03-01-exp1-------------------%%
clc; clear;
N = 100000;%样本数量
miu = [5; 7];%均值
sigma = [9, 2.4; 2.4, 1];%协方差矩阵
X = mvnrnd(miu, sigma, N);%生成 N 个高斯分布的二维样本矢量
figure(1); plot(X(:, 1), X(:, 2), 'o'); title('样本集合 X 的二维散点图
');
m = mean(X)';%样本集合 X 的均值向量
mm = repmat(m, 1, N);%%用矩阵的方法计算，repmat 是复制和平铺矩阵
%S = (X' - mm) * (X' - mm)';
S = (N - 1) * cov(X);%散布矩阵
[V, D] = eig(S);%D 的对角线元素是特征值，V 的列是相应的特征向量
Y = V * (X' - mm);
figure(2); plot(Y(1, :), Y(2, :), '+'); title('集合 Y 的二维散点图');
figure(3); plot(X(:, 1), X(:, 2), 'o');
hold on; plot(Y(1, :), Y(2, :), '+'); title('样本集合 X 和集合 Y 的二维
数据散点图');
```

2. 利用 PCA 进行特征空间降维

```
%%----------------Proj03-01：主分量分析 PCA--------------%%
%%--------------------Proj03-01-exp2-------------------%%
clc; clear;
N = 20;%样本数量
miu = [10; 15; 15];%均值
sigma = [90, 2.5, 1.2; 2.5, 35, 0.2; 1.2, 0.2, 0.02];%协方差矩阵
X = mvnrnd(miu, sigma, N);%生成 N 个高斯分布的二维样本矢量
figure(1); plot3(X(:, 1), X(:, 2), X(:, 3), 'o'); title('样本集合 X
的三维散点图');
m = mean(X)';%样本集合 X 的均值向量
mm = repmat(m, 1, N);%%用矩阵的方法计算，repmat 是复制和平铺矩阵
S = (X' - mm) * (X' - mm)';
S1 = (N - 1) * cov(X);
[V, D] = eig(S1);%D 的对角线元素是特征值，V 的列是相应的特征向量
[D_sort, index] = sort(diag(D),'descend');
V_sort = V(:,index);
% Y = V * (X' - mm);
Y1 = V_sort(:, 1)' * (X' - mm);
Y2 = V_sort(:, 2)' * (X' - mm);
Y = [Y1; Y2];
```

```
figure(2); plot(Y(1, :), Y(2, :), '+'); title('样本集合 Y 的二维散点图
');
VV = inv(V);%求逆，这里使用没有排序之前的向量矩阵，才能得到垂直投影方向，用排
序后的向量矩阵不能得到垂直投影方向
W = VV(:, 1:2);
Z = W * Y + mm;
figure(3); XX = plot3(X(:, 1), X(:, 2), X(:, 3), 'o');
hold on; ZZ = plot3(Z(1, :), Z(2, :), Z(3, :), '*');
legend([XX, ZZ], '集合 X 数据', '集合 Z 数据');
title('集合 Z 和集合 X 的三维数据散点图');
grid on;
for i = 1: N
    plot3([Z(1, i), X(i, 1)], [Z(2, i), X(i, 2)], [Z(3, i), X(i, 3)]);
end
grid on;
E = (X' - Z).^2;
Square_E = sum(sum(E));%计算所有的这些误差平方之和
MeanSquare_E = (1/N) * Square_E;%计算它们的均方误差
fprintf('误差平方之和 = %f\n', Square_E);
fprintf('均方误差 = %f\n', MeanSquare_E);
```