

# 25

## Selected Use Cases of Data

# 数据应用

将线性代数工具用于数据科学和机器学习实践



琴弦的低吟浅唱中易闻几何；

天体的星罗棋布上足见音律。

*There is geometry in the humming of the strings. There is music in the spacing of the spheres.*

—— 毕达哥拉斯 (Pythagoras) | 古希腊哲学家、数学家和音乐理论家 | 570 ~ 495 BC



- ◀ statsmodels.api.add\_constant() 线性回归增加一列常数 1
- ◀ statsmodels.api.OLS() 最小二乘法函数
- ◀ numpy.linalg.eig() 特征值分解
- ◀ numpy.linalg.svd() 奇异值分解
- ◀ sklearn.decomposition.PCA() 主成分分析函数

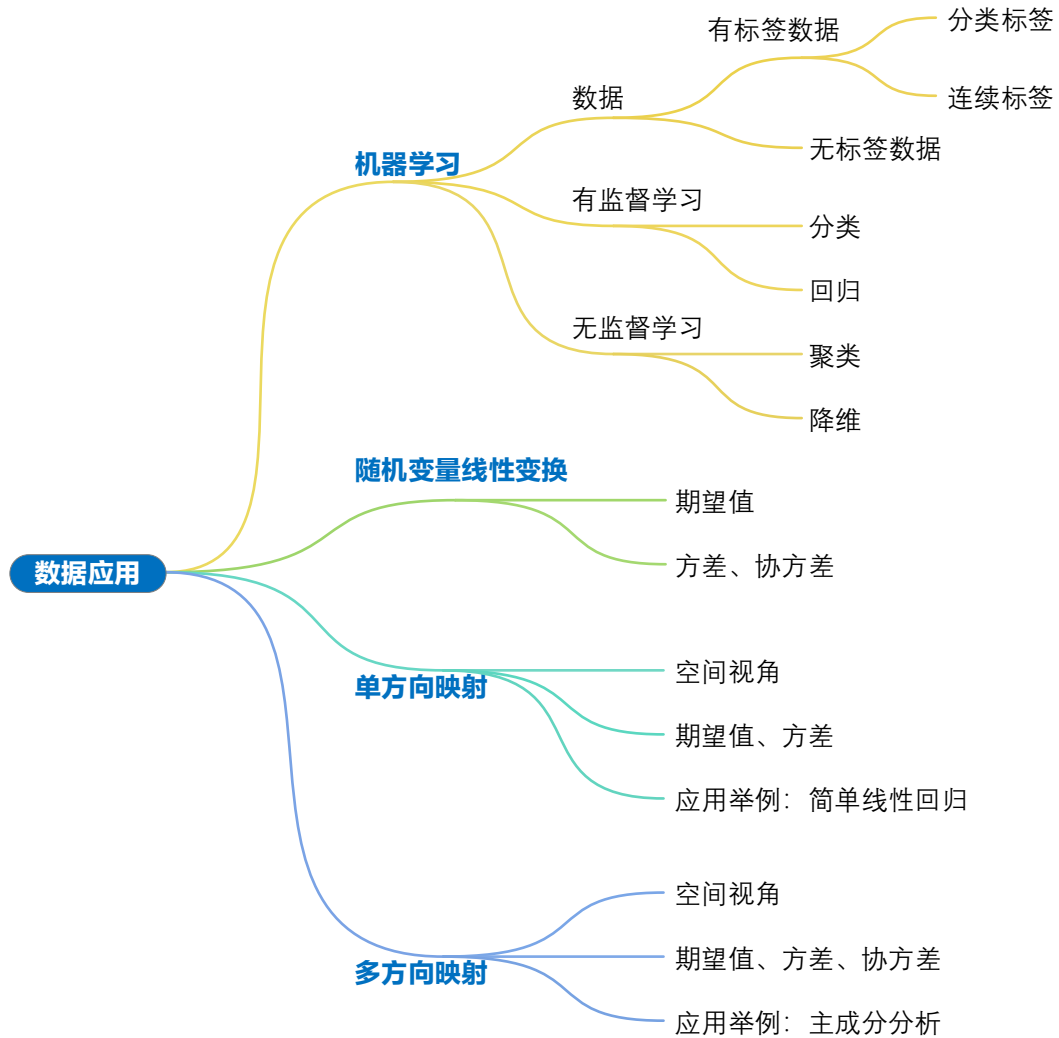
本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微课视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)



# 25.1 从线性代数到机器学习

本书第 23、24 章，即“数据三部曲”前两章，分别从空间、矩阵分解两个角度总结了本书之前介绍的重要线性代数工具。我们寻找向量空间、完成矩阵分解，并不仅仅因为它们有趣。实际上，本书中介绍的线性代数工具有助于我们用样本数据搭建数据科学、机器学习模型。

在前两章的基础上，本章一方面引出《统计至简》有关多元统计内容，另一方面预告本书线性代数工具在《数据科学》和《机器学习》中几个应用场景。

## 机器学习

本章首先聊一聊，什么是机器学习？

根据维基百科定义，机器学习算法是一类从数据中自动分析获得规律，并利用规律对未知数据进行预测的算法。

机器学习处理的问题有如下特征：(a) 基于数据，模型需要通过样本数据训练；(b) 黑箱或复杂系统，难以找到**控制方程** (governing equations)。控制方程指的是能够比较准确、完整描述某一现象或规律的数学方程，比如用  $y = ax^2 + bx + c$  描述抛物线轨迹。

而机器学习处理的数据通常为多特征数据，这就是为什么任何机器学习算法离不开线性代数工具。

## 有标签数据、无标签数据

根据输出值有无标签，如图 1 所示，数据可以分为**有标签数据** (labelled data) 和**无标签数据** (unlabelled data)。

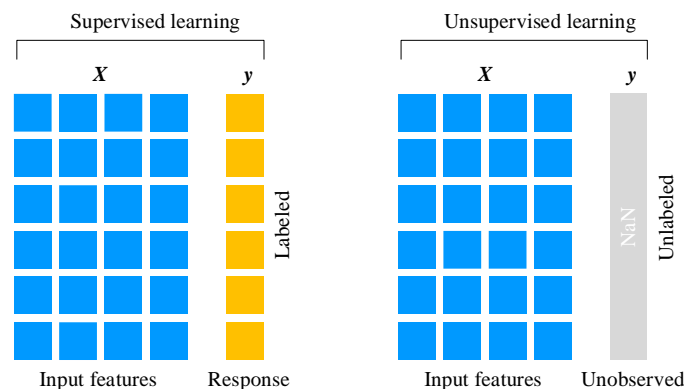


图 1. 根据有无标签分类数据

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。  
版权归清华大学出版社所有，请勿商用，引用请注明出处。  
代码及 PDF 文件下载：<https://github.com/Visualize-ML>  
本书配套微课视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>  
欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)

显然，鸢尾花数据集是有标签数据，因为数据的每一行代表一朵花，而每一朵花都对应一个特定的鸢尾花类别（图 2 最后一列），这个类别就是标签。

Index	Sepal length $X_1$	Sepal width $X_2$	Petal length $X_3$	Petal width $X_4$	Species $C$
1	5.1	3.5	1.4	0.2	Setosa $C_1$
2	4.9	3	1.4	0.2	
3	4.7	3.2	1.3	0.2	
...	...	...	...	...	
49	5.3	3.7	1.5	0.2	
50	5	3.3	1.4	0.2	Versicolor $C_2$
51	7	3.2	4.7	1.4	
52	6.4	3.2	4.5	1.5	
53	6.9	3.1	4.9	1.5	
...	...	...	...	...	
99	5.1	2.5	3	1.1	Virginica $C_3$
100	5.7	2.8	4.1	1.3	
101	6.3	3.3	6	2.5	
102	5.8	2.7	5.1	1.9	
103	7.1	3	5.9	2.1	
...	...	...	...	...	
149	6.2	3.4	5.4	2.3	
150	5.9	3	5.1	1.8	

图 2. 鸢尾花数据表格，单位为厘米 (cm)

很多场景，样本数据并没有标签。举个例子，图 3 所示为 2020 年度中 9 支股票的每个营业日股价数据。图 3 中数据共有 253 行，每行代表一个日期几只股票股价水平。

列方向来看，表格共有 10 列，第 1 列为营业日日期，其余 9 列每列为股价数据。从时间序列 (timeseries) 角度来看，图 3 中第一列时间点起到一个时间先后排序作用。图 3 数据显然没有类似图 2 标签。本系列丛书《数据科学》一册将专门讲解时间序列。

此外，本书很多应用场景中，我们并不考虑鸢尾花数据的标签；也就是说，我们将鸢尾花标签一列删除，得到无标签数据矩阵  $\mathbf{X}_{150 \times 4}$ 。

Date	TSLA	TSM	COST	NVDA	FB	AMZN	AAPL	NFLX	GOOGL
2-Jan-2020	86.05	58.26	281.10	239.51	209.78	1898.01	74.33	329.81	1368.68
3-Jan-2020	88.60	56.34	281.33	235.68	208.67	1874.97	73.61	325.90	1361.52
6-Jan-2020	90.31	55.69	281.41	236.67	212.60	1902.88	74.20	335.83	1397.81
7-Jan-2020	93.81	56.60	280.97	239.53	213.06	1906.86	73.85	330.75	1395.11
8-Jan-2020	98.43	57.01	284.19	239.98	215.22	1891.97	75.04	339.26	1405.04
9-Jan-2020	96.27	57.48	288.75	242.62	218.30	1901.05	76.63	335.66	1419.79
...	...	...	...	...	...	...	...	...	...
30-Dec-2020	694.78	108.49	373.71	525.83	271.87	3285.85	133.52	524.59	1736.25
31-Dec-2020	705.67	108.63	376.04	522.20	273.16	3256.93	132.49	540.73	1752.64

图 3. 股票收盘股价数据

## 有标签数据：分类、连续

本 PDF 文件为作者草稿，发布目的为方便读者在移动终端学习，终稿内容以清华大学出版社纸质出版物为准。

版权归清华大学出版社所有，请勿商用，引用请注明出处。

代码及 PDF 文件下载：<https://github.com/Visualize-ML>

本书配套微课视频均发布在 B 站——生姜 DrGinger：<https://space.bilibili.com/513194466>

欢迎大家批评指教，本书专属邮箱：[jiang.visualize.ml@gmail.com](mailto:jiang.visualize.ml@gmail.com)

有标签数据中，标签数值可以是**分类** (categorical)，也可以是**连续** (continuous)。

分类标签很好理解，比如鸢尾花数据的标签有三类 setosa、virginica、versicolor。它们可以用数字 0、1、2 来代表。

而有些数据的标签是连续的。本系列丛书《数学要素》一册中鸡兔同笼的回归问题中，鸡兔数量就是个好例子。横轴鸡的数量是回归问题的自变量；纵轴的兔子数量是因变量，就是连续标签。

再举个例子，用图 3 中 9 只股价来构造一个投资组合，目标是跟踪标普 500 涨跌；这时，标普 500 同时期的数据就是连续标签，显然这个标签对应的数据为连续数值。

### 有监督学习、无监督学习

根据数据是否有标签，机器学习可以分为两大类：

- ◀ **有监督学习** (supervised learning) 训练有标签值样本数据并得到模型，通过模型对新样本数据标签进行标签推断。
- ◀ **无监督学习** (unsupervised learning) 训练没有标签值的数据，并发现样本数据的结构。

### 四大类

如图 4 所示，根据标签类型，机器学习还可进一步细分成四大类问题。

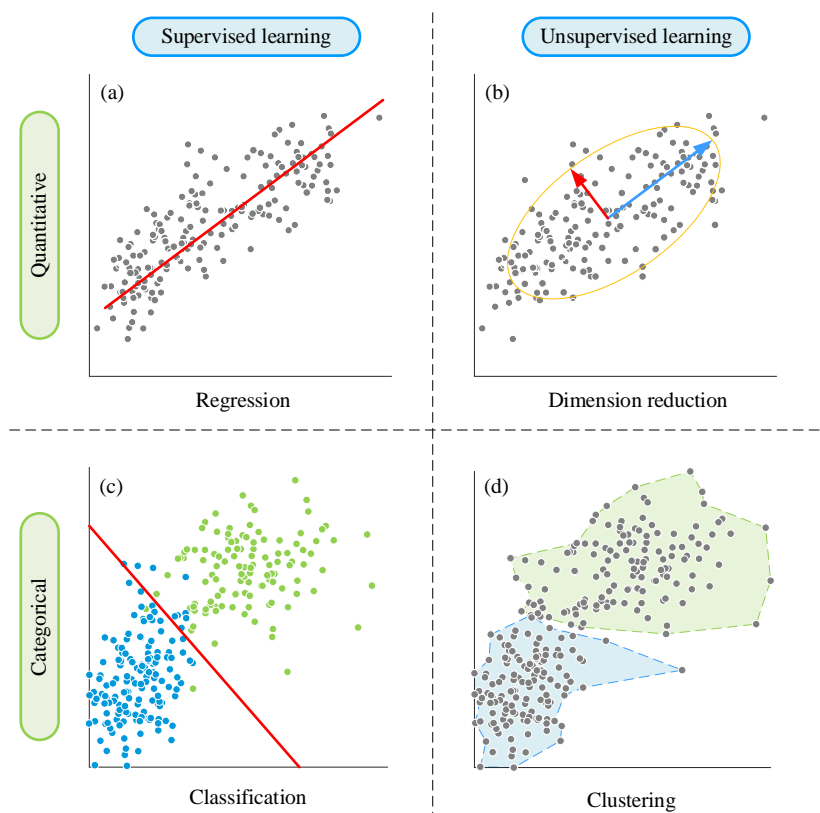


图 4. 根据数据是否有标签、标签类型细分机器学习算法

有监督学习中，如果标签为连续数据，对应的问题为**回归** (regression)，如图 4 (a)。如果标签为分类数据，对应的问题则是**分类** (classification)，如图 4 (c)。

无监督学习中，样本数据没有标签。如果目标是寻找规律、简化数据，这类问题叫做**降维** (dimension reduction)，比如主成分分析目的之一就是找到数据中占据主导地位的成分，如图 4 (b)。如果模型的目标是根据数据特征将样本数据分成不同的组别，这种问题叫做**聚类** (clustering)，如图 4 (d)。

实际上，数据科学和机器学习本来不分家，但是为了方便大家学习，作者根据图 4 所示规律将内容分成《数据科学》和《机器学习》两册。

《数据科学》主要解决图 4 (a) 和 (b) 两图对应的回归以及降维问题。

《机器学习》则关注图 4 (c) 和 (d) 所示分类和聚类问题，难度有所提高。

本系列丛书《数学要素》、《矩阵力量》、《统计至简》这三册为《数据科学》和《机器学习》提供了数学工具。特别地，本册《矩阵力量》提供的线性代数工具，是所有数学工具从一元到多元的推手，比如多元微积分、多元概率统计、多元优化等等。

本章下文就试图把几何、线性代数、概率统计、机器学习应用这几个元素串起来，让大家领略线性代数工具无处不在的力量。

## 25.2 从随机变量的线性变换说起

本节将随机变量的线性变换，和向量的仿射变换联系起来。这一节内容相对来说有一定难度，但是极其重要。本节是多元统计的理论基础。



本系列丛书《统计至简》一册还会深入探讨本节内容。

### 线性变换

如果  $X$  为一个随机变量，对  $X$  进行函数变换，可以得到其他的随机变量  $Y$ ：

$$Y = h(X) \quad (1)$$

特别地，如果  $h()$  为线性函数，则  $X$  到  $Y$  进行的就线性变换，比如：

$$Y = h(X) = aX + b \quad (2)$$

其中， $a$  和  $b$  为常数。这相当于几何中的缩放、平移两步操作。在线性代数中，上式相当于仿射变换。

(2) 中， $Y$  的期望和  $X$  的期望之间关系：

$$E(Y) = aE(X) + b \quad (3)$$

(2) 中,  $Y$  和  $X$  方差之间关系:

$$\text{var}(Y) = \text{var}(aX + b) = a^2 \text{var}(X) \quad (4)$$

## 二元随机变量

如果  $Y$  和二元随机变量  $(X_1, X_2)$  存在如下关系:

$$Y = aX_1 + bX_2 \quad (5)$$

(5) 可以写成:

$$Y = \begin{bmatrix} a & b \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \quad (6)$$

相信大家已经在上式中看到了本书反复讨论的线性映射关系。

$Y$  和二元随机变量  $(X_1, X_2)$  期望值之间存在如下关系:

$$E(Y) = E(aX_1 + bX_2) = aE(X_1) + bE(X_2) \quad (7)$$

(7) 可以写成如下矩阵运算形式:

$$E(Y) = \begin{bmatrix} a & b \end{bmatrix} \begin{bmatrix} E(X_1) \\ E(X_2) \end{bmatrix} \quad (8)$$

$Y$  和二元随机变量  $(X_1, X_2)$  方差、协方差存在如下关系:

$$\text{var}(Y) = \text{var}(aX_1 + bX_2) = a^2 \text{var}(X_1) + b^2 \text{var}(X_2) + 2ab \text{cov}(X_1, X_2) \quad (9)$$

(9) 可以写成:

$$\text{var}(Y) = \begin{bmatrix} a & b \end{bmatrix} \underbrace{\begin{bmatrix} \text{var}(X_1) & \text{cov}(X_1, X_2) \\ \text{cov}(X_1, X_2) & \text{var}(X_2) \end{bmatrix}}_{\Sigma} \begin{bmatrix} a \\ b \end{bmatrix} \quad (10)$$

相信大家已经在上式中看到了如下协方差矩阵:

$$\Sigma = \begin{bmatrix} \text{var}(X_1) & \text{cov}(X_1, X_2) \\ \text{cov}(X_1, X_2) & \text{var}(X_2) \end{bmatrix} \quad (11)$$

也就是说, (10) 可以写成:

$$\text{var}(Y) = \begin{bmatrix} a & b \end{bmatrix} \Sigma \begin{bmatrix} a \\ b \end{bmatrix} \quad (12)$$

### D 维随机变量

如果  $D$  维随机变量  $\zeta = [Z_1, Z_2, \dots, Z_D]^T$  服从多元高斯分布  $N(\mathbf{0}, \mathbf{I})$ ，即均值为  $\mathbf{0}$ ，协方差矩阵为单位矩阵：

$$\zeta = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_D \end{bmatrix}, \quad \mu_\zeta = E(\zeta) = \mathbf{0} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \quad \text{var}(\zeta) = \mathbf{I}_{D \times D} = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix} \quad (13)$$

其中，希腊字母  $\zeta$  读作 zeta。

而  $D$  维随机变量  $\chi = [X_1, X_2, \dots, X_D]^T$  和  $\zeta$  存在如下线性关系：

$$\chi = \begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_D \end{bmatrix} = \mathbf{V}^T \zeta + \mu = \mathbf{V}^T \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_D \end{bmatrix} + \begin{bmatrix} \mu_1 \\ \mu_2 \\ \vdots \\ \mu_D \end{bmatrix} \quad (14)$$

▲ 注意， $\chi$  为列向量，列向量元素个数为  $D$ ，即  $D$  行。

$\chi$  的期望值 (即质心) 为：

$$\mu_\chi = E(\chi) = \mu \quad (15)$$

注意，我们在此约定  $E(\chi)$  为列向量。求期望值运算符  $E(\bullet)$  作用于列向量  $\chi$ ，结果还是列向量。而  $E(X)$  代表  $E(\bullet)$  作用于数据矩阵  $X$ 。 $X$  的每一列代表一个随机变量，因此  $E(X)$  为行向量。

$\chi$  的协方差为：

$$\begin{aligned} \text{var}(\chi) &= \Sigma_\chi = \text{cov}(\chi, \chi) \\ &= E\left((\chi - E(\chi))(\chi - E(\chi))^T\right) \\ &= \frac{(\chi - \mu_\chi)(\chi - \mu_\chi)^T}{n} = \mathbf{V}^T \frac{\zeta \zeta^T}{n} \mathbf{V} = \mathbf{V}^T \mathbf{I}_{D \times D} \mathbf{V} = \mathbf{V}^T \mathbf{V} \end{aligned} \quad (16)$$

也就是说  $\chi$  服从  $N(\mu, \mathbf{V}\mathbf{V})$ 。

▲ 注意，(16) 计算总体方差，因此分母为  $n$ 。此外注意  $\zeta \zeta^T$  转置  $T$  所在位置，有别于本书前文计算数据矩阵  $X$  的协方差矩阵时遇到的  $X^T X$ 。

如果  $\chi$  和  $\gamma = [Y_1, Y_2, \dots, Y_D]^T$  满足如下线性映射关系：

$$\gamma = \mathbf{A}\chi \quad (17)$$

$\gamma$  的期望值 (即质心) 为：



$$\boldsymbol{\mu}_\gamma = \mathbf{E}(\gamma) = \mathbf{A}\boldsymbol{\mu} \quad (18)$$

$\gamma$  的协方差为:

$$\text{var}(\gamma) = \boldsymbol{\Sigma}_\gamma = \mathbf{A}\boldsymbol{\Sigma}_x\mathbf{A}^\text{T} \quad (19)$$

也就是说  $\gamma$  服从  $N(\mathbf{A}\boldsymbol{\mu}, \mathbf{A}\boldsymbol{\Sigma}_x\mathbf{A})$ 。

相信很多读者对本节内容已经感到云里雾里，下面几节展开讲解本节内容。

## 25.3 单方向映射

### 随机变量视角

$D$  个随机变量,  $X_1, X_2 \dots X_D$ , 通过如下组合构造随机变量  $Y$ :

$$Y = v_1 X_1 + v_2 X_2 + \dots + v_D X_D \quad (20)$$

举个例子, 制作八宝粥时, 用到如下八种谷物——大米 ( $X_1$ )、小米 ( $X_2$ )、糯米 ( $X_3$ )、紫米 ( $X_4$ )、绿豆 ( $X_5$ )、红枣 ( $X_6$ )、花生 ( $X_7$ )、莲子 ( $X_8$ )。  $v_1, v_2 \dots v_D$  相当于八种谷物的配比。

### 向量视角

从向量角度看 (20):

$$\hat{\mathbf{y}} = v_1 \mathbf{x}_1 + v_2 \mathbf{x}_2 + \dots + v_D \mathbf{x}_D \quad (21)$$

(21) 中  $\hat{\mathbf{y}}$  头上“戴帽子”为了呼应下一节的线性回归, 避免混淆。如图 5 所示, (21) 就是线性组合。

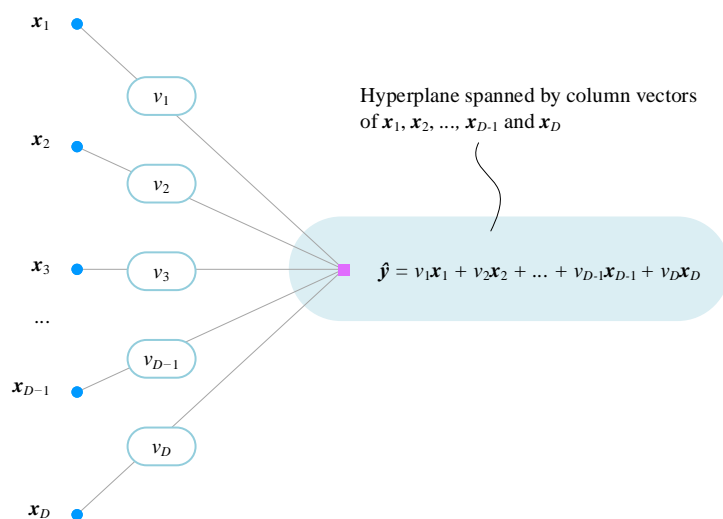


图 5.  $\mathbf{x}_1, \mathbf{x}_2 \dots \mathbf{x}_D$  线性组合

令  $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_D]$ , (21) 相当于  $\mathbf{X}$  向  $\mathbf{v}$  向量映射, 得到列向量  $\hat{\mathbf{y}}$  :

$$\hat{\mathbf{y}} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_D \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_D \end{bmatrix} = \begin{bmatrix} \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_D \end{bmatrix} \mathbf{v} = \mathbf{X}\mathbf{v} \quad (22)$$

特别地, 如果  $\mathbf{v}$  为单位向量, 上式就是正交投影。

## 空间视角

如图 6 所示, 从空间角度,  $\text{span}(\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_D)$  张成超平面  $H$ , 而  $\hat{\mathbf{y}}$  在超平面  $H$  中。 $\hat{\mathbf{y}}$  的坐标就是  $(v_1, v_2, \dots, v_D)$ 。

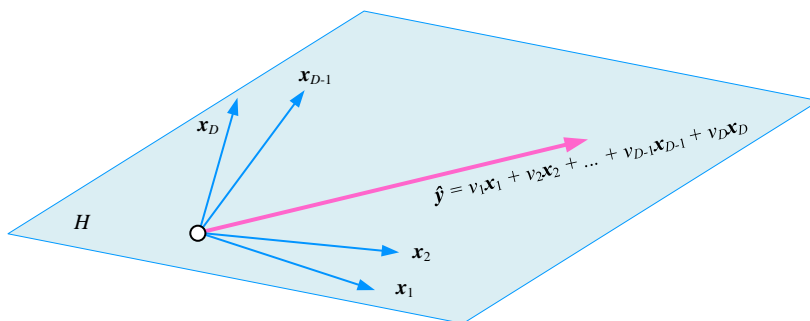


图 6.  $\hat{\mathbf{y}}$  在超平面  $H$  中

## 行向量视角

本章前文说的是列向量视角, 我们下面再看行向量视角。数据矩阵  $\mathbf{X}$  中的每一行对应行向量  $\mathbf{x}^{(i)}$ ,  $\mathbf{x}^{(i)}\mathbf{v} = \hat{y}^{(i)}$  相当于  $D$  维坐标映射到  $\text{span}(\mathbf{v})$  得到一个点。



请大家回忆本书第 10 章讲过的用张量积完成“二次投影”。

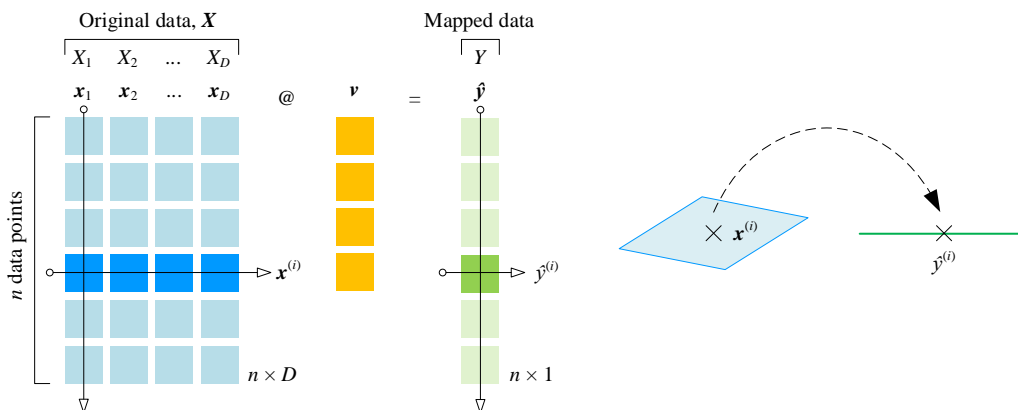
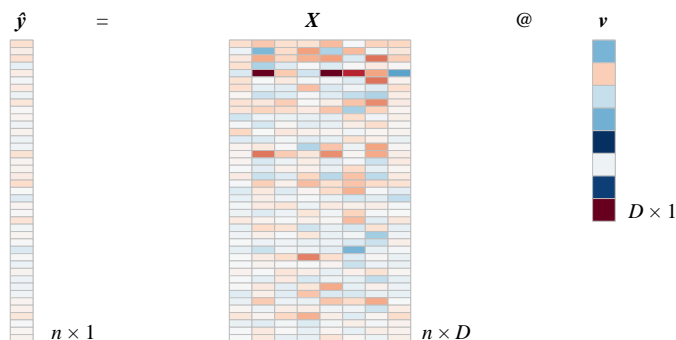


图 7. 数据矩阵  $X$  向  $v$  映射的行向量视角

## 期望值

下面用具体数据举例说明如何计算  $\hat{y}$  的期望值。图 8 所示热图对应数据矩阵  $X$  向  $v$  映射运算过程。

图 8. 矩阵  $X$  向  $v$  映射热图

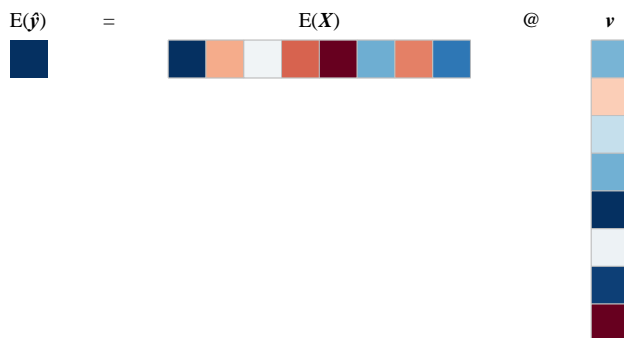
根据上一节内容，列向量  $\hat{y}$  期望值  $E(y)$  和矩阵  $X$  期望值  $E(X)$  关系为：

$$E(\hat{y}) = E(Xv) = E(X)v \quad (23)$$

其中， $E(X)$  为行向量：

$$E(X) = [E(x_1) \ E(x_2) \ \dots \ E(x_D)] \quad (24)$$

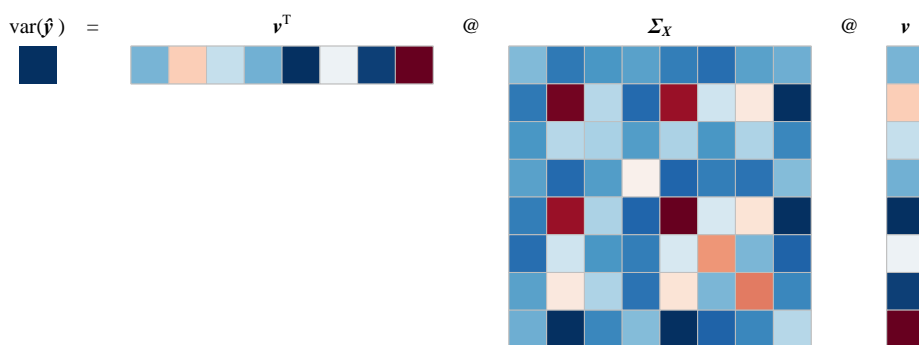
计算  $E(\hat{y})$  过程热图如图 9 所示。

图 9. 计算  $E(\hat{y})$  矩阵运算热图

## 方差

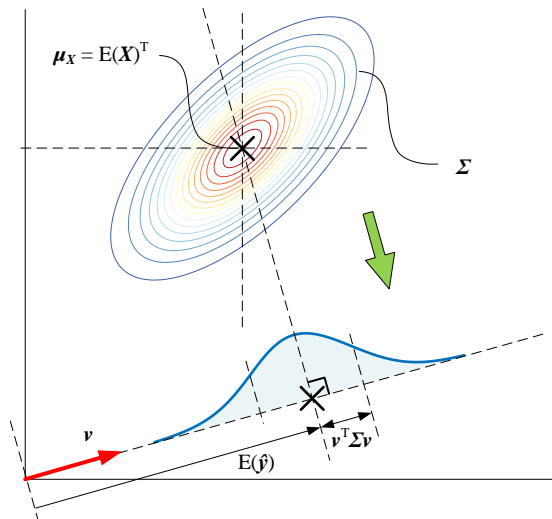
方差  $\text{var}(\hat{y})$  和数据矩阵  $X$  协方差矩阵  $\Sigma_X$  关系为：

$$\begin{aligned}
 \text{var}(\hat{y}) &= \frac{(\hat{y} - E(\hat{y}))^T (\hat{y} - E(\hat{y}))}{n-1} \\
 &= \frac{(Xv - E(X)v)^T (Xv - E(X)v)}{n-1} \\
 &= v^T \underbrace{\frac{(X - E(X))^T (X - E(X))}{n-1}}_{\Sigma_X} v \\
 &= v^T \Sigma_X v
 \end{aligned} \tag{25}$$

图 10 所示为计算  $\text{var}(\hat{y})$  矩阵热图。图 10. 计算  $\text{var}(\hat{y})$  矩阵运算热图

## 几何视角

图 11 所示为几何视角下的上述映射过程。注意，图 11 假设样本数据矩阵  $X$  服从二元高斯分布  $N(\mu_X, \Sigma)$ ，因此我们用椭圆代表它的分布。

图 11. 服从二元高斯分布的数据矩阵  $X$  向  $v$  映射得到  $\hat{y}$

## 25.4 线性回归

**线性回归** (linear regression) 是最为常用的回归算法。这种模型利用线性关系建立因变量与一个或多个自变量之间的联系。

**简单线性回归** (Simple Linear Regression, SLR) 为一元线性回归模型，是指模型中只含有一个自变量 ( $x$ ) 和一个因变量 ( $y$ )，即  $y = b_0 + b_1x_1 + \varepsilon$ 。

**多元线性回归** (multivariate regression) 模型则引入多个自变量 ( $x_1, x_2, \dots, x_D$ )，即回归分析中引入多个因子解释因变量 ( $y$ )。多元线性回归模型的数学表达式如下：

$$y = b_0 + b_1x_1 + b_2x_2 + \dots + b_Dx_D + \varepsilon \quad (26)$$

其中， $b_0$  为截距项； $b_1, b_2, \dots, b_D$  代表自变量系数； $\varepsilon$  为残差项； $D$  为自变量个数。

用向量代表具体值，(26) 可以写成：

$$y = \underbrace{b_0\mathbf{1} + b_1\mathbf{x}_1 + b_2\mathbf{x}_2 + \dots + b_D\mathbf{x}_D}_{\hat{y}} + \varepsilon \quad (27)$$

**▲ 注意**，全  $\mathbf{1}$  列向量也代表一个方向。而  $y$  代表监督学习中的连续标签。

换一种方式表达 (27)：

$$y = \underbrace{\mathbf{X}\mathbf{b}}_{\hat{y}} + \varepsilon \quad (28)$$

其中，

$$\mathbf{X}_{n \times (D+1)} = [\mathbf{I} \quad \mathbf{x}_1 \quad \mathbf{x}_2 \quad \dots \quad \mathbf{x}_D] = \begin{bmatrix} 1 & x_{1,1} & \dots & x_{1,D} \\ 1 & x_{2,1} & \dots & x_{2,D} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & \dots & x_{n,D} \end{bmatrix}_{n \times (D+1)}, \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_D \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon^{(1)} \\ \varepsilon^{(2)} \\ \vdots \\ \varepsilon^{(n)} \end{bmatrix} \quad (29)$$

**▲ 注意**，(29) 中设计矩阵  $\mathbf{X}$  包含全  $\mathbf{1}$  列向量，也就是说这个  $\mathbf{X}$  有  $D+1$  列。

### 线性组合

图 12 所示为多元 OLS 线性回归数据关系，图中  $y$  就是连续标签构成的列向量。

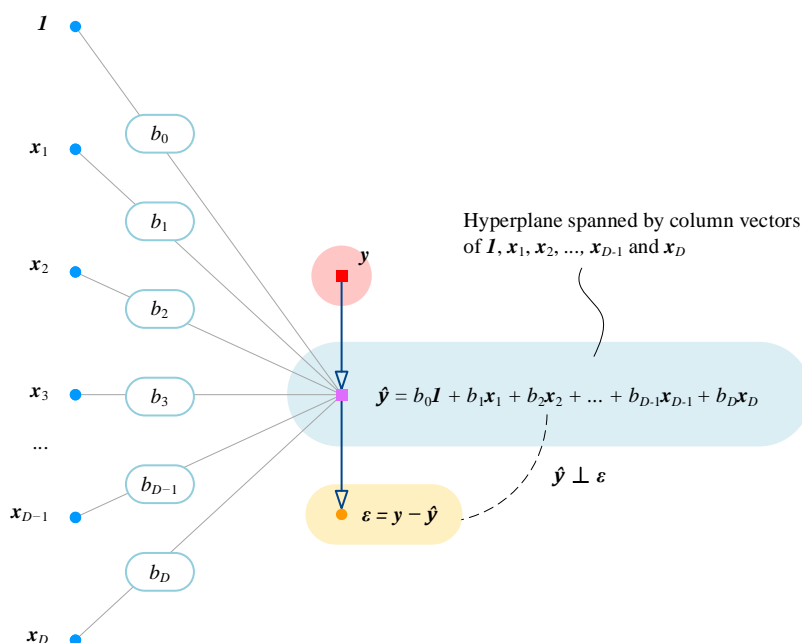


图 12. 多元 OLS 线性回归数据关系

### 投影视角

预测值构成的列向量  $\hat{\mathbf{y}}$ ，通过下式计算得到：

$$\hat{\mathbf{y}} = \mathbf{X}\mathbf{b} \quad (30)$$

⚠ 注意，这里我们用了“戴帽子”的  $\hat{\mathbf{y}}$ ，它代表对  $\mathbf{y}$  的估计。 $\mathbf{y}$  和  $\hat{\mathbf{y}}$  形状相同，两者之差为残差。

预测值向量  $\hat{\mathbf{y}}$  是自变量向量  $\mathbf{I}, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_D$  的线性组合。从空间角度来看， $[\mathbf{I}, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_D]$  构成一个超平面  $H = \text{span}(\mathbf{I}, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_D)$ 。 $\hat{\mathbf{y}}$  是  $\mathbf{y}$  在超平面  $H$  上的投影。

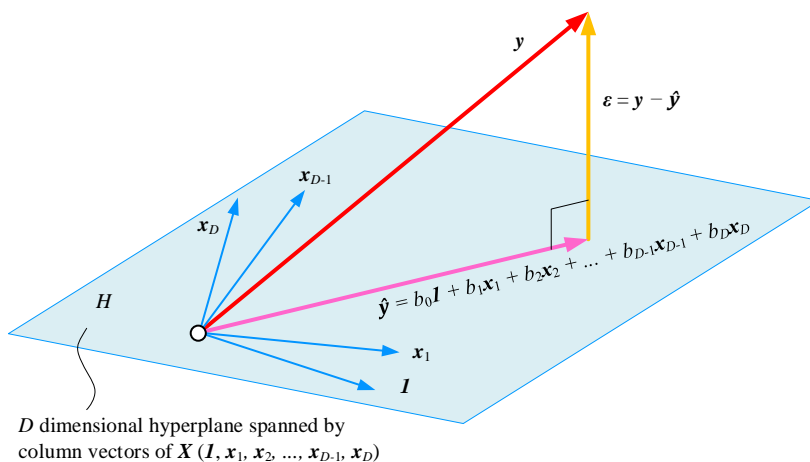


图 13. 几何角度解释多元 OLS 线性回归

而  $\mathbf{y}$  和  $\hat{\mathbf{y}}$  的差对应残差项  $\boldsymbol{\varepsilon}$ :

$$\boldsymbol{\varepsilon} = \mathbf{y} - \hat{\mathbf{y}} = \mathbf{y} - \mathbf{X}\mathbf{b} \quad (31)$$

如图 13 所示, 残差向量  $\boldsymbol{\varepsilon}$  垂直于  $\text{span}(\mathbf{I}, \mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_D)$ :

$$\boldsymbol{\varepsilon} \perp \mathbf{X} \Rightarrow \mathbf{X}^T \boldsymbol{\varepsilon} = \mathbf{0} \quad (32)$$

将 (31) 代入 (32) 得到:

$$\mathbf{X}^T (\mathbf{y} - \mathbf{X}\mathbf{b}) = \mathbf{0} \Rightarrow \mathbf{X}^T \mathbf{X}\mathbf{b} = \mathbf{X}^T \mathbf{y} \quad (33)$$

求解得到  $\mathbf{b}$ :

$$\mathbf{b} = (\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T \mathbf{y} \quad (34)$$

本书中, 我们已经不止一起提到 (34)。请大家注意从数据、向量、几何、空间、优化等视角理解 (34)。此外,  $(\mathbf{X}^T \mathbf{X})^{-1} \mathbf{X}^T$  叫做  $\mathbf{X}$  的广义逆, 或伪逆。还请大家注意, 只有  $\mathbf{X}$  为列满秩时,  $\mathbf{X}^T \mathbf{X}$  才存在逆。

## QR 分解

利用 QR 分解结果求解  $\mathbf{b}$ 。把  $\mathbf{X} = \mathbf{Q}\mathbf{R}$  代入 (34) 得到:

$$\begin{aligned} \mathbf{b} &= ((\mathbf{Q}\mathbf{R})^T \mathbf{Q}\mathbf{R})^{-1} (\mathbf{Q}\mathbf{R})^T \mathbf{y} = \left( \mathbf{R}^T \underbrace{\mathbf{Q}^T \mathbf{Q}}_{\mathbf{I}} \mathbf{R} \right)^{-1} \mathbf{R}^T \mathbf{Q}^T \mathbf{y} \\ &= \mathbf{R}^{-1} \underbrace{(\mathbf{R}^T)^{-1} \mathbf{R}^T}_{\mathbf{I}} \mathbf{Q}^T \mathbf{y} = \mathbf{R}^{-1} \mathbf{Q}^T \mathbf{y} \end{aligned} \quad (35)$$

## 奇异值分解

类似地, 利用 SVD 分解结果,  $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ ,  $\mathbf{b}$  可以整理为:

$$\begin{aligned} \mathbf{b} &= ((\mathbf{U}\mathbf{S}\mathbf{V}^T)^T \mathbf{U}\mathbf{S}\mathbf{V}^T)^{-1} (\mathbf{U}\mathbf{S}\mathbf{V}^T)^T \mathbf{y} = \left( (\mathbf{S}\mathbf{V}^T)^T \underbrace{\mathbf{U}^T \mathbf{U}}_{\mathbf{I}} \mathbf{S}\mathbf{V}^T \right)^{-1} (\mathbf{S}\mathbf{V}^T)^T \mathbf{U}^T \mathbf{y} \\ &= ((\mathbf{S}\mathbf{V}^T)^T \mathbf{S}\mathbf{V}^T)^{-1} (\mathbf{S}\mathbf{V}^T)^T \mathbf{U}^T \mathbf{y} \\ &= (\mathbf{S}\mathbf{V}^T)^{-1} \underbrace{((\mathbf{S}\mathbf{V}^T)^T)^{-1} (\mathbf{S}\mathbf{V}^T)^T}_{\mathbf{I}} \mathbf{U}^T \mathbf{y} = (\mathbf{S}\mathbf{V}^T)^{-1} \mathbf{U}^T \mathbf{y} \end{aligned} \quad (36)$$

也就是说, 对比 SVD 分解 ( $\mathbf{X} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ ) 和 QR 分解 ( $\mathbf{X} = \mathbf{Q}\mathbf{R}$ ),  $\mathbf{U}$  可以视作  $\mathbf{Q}$ , 因为两者都是正交矩阵; 而  $\mathbf{S}\mathbf{V}^T$  可以视作  $\mathbf{R}$ 。

虽然  $\mathbf{U}$  和  $\mathbf{Q}$  都是正交矩阵, 两者从本质上是不同的。请大家自行回忆上一章内容, 对比两种分解。

实际上，我们不需要大费周章，直接将 QR 分解或 SVD 分解结果直接代入  $y = Xb$  等式便可以求得  $b$ 。

### 优化视角

下面以本节多元线性回归为例，介绍如何利用**最小二乘法** (Ordinary Least Squares, OLS)，即最小化误差的平方和，寻找最佳参数  $b$ 。

残差项平方和可以写成：

$$\sum_{i=1}^n \varepsilon_i^2 = \varepsilon^T \varepsilon \quad (37)$$

将 (31) 带入 (37)，展开得到：

$$\sum_{i=1}^n \varepsilon_i^2 = (y - Xb)^T (y - Xb) = (y^T - b^T X^T)(y - Xb) = y^T y - y^T Xb - b^T X^T y + b^T X^T Xb \quad (38)$$

上式， $y^T Xb$  和  $b^T X^T y$  都是标量，转置不影响结果：

$$b^T X^T y = (b^T X^T y)^T = y^T Xb \quad (39)$$

因此 (38) 可以写成：

$$\sum_{i=1}^n \varepsilon_i^2 = y^T y - 2y^T Xb + b^T X^T Xb \quad (40)$$

构造最小化问题，令目标函数  $f(b)$  为：

$$f(b) = y^T y - 2y^T Xb + b^T X^T Xb \quad (41)$$

$f(b)$  对向量  $b$  求一阶导为  $0$  得到如下等式：

$$\frac{\partial f(b)}{\partial b} = -2X^T y + 2X^T Xb = 0 \quad (42)$$

整理 (42)，得到：

$$X^T Xb = X^T y \quad (43)$$

通过优化视角，我们也得到了(33)。

此外， $f(b)$  对向量  $b$  求二阶导得到黑塞矩阵：

$$\frac{\partial^2 f(b)}{\partial b \partial b^T} = 2X^T X \quad (44)$$

如果  $X$  列满秩，它的格拉姆矩阵  $X^T X$  正定。因此，满足 (43) 的鞍点  $b$  为极小值点。进一步， $f(b)$  为二次型，可以判定  $b$  为最小值点。





本系列丛书《统计至简》一册将介绍多元线性回归和条件概率之间关系。

## 25.5 多方向映射

矩阵  $X$  向  $v_1$  和  $v_2$  两个不同方向投影：

$$y_1 = \begin{bmatrix} x_1 & x_2 & \cdots & x_D \end{bmatrix} \begin{bmatrix} v_{1,1} \\ v_{2,1} \\ \vdots \\ v_{D,1} \end{bmatrix} = Xv_1, \quad y_2 = \begin{bmatrix} x_1 & x_2 & \cdots & x_D \end{bmatrix} \begin{bmatrix} v_{1,2} \\ v_{2,2} \\ \vdots \\ v_{D,2} \end{bmatrix} = Xv_2 \quad (45)$$

还是用八宝粥的例子，(45) 相当于两个不同配方的八宝粥。

合并 (45) 两个等式，得到：

$$\begin{aligned} Y_{n \times 2} &= \begin{bmatrix} y_1 & y_2 \end{bmatrix} = \begin{bmatrix} x_1 & x_2 & \cdots & x_D \end{bmatrix} \begin{bmatrix} v_1 & v_2 \end{bmatrix} \\ &= X_{n \times D} V_{D \times 2} \end{aligned} \quad (46)$$

图 14 所示为上述矩阵运算示意图。请大家自行从向量空间视角分析上式。

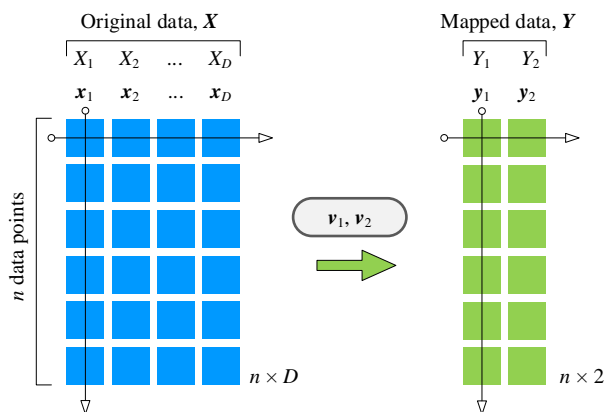
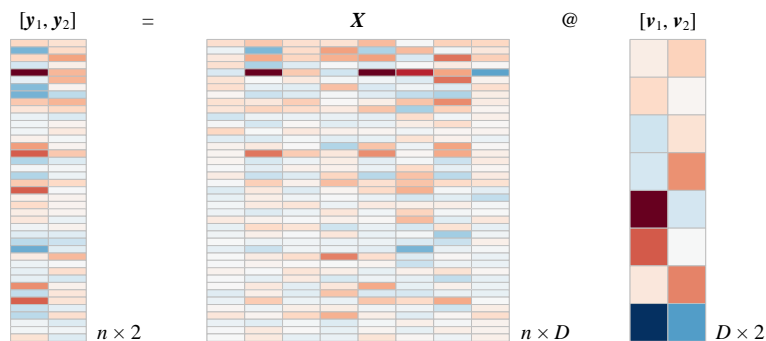


图 14. 数据朝两个方向映射

图 15 所示为数据  $X$  朝两个方向映射对应的运算热图。

图 15. 数据  $X$  朝两个方向映射对应的运算热图

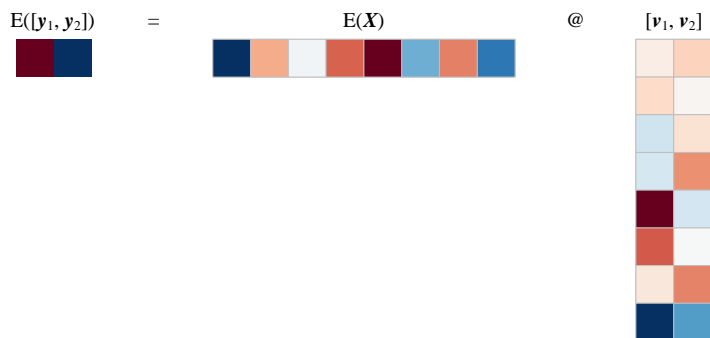
## 期望值

期望值  $[E(y_1), E(y_2)]$  和期望值向量  $E(X)$  关系为：

$$[E(y_1) \ E(y_2)] = [E(X)v_1 \ E(X)v_2] = E(X)V \quad (47)$$

比较 (18) 和 (47)，两个等式不同点在于转置。(18) 中随机变量向量为列向量，而上式中  $E(X)$  为行向量。

图 16 所示为计算期望值向量  $[E(y_1), E(y_2)]$  的热图。

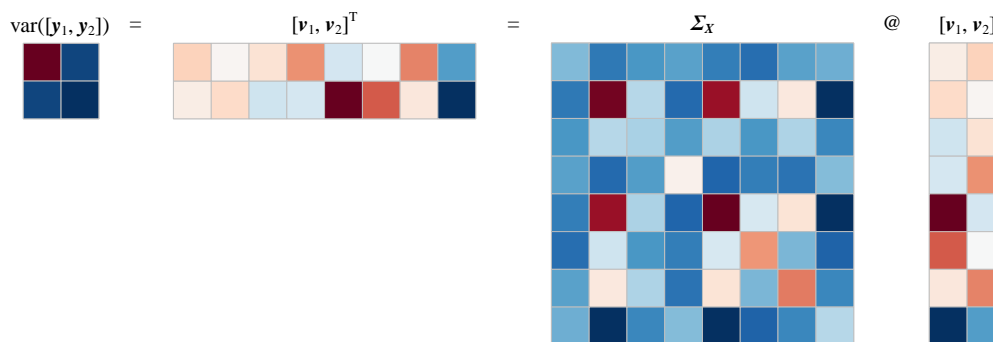
图 16. 计算期望值  $[E(y_1), E(y_2)]$  矩阵运算热图

## 协方差

$[y_1, y_2]$  协方差为：

$$\Sigma_Y = \begin{bmatrix} \sigma_{y_1}^2 & \rho_{y_1, y_2} \sigma_{y_1} \sigma_{y_2} \\ \rho_{y_1, y_2} \sigma_{y_1} \sigma_{y_2} & \sigma_{y_2}^2 \end{bmatrix} = \begin{bmatrix} v_1^T \\ v_2^T \end{bmatrix} \Sigma_X \begin{bmatrix} v_1 & v_2 \end{bmatrix} = V^T \Sigma_X V \quad (48)$$

(19) 和 (48) 也差在转置运算。注意，上式中  $V$  并非方阵。

图 17. 计算  $[y_1, y_2]$  协方差矩阵运算热图

## 25.6 主成分分析

**主成分分析** (principal component analysis, PCA) 最初由**卡尔·皮尔逊** (Karl Pearson) 在 1901 提出。主成分分析就是多方向映射。

通过线性变换，PCA 将多维数据投影到一个新的正交坐标系，把原始数据中的最大方差成分提取出来。PCA 也是数据降维的重要方法之一。

如图 18 所示，PCA 的一般步骤如下：

- ▶ 对原始数据  $X_{n \times D}$  作**标准化** (standardization) 处理，得到  $z$  分数  $Z_X$ ；
- ▶ 计算  $z$  分数  $X_z$  协方差矩阵，即原始数据  $X$  的相关性系数矩阵  $P$ ；
- ▶ 计算  $P$  特征值  $\lambda_i$  与特征向量矩阵  $V_{D \times D}$ ；
- ▶ 对特征值  $\lambda_i$  从大到小排序，选择其中特征值最大的  $p$  个特征向量作为主成分方向；
- ▶ 将标准化数据投影到规范正交基  $[v_1, v_2, \dots, v_p]$  构建的新空间中，得到  $Y_{n \times p}$ 。

上述 PCA 流程仅仅是几种技术路线之一，本节最后会列出六种常用 PCA 技术路线。

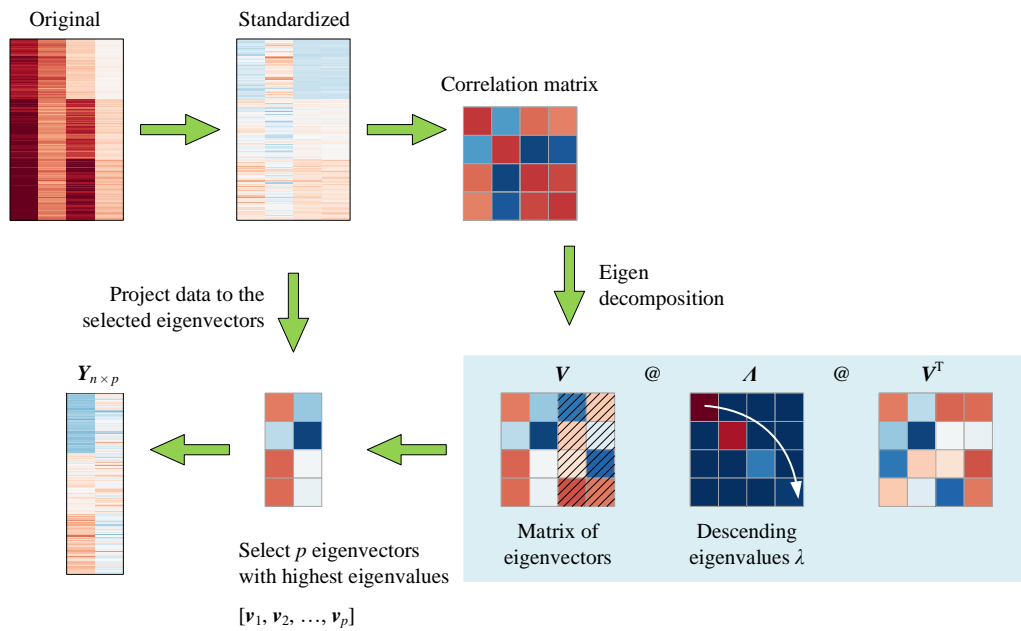


图 18. 主成分分析过程，基于特征值分解

数据标准化中包括去均值，这样新数据每个特征的均值为 0。这相当于把数据的质心移到原点。而标准化还包括用均方差完成“缩放”，以防止不同特征上方差差异过大。

原始数据各个特征方差差别不大时，不需要对  $X$  标准化，只需要中心化获得  $X_c$  即可。

作为重要的降维工具，PCA 可以显著减少数据的维数，同时保留数据中对方差贡献最大的成分。另外对于多维数据，PCA 可以作为一种数据可视化的工具。PCA 结果还可以用来构造回归模型。本系列丛书《数据科学》将深入介绍这些话题。

### 线性组合

如图 19 所示，主成分分析过程本质上也是线性组合，即  $X_{n \times D}$  ( $X_c$  或  $Z_X$ ) 线性组合组合得到  $Y_{n \times D}$  列向量，并选取结果中  $1 \sim p$  列列向量作为主成分。

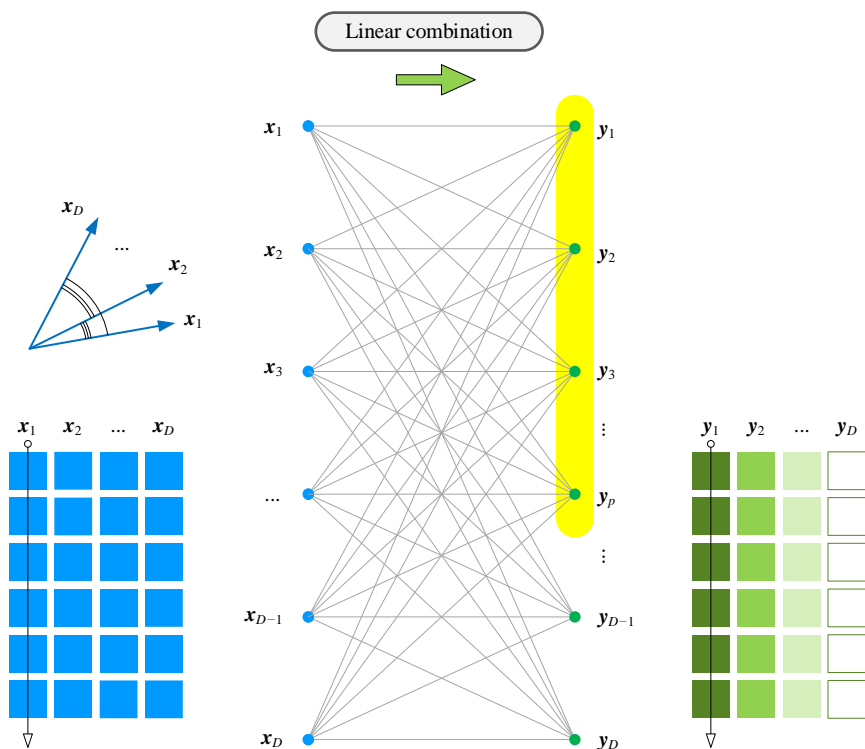


图 19. 线性组合

## 六条技术路线

表 1 总结了 PCA 六条主要技术路线，其中用到了奇异值分解、特征值分解两种矩阵分解工具。矩阵分解的对象对应六种不同矩阵，这六种矩阵都衍生自原始数据矩阵  $X$ 。

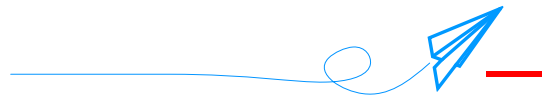
表 1 还通过颜色告诉我们，这六条技术路线本质上就是三种路线。比如，对原始数据  $X$  奇异值分解，等价于对其格拉姆矩阵  $G$  特征值分解。

➡ 我们将在《统计至简》一册探讨这六条技术路线的区别和联系。

表 1. 六条 PCA 技术路线

对象	方法	结果
原始数据矩阵 $X$	奇异值分解	$X = U_X S_X V_X^T$
格拉姆矩阵 $G = X^T X$	特征值分解	$G = V_X \Lambda_X V_X^T$
中心化数据矩阵 $X_c = X - E(X)$	奇异值分解	$X_c = U_c S_c V_c^T$

协方差矩阵 $\Sigma = \frac{(X - E(X))^T (X - E(X))}{n-1}$	特征值分解	$\Sigma = V \Lambda V^T$
标准化数据 (z 分数) $Z_X = (X - E(X)) D^{-1}$ $D = \text{diag}(\text{diag}(\Sigma))^{\frac{1}{2}}$	奇异值分解	$Z_X = U_Z S_Z V_Z^T$
相关性系数矩阵 $P = D^{-1} \Sigma D^{-1}$ $D = \text{diag}(\text{diag}(\Sigma))^{\frac{1}{2}}$	特征值分解	$P = V_Z \Lambda_Z V_Z^T$



本章是“数据三部曲”的最后一章，也是本书的最后一章。

通过这一章内容，作者希望能给大家提供一个更高的视角，让大家看到代数、线性代数、几何、概率统计、微积分、优化问题之间的联系，也同时展望线性代数工具在数据科学、机器学习领域的应用。

作者希望大家看完本册后，能对线性代数的印象彻底改观。

向量、矩阵、矩阵乘法、矩阵分解、向量空间等等不再是不知所云的线性代数概念，它们是解决实际问题无坚不摧的刀枪剑戟。

总有一天，我们会忘记线性代数的细枝末节；但是，那一天到来时，希望我们还能记得这几句话：

有数据的地方，必有矩阵！

有矩阵的地方，更有向量！

有向量的地方，就有几何！

有几何的地方，皆有空间！

有数据的地方，定有统计！

让我们在《统计至简》一册，不见不散！