

# 第四章 数据的分类分析方法

Lianghai Xiao

<https://github.com/styluck/mlb>

作业邮箱: alswhfx@126.com

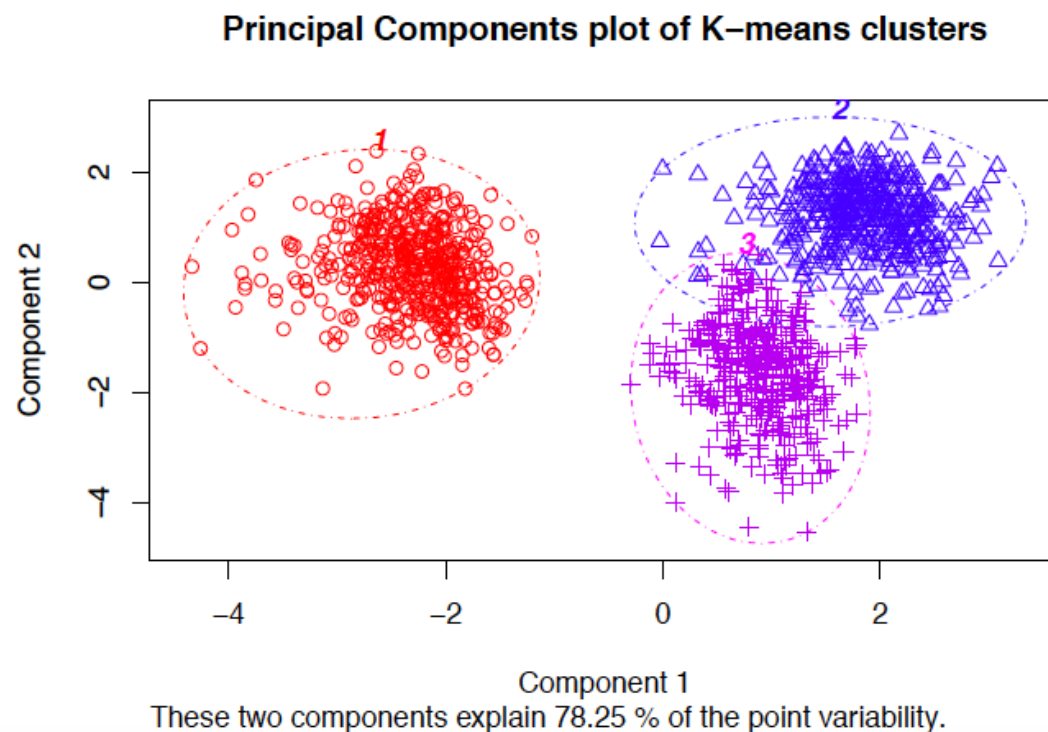
# 目录

- 数据特征的提取与描述
  - 主成分分析
  - 因子分析
- 数据的分类分析方法
  - 聚类分析
  - 判别分析
  - Logistic回归
  - 支持向量机

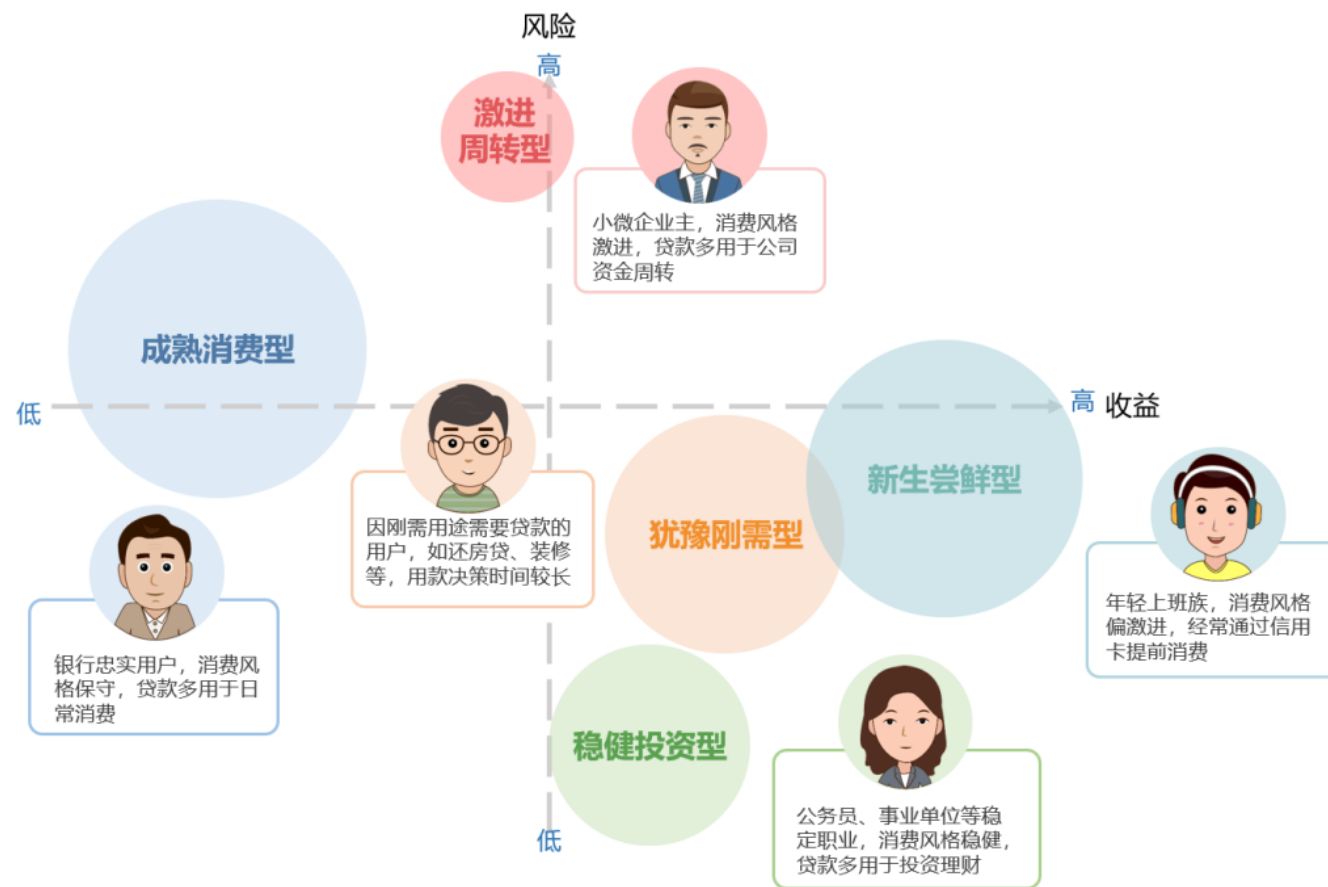
# 聚类分析

# 聚类

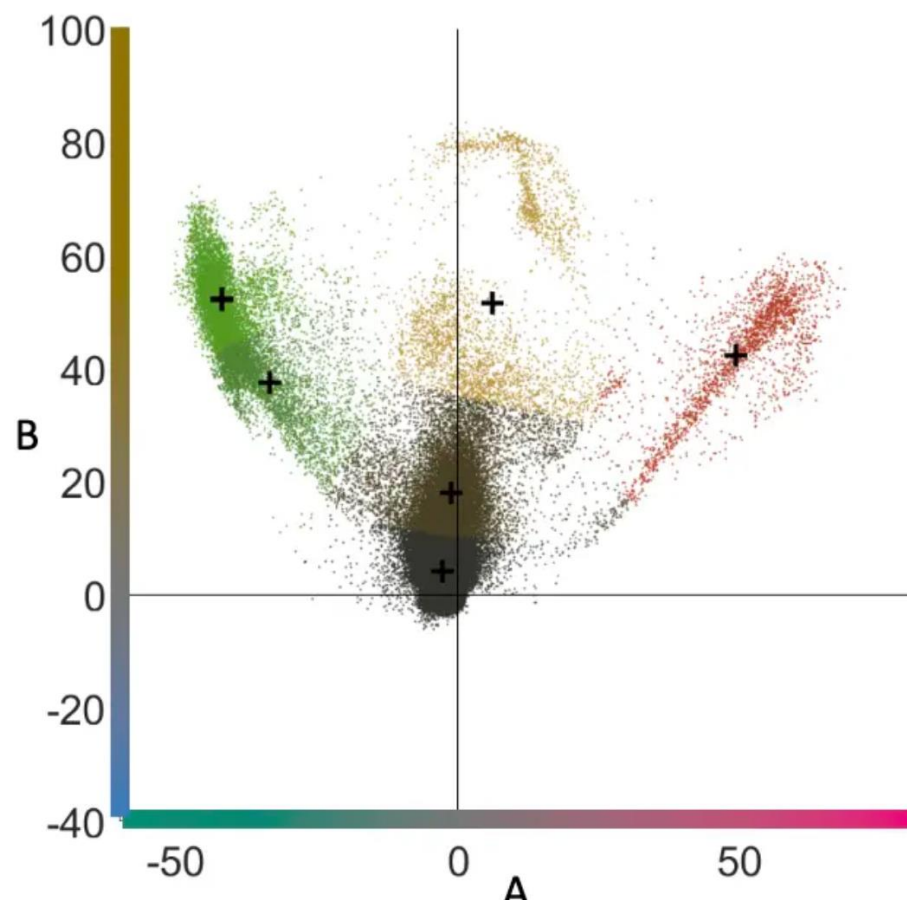
- 聚类分析 (Cluster Analysis) : 将对象的集合分组为由类似对象组成的多个类的分析过程。
- 简单来说, 就是发现数据中的自然结构, 使得同一类中的对象彼此相似, 不同类中的对象相互差异较大。
- 聚类分析不需要对分类的数目和结构作出预先假定, 属于无监督学习 (Unsupervised Learning) 。



# 聚类的应用

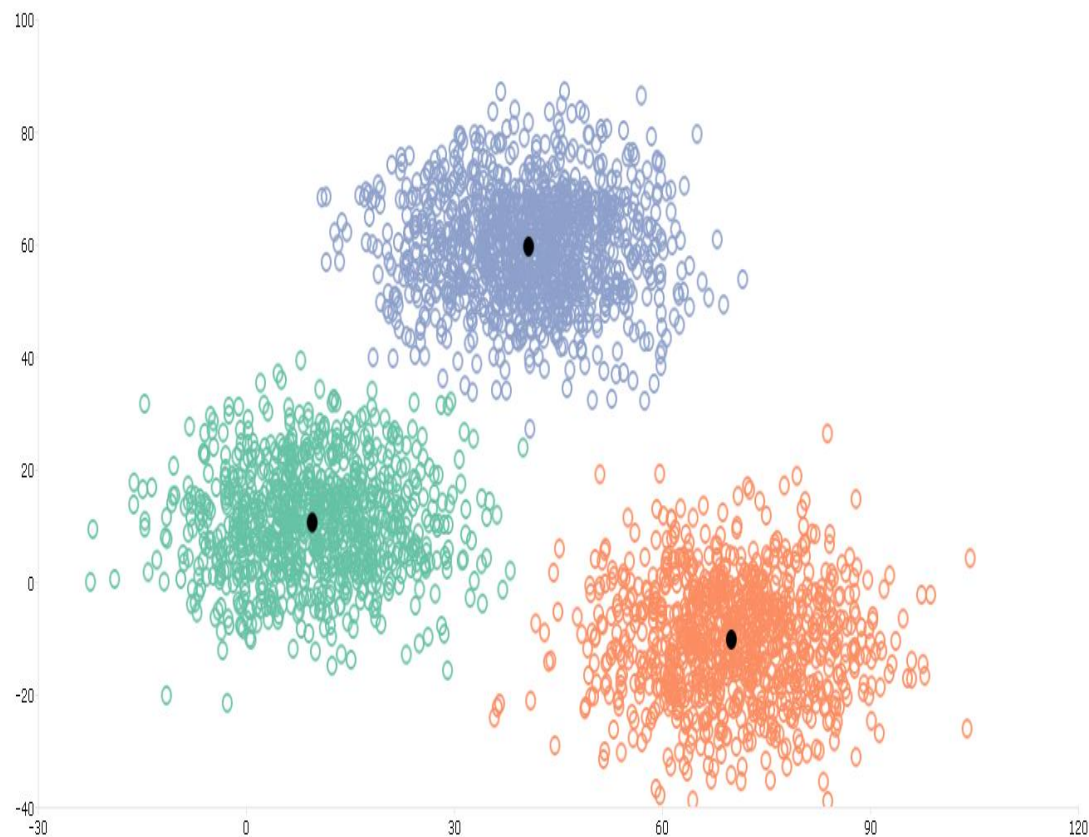


# 聚类的应用



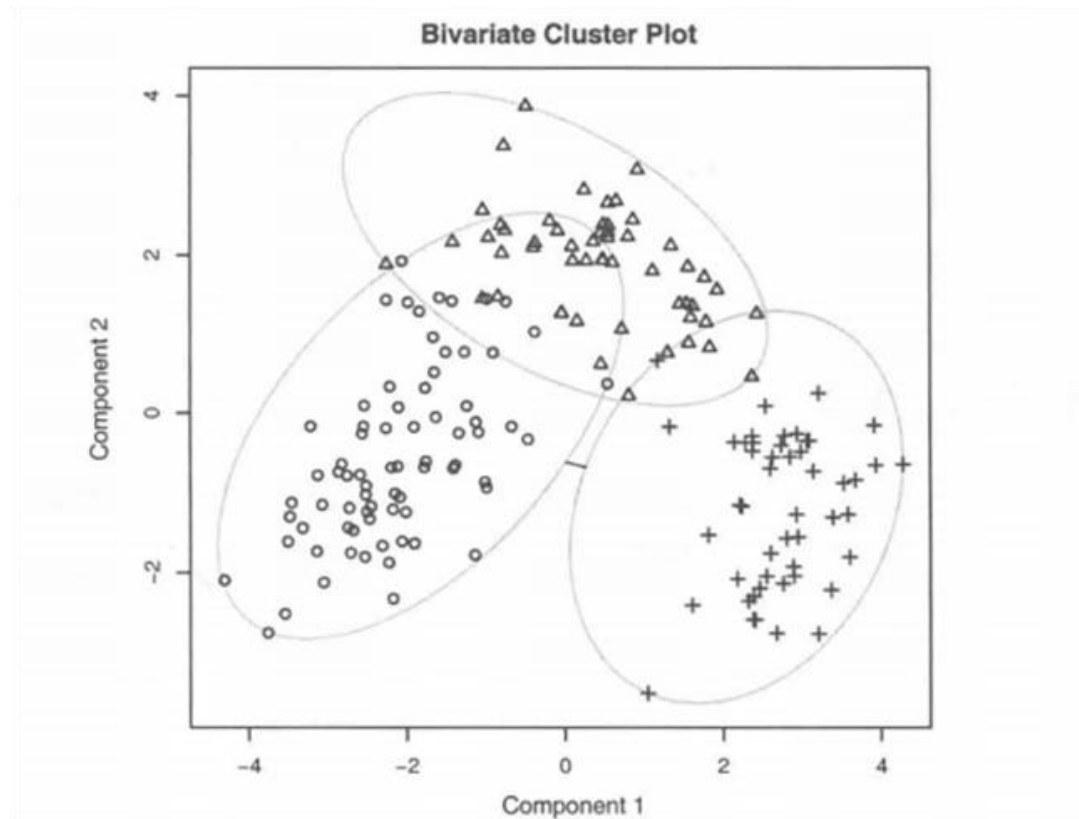
# Q型聚类：

- 也称为样本聚类（Sample Clustering）或硬聚类（Hard Clustering）。
- Q型聚类基于样本之间的相似性度量，将性质相似的样本归为一类。
- 每个样本只属于一个类，类与类之间没有交集。



# R型聚类

- 也称为变量聚类（Variable Clustering）或属性聚类。
- R 型聚类基于变量之间的相关性度量，将相关性高的变量归为一类。
- 与Q型聚类不同，R型聚类的每个样本可以属于多个分类，类与类之间可能存在交集。



# 距离

- 距离度量是用来评估数据点之间相似性或差异性的一种方法。
- 选择合适的距离度量对于聚类结果的质量和解释性至关重要。常用的距离有：
  - 欧式距离 (Euclidean Distance)
  - 曼哈顿距离 (Manhattan Distance)
  - 闵可夫斯基距离 (Minkowski Distance)
  - 汉明距离 (Hamming Distance)
  - 兰氏距离 (Lance-Williams Distance)

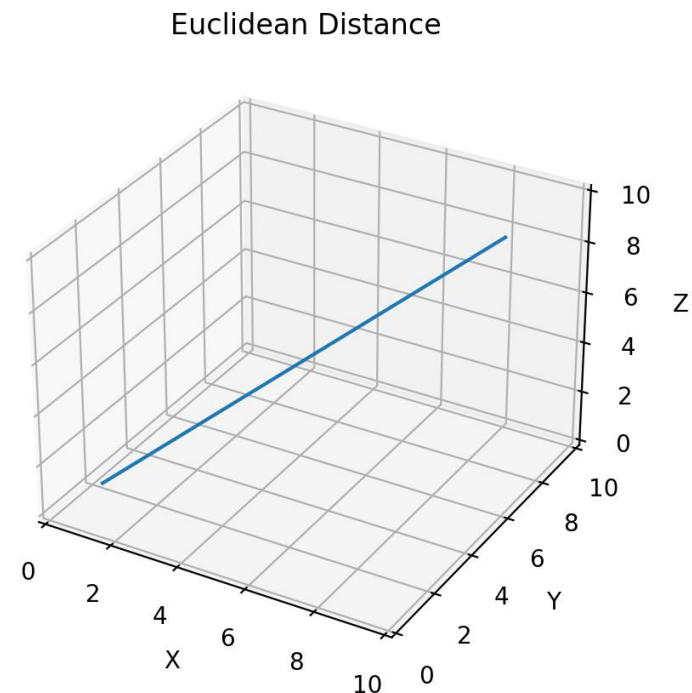
# 欧氏距离 (Euclidean Distance)

- 定义为两点之间的直线距离，在二维空间中，如果两点的坐标分别为  $(x_1, y_1)$  和  $(x_2, y_2)$ ，那么它们之间的欧氏距离  $d$  计算公式为：

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- 推广到多维空间，两点  $x$  和  $y$  之间的欧氏距离为：

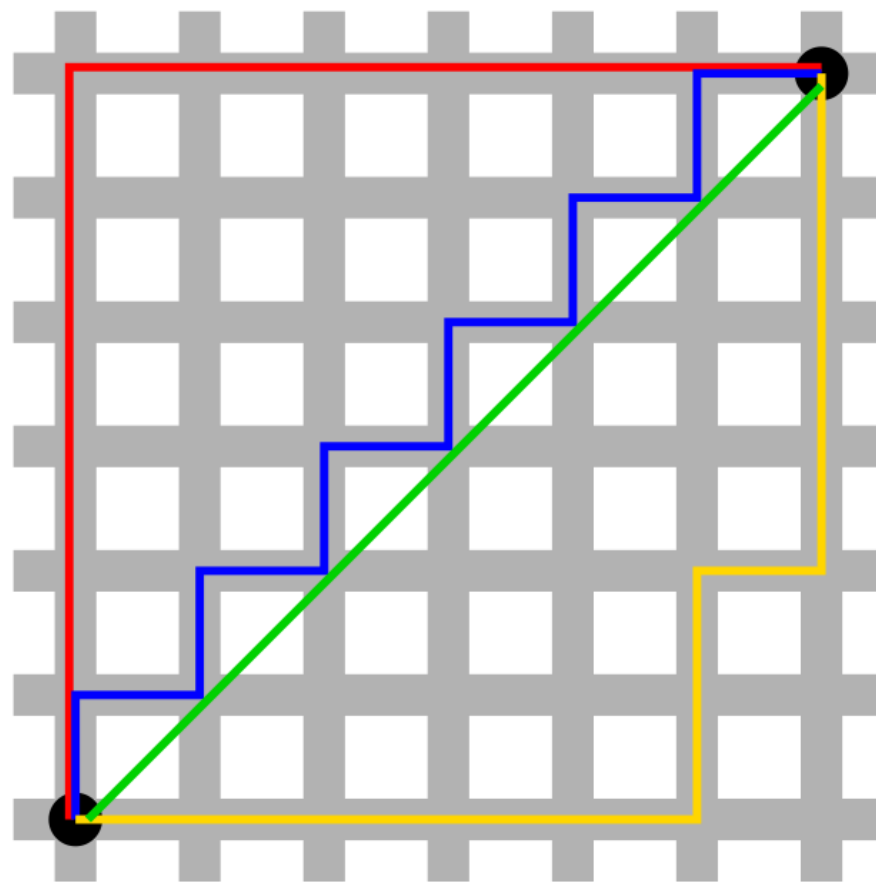
$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$



# 曼哈顿距离 (Manhattan Distance)

- 也称为城市街区距离，因为在城市中，出租车通常只能沿着街道行驶，所以这个距离度量反映了出租车行驶的最短路径。计算两点在坐标轴上的绝对轴距总和

$$d(x, y) = \sum_{i=1}^n |x_i - y_i|$$

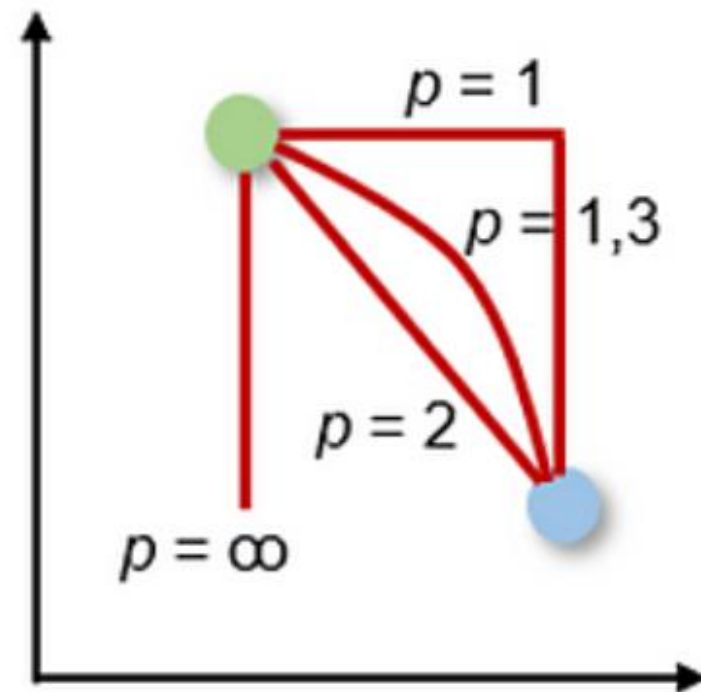


# 闵可夫斯基距离 (Minkowski Distance)

- 闵可夫斯基距离可以表示为：

$$d(x, y) = \left( \sum_{i=1}^n |x_i - y_i|^p \right)^{\frac{1}{p}}$$

- 其中  $p$  是一个参数。闵可夫斯基距离是欧氏距离和曼哈顿距离的一般形式，当  $p = 1$  时，闵可夫斯基距离就是曼哈顿距离；当  $p = 2$  时，就是欧氏距离。

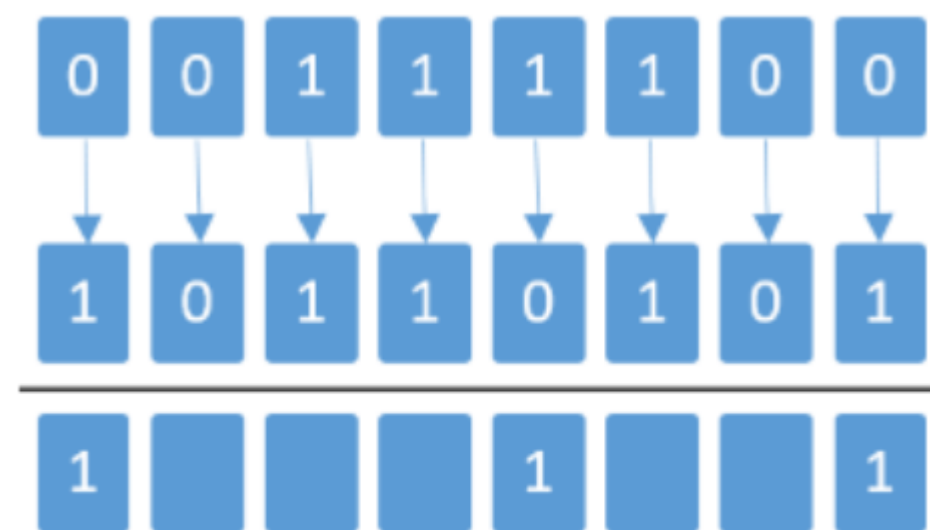


# 汉明距离 (Hamming Distance)

- 简单来说，汉明距离就是比较两个相同长度的数据序列（如二进制序列、字符序列等），统计其中对应位置元素不同的数量。

$$d(x, y) = \sum_{i=1}^n I(x_i \neq y_i)$$

- 其中 $I(\cdot)$ 是指示函数，当括号内条件成立时， $I$ 的值为1，反之为0。

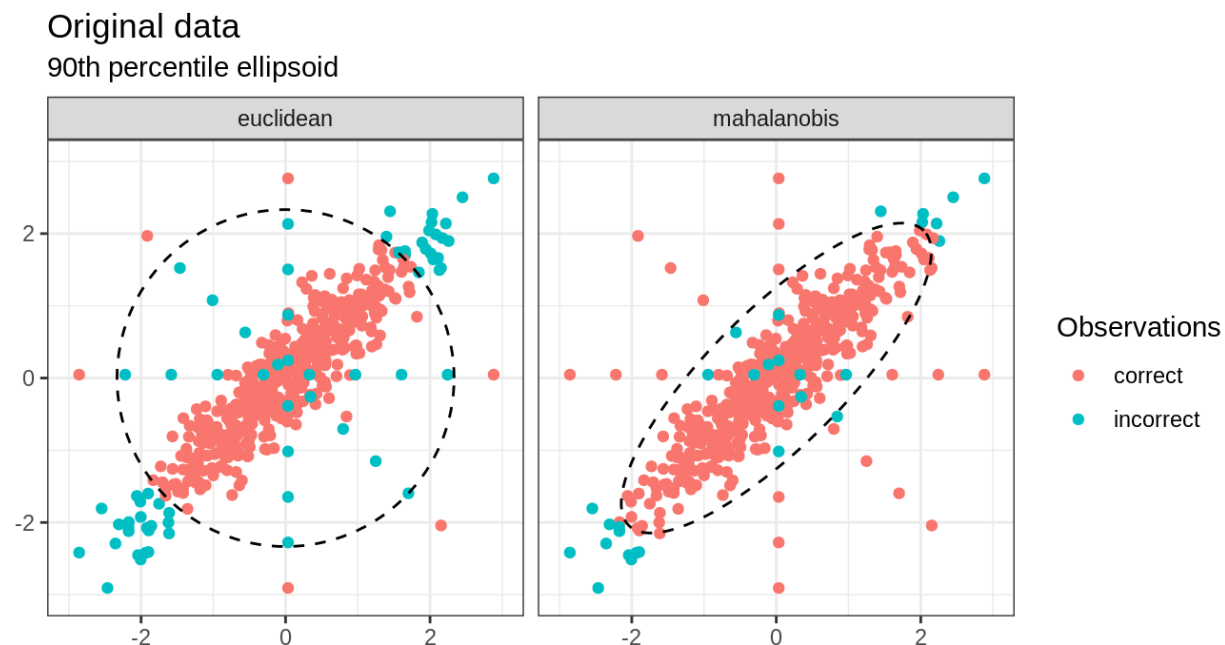


Hamming Distance = 3

# 马氏距离 (Mahalanobis Distance)

- 马氏距离考虑了数据的协方差结构的距离度量，用于衡量一个样本点与一个分布中心之间的距离，或者两个样本点在某个分布下的相对距离。其计算公式为：

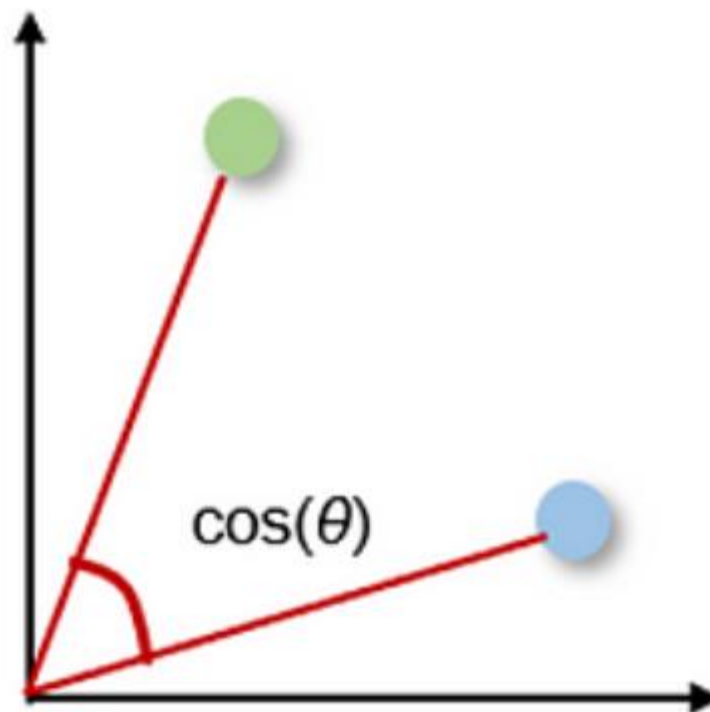
$$d^2(x, y) = (x - y)^T \Sigma^{-1} (x - y)$$



# 余弦相似度 (Cosine Similarity)

- 通过计算两个向量的夹角余弦值来评估它们的相似度，对于两个非零向量  $x$  和  $y$ ，余弦相似度  $s$  定义为：

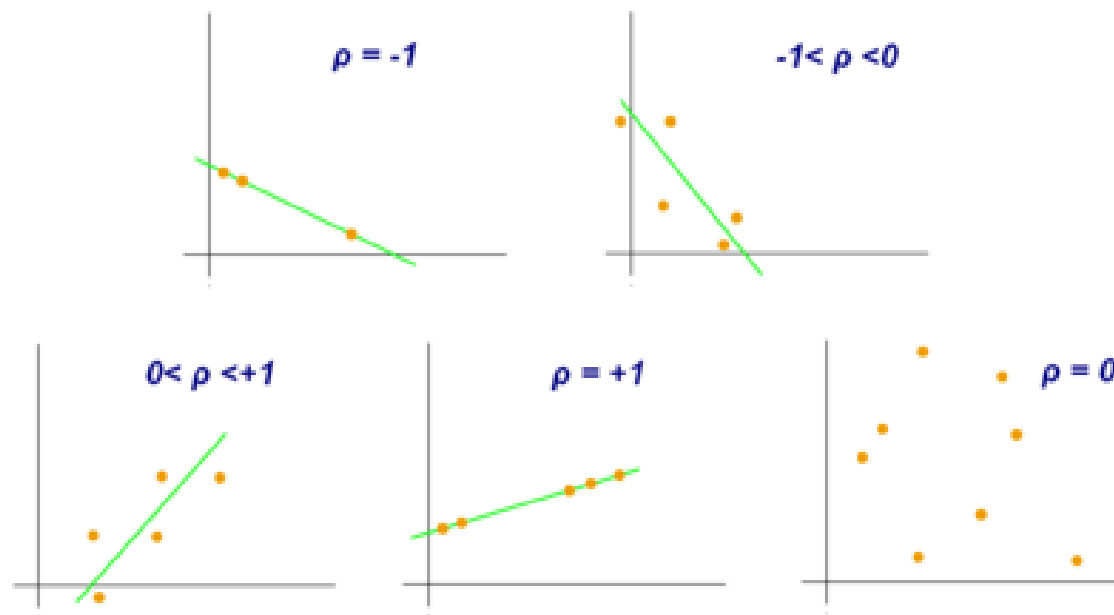
$$\begin{aligned} s(x, y) &= \frac{x \cdot y}{|x||y|} \\ &= \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}} \end{aligned}$$



# 相关系数 (Pearson Correlation Coefficient)

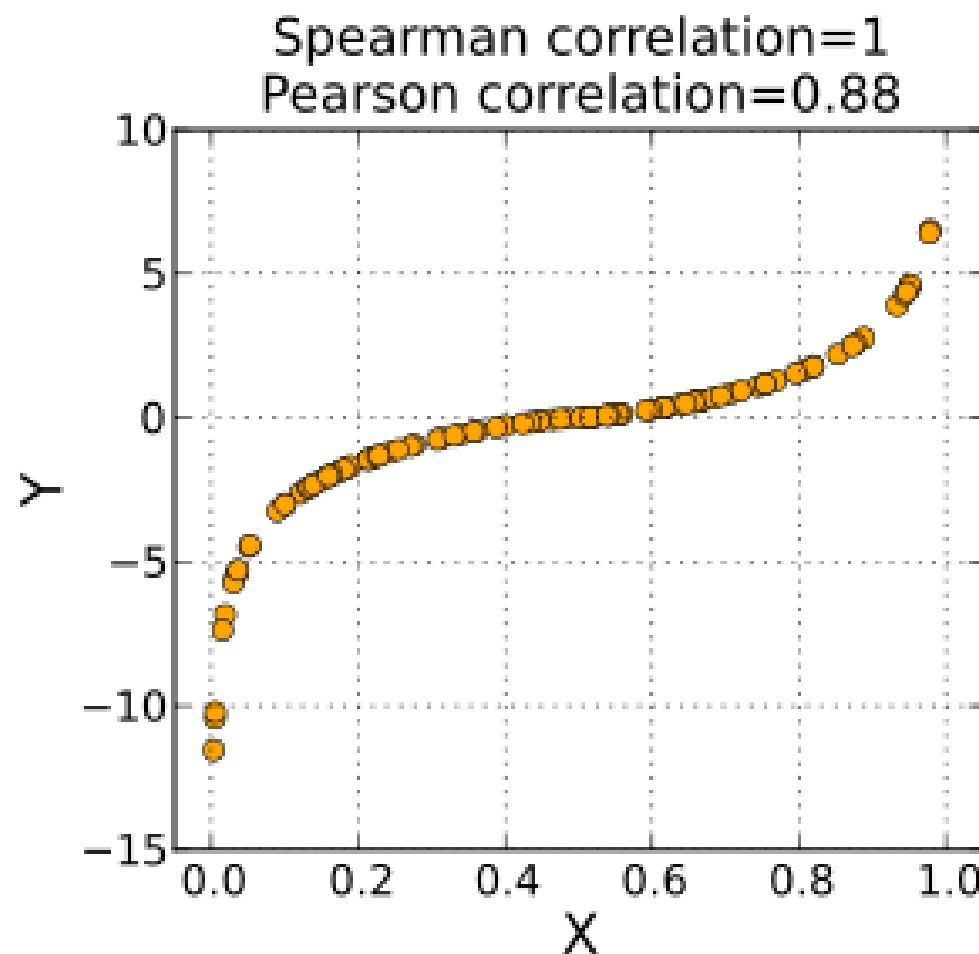
- 最常用的相关系数是皮尔逊相关系数 (Pearson Correlation Coefficient)，计算公式为：

$$r = \frac{\sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}}$$



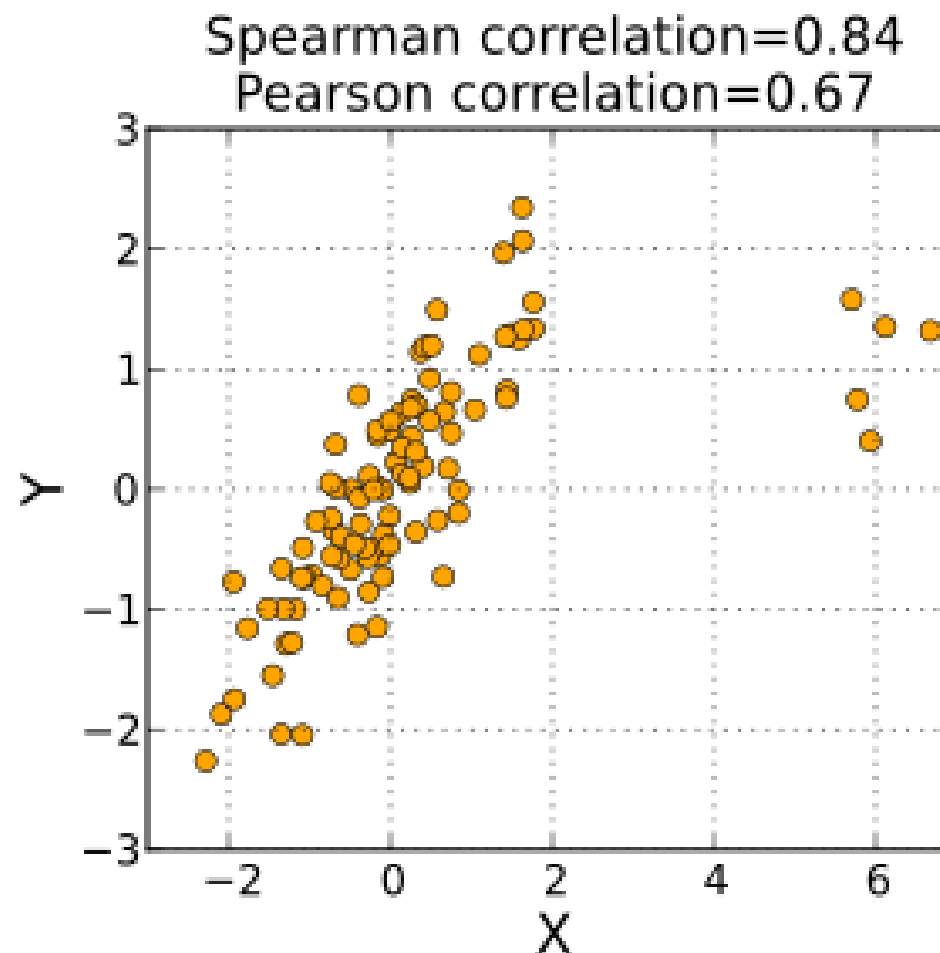
# 秩相关系数 (Spearman's Rank Correlation Coefficient)

- 基于变量的秩次 (排序后的位置) 而不是变量的实际数值来计算相关性。
- 先分别对两个变量 $X$ 和 $Y$ 的观测值进行排序, 得到它们的秩次 $R(X)$ 和 $R(Y)$ 。然后计算秩次之间的差异 $d_i = R(X_i) - R(Y_i)$ 。
- 秩相关系数 $r_s$ 的计算公式为 $r_s = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2-1)}$ , 其中 $n$ 是观测值的数量。



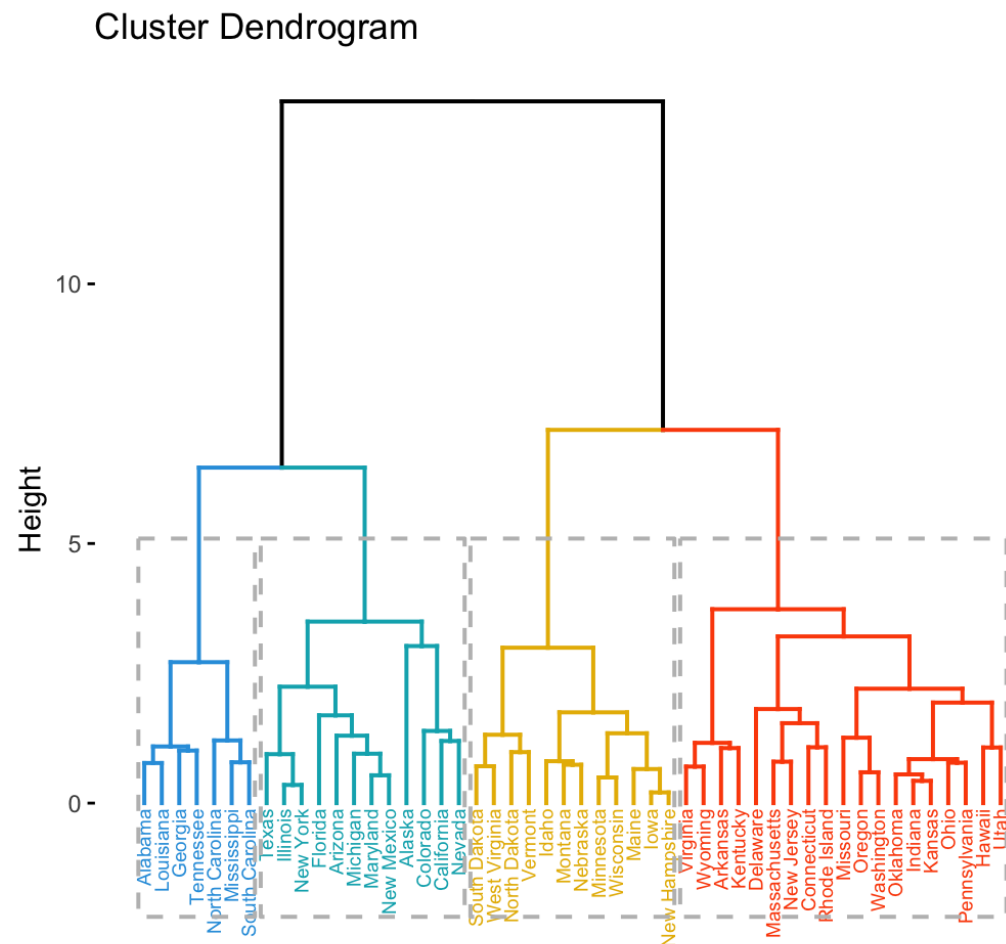
# 秩相关系数 (Spearman's Rank Correlation Coefficient)

- 基于变量的秩次（排序后的位置）而不是变量的实际数值来计算相关性。
- 先分别对两个变量 $X$ 和 $Y$ 的观测值进行排序，得到它们的秩次 $R(X)$ 和 $R(Y)$ 。然后计算秩次之间的差异 $d_i = R(X_i) - R(Y_i)$ 。
- 秩相关系数 $r_s$ 的计算公式为 $r_s = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2-1)}$ ，其中 $n$ 是观测值的数量。



# 层次聚类法

- 层次聚类法（Hierarchical Clustering），又称为系统聚类法。它不需要事先指定聚类的数量，而是生成一个由层次结构组成的聚类树（Dendrogram），过不断地合并或者分裂数据子集来构建聚类层次，最终形成一个树形的聚类结构。这个树可以刻画聚类的数量和聚类的层次。

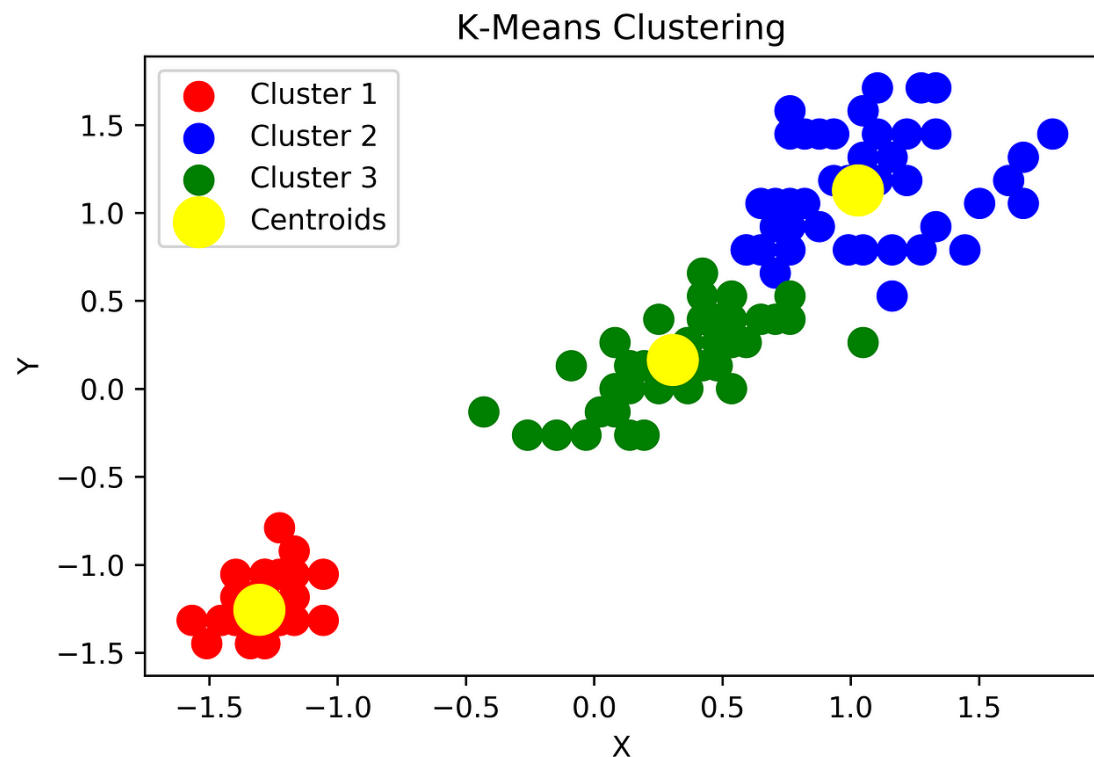


# 层次聚类法

- 在层次聚类法中，类的合并通常基于以下距离度量方法之一：
- **最短距离法（Single Linkage）**：根据两个类中最近的成员之间的距离来合并类。
- **最长距离法（Complete Linkage）**：根据两个类中最远成员之间的距离来合并类。
- **类平均距离法（Average Linkage）**：使用两个类所有成员之间距离的平均值来合并类。
- **中心距离法（Centroid Linkage）**：使用两个类的质心之间的距离来合并类。

# K-means聚类方法

- K-means聚类法是一种流行的迭代聚类算法，用于将数据分为多个簇，使得簇内的点尽可能相似，而簇间的点尽可能不同。
- 基本思想是通过迭代的方将数据点分配到最近的簇中，并更新簇中心，直到达到预设的迭代次数。
- K-means聚类方法是一种效率非常高的聚类方法，但需要事先确定簇的数量。如果K值选择不当，会导致聚类结果不符合实际情况。



# K-means聚类方法

- K-means聚类法的基本步骤如下：

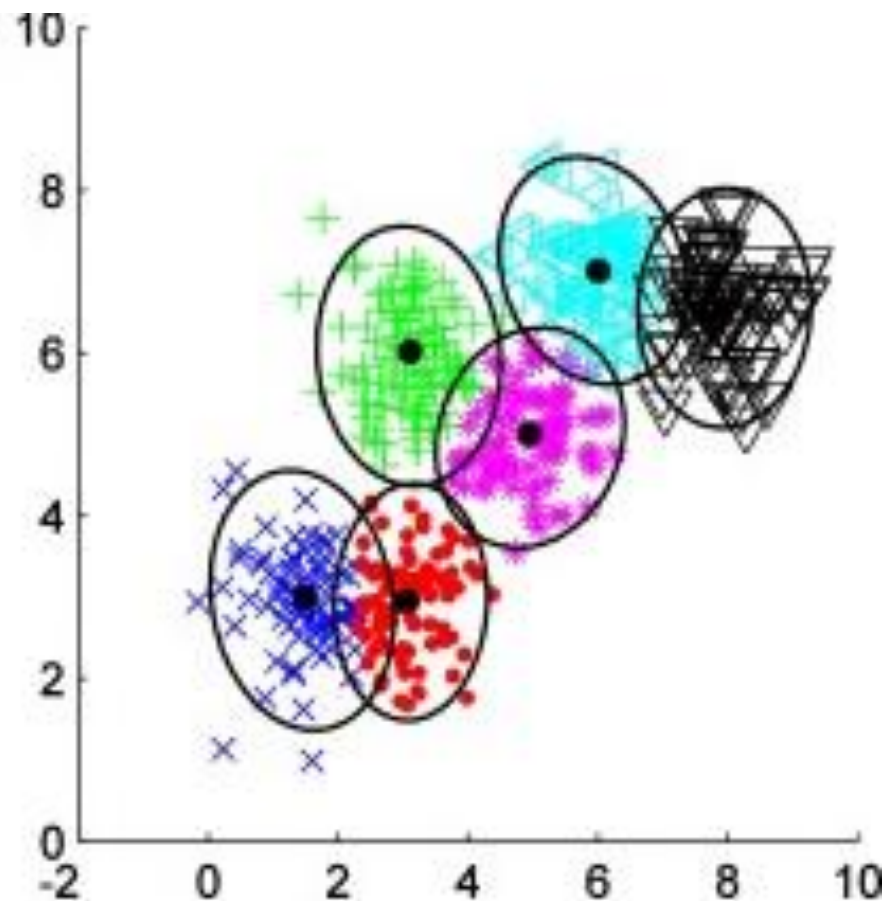
- 1.初始化：** 选择一个初始的类数目 $k$ ，然后随机选择 $k$ 个数据点作为初始的类中心（质心）。
- 2.分配：** 对于数据集中的每个点，计算它与每个类中心的距离，并将其分配到最近的类中心，形成 $k$ 个类。
- 3.更新：** 重新计算每个类的中心，通常是类内所有点的均值。
- 4.迭代：** 重复步骤2和3，直到满足以下停止条件之一：
  1. 类中心不再显著变化，即连续两次迭代的类中心变化量小于某个预设的阈值。
  2. 达到预设的迭代次数

# K-means聚类方法

- 优点：
  - 简单、直观，易于理解和实现。
  - 计算效率高，适合处理大型数据集。
- 缺点
  - 对初始类中心敏感，可能导致局部最优解。
  - 需要预先指定类的数目 $k$ ，但在很多情况下 $k$ 并不容易选择。
  - 对噪声和异常值敏感，可能会影响聚类结果。
  - 只能发现球形的类，对于非球形分布的数据可能不是最佳选择。

# 模糊C均值聚类法

- 模糊C均值聚类法 (Fuzzy C-Means Clustering, 简称FCM) 是一种基于模糊划分的聚类算法。模糊C均值聚类法允许每个数据点以一定的隶属度属于多个聚类。
- 模糊C均值聚类法同样需要预先确定聚类的数目, 通过迭代的方式来优化聚类结果, 在迭代过程中都涉及到对聚类中心的更新。



# 模糊C均值聚类法

- 1.初始化：**选择类的数目  $C$ ，并随机初始化类中心。
- 2.隶属度计算：**对于每个数据点，计算它到每个类中心的隶属度，隶属度的值介于0和1之间。隶属度越高，表示数据点属于特定类的可能性越大。
- 3.类中心更新：**根据数据点的隶属度，更新每个类的中心。
- 4.迭代：**重复步骤2和3，直到满足停止条件，如类中心的变化小于某个阈值或达到预设的迭代次数。
- 5.终止：**当满足停止条件后，聚类过程结束。

# 模糊C均值聚类法

- 优点：
  - 由于该方法允许数据点同时属于多个聚类，对于非球形、有重叠部分的聚类数据可能有更好的适应性。
- 缺点：
  - 同样需要预先指定类的数目 $C$ 。
  - 对初始类中心敏感。
  - 计算复杂度比k-mean高，在处理大型数据集时速度较慢。

# 总结聚类分析

- Q型聚类法：每个样本只属于一个类，类与类之间没有交集。
  - 层次聚类法
  - k-means聚类法
- R型聚类法：每个样本可以属于多个分类，类与类之间可能存在交集。
  - 模糊C均值聚类法

# 练习

- 使用k-means方法分割一张图像

# 判别分析

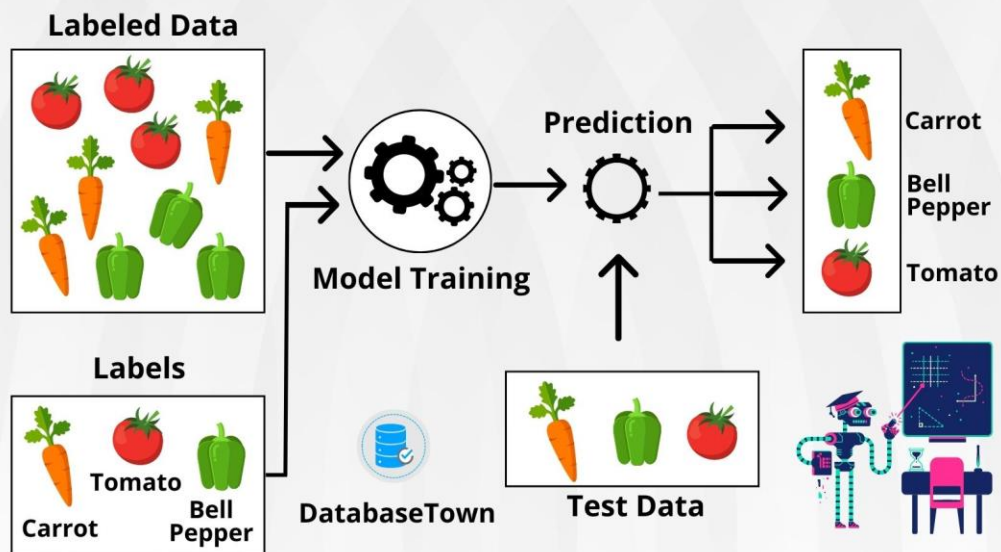
# 分类和聚类

- 分类（Classification）和聚类（Clustering）是数据挖掘和机器学习中两种常见的数据分析技术，它们存在一些关键的区别：
- 目的：
  - **分类**：是一种监督学习任务，它是基于已有的标记数据（训练数据）来构建模型，将新的数据划分到已知的类别中。目的是通过分析训练数据集中的已知类别，建立一个模型，然后用这个模型预测新数据点的类别。
  - **聚类**：是一种无监督学习任务，它是在没有预先定义类别标签的情况下，根据数据自身的相似性将数据对象划分为不同的组。目的是将数据集中的样本根据相似性分组，使得同一组内的样本尽可能相似，而不同组之间的样本尽可能不同。

# 分类和聚类

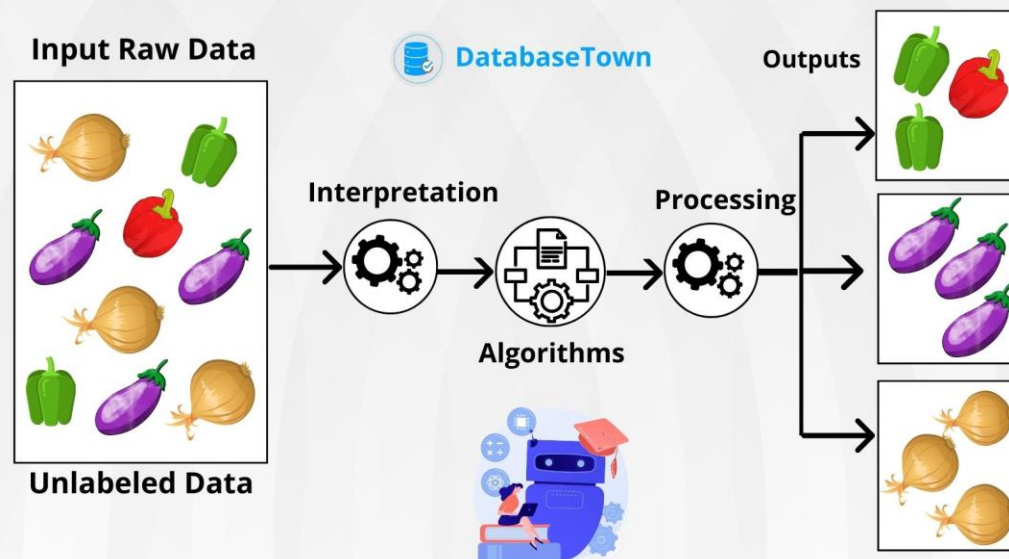
## SUPERVISED LEARNING

Supervised machine learning is a branch of artificial intelligence that focuses on training models to make predictions or decisions based on labeled training data.



## UNSUPERVISED LEARNING

Unsupervised learning is a type of machine learning where the algorithm learns from unlabeled data without any predefined outputs or target variables.



# 混淆矩阵(Confusion Matrix)

- 展示分类模型预测结果与实际结果的比较情况。

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

- 真正例 (TP) : 实际为正类, 且模型预测为正类的样本数量。
- 假负例 (FN) : 实际为正类, 但模型预测为负类的样本数量。
- 假正例 (FP) : 实际为负类, 而模型预测为正类的样本数量。
- 真负例 (TN) : 实际为负类, 并且模型预测为负类的样本数量。

# 混淆矩阵(Confusion Matrix)

- 准确率 (Accuracy) :

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN}$$

- 精确率 (Precision) :

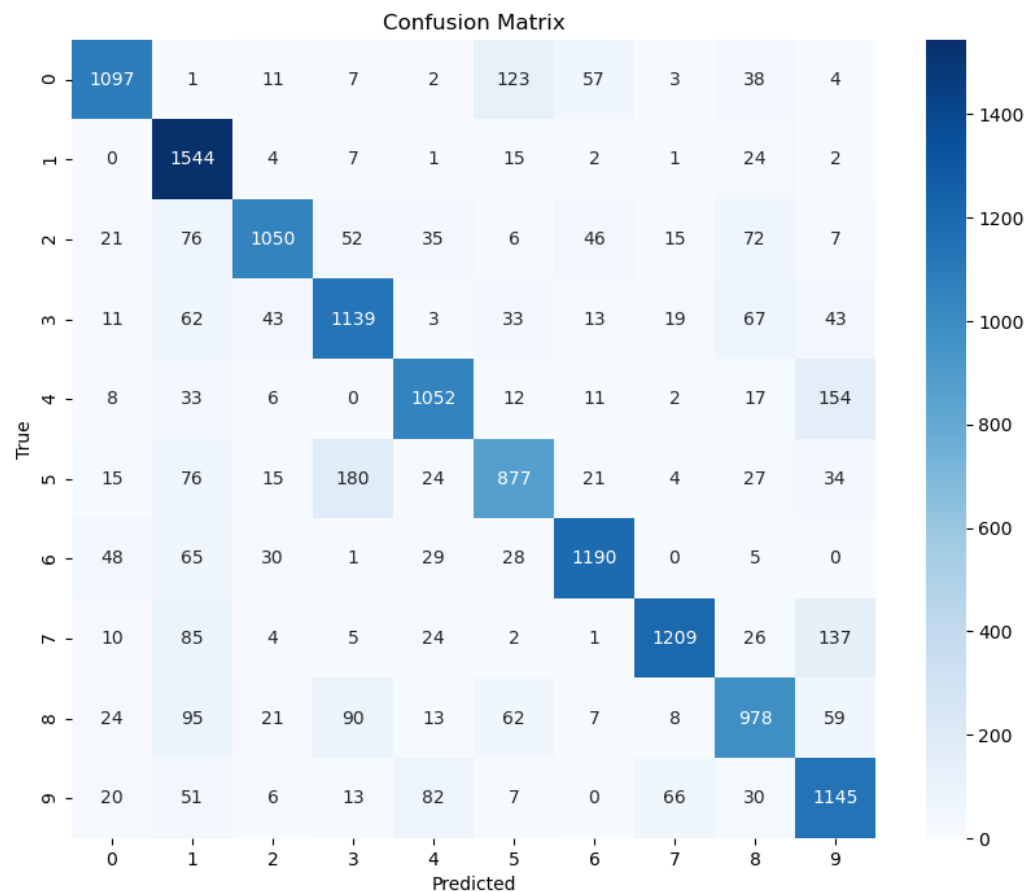
$$\text{Precision} = \frac{TP}{TP + FP}$$

- 召回率 (Recall) :

$$\text{Recall} = \frac{TP}{TP + FN}$$

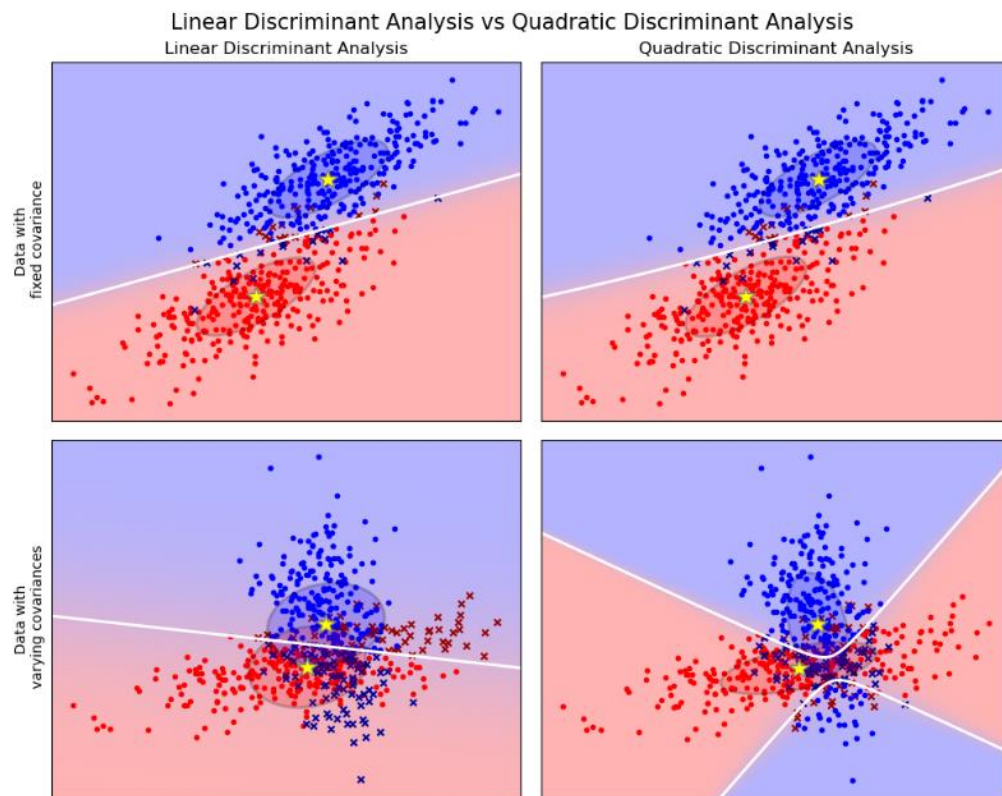
- F1 - Score:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$



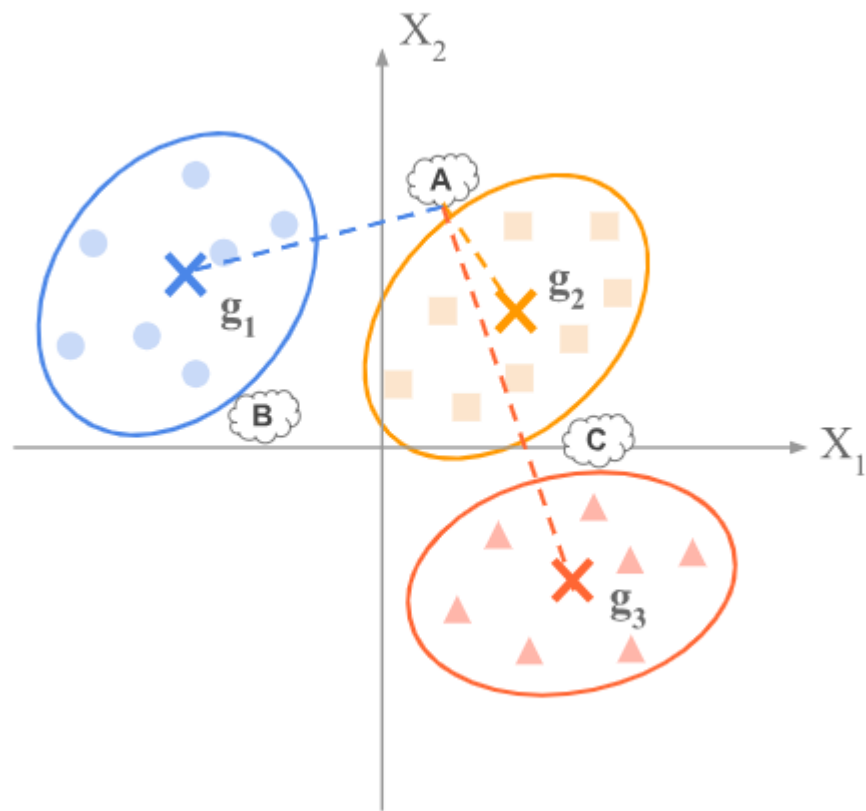
# 判别分析

- 判别分析（Discriminant Analysis）用于研究两个或多个已知类别的对象群之间的差异，并通过这些差异来预测新对象的类别。
- 具体方法是根据已知分类的样本数据构建一个判别函数，然后利用这个判别函数对未知分类的样本进行分类。



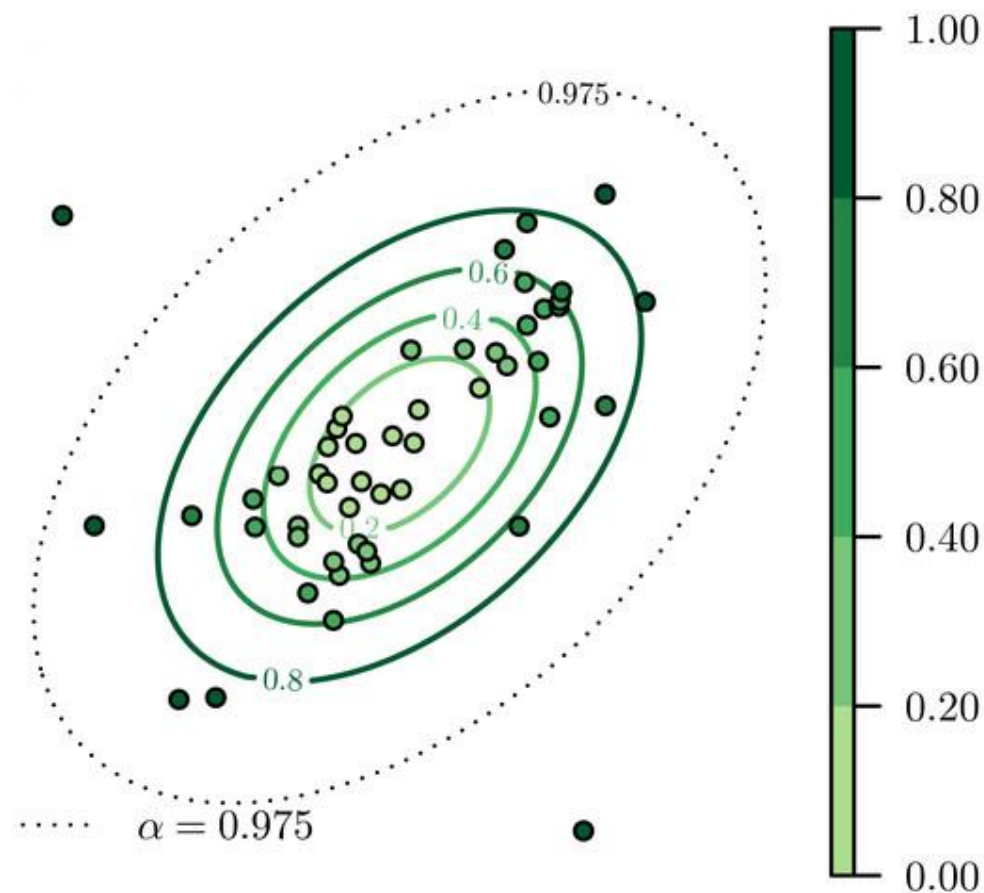
# 距离判别 (Distance Discriminant)

- 基于样本到各个类别中心的距离来进行判别。它假设每个类别数据的分布是多元正态分布，并且各个类别具有相同的协方差矩阵。
- 有两类样本 A 和 B，分别计算新样本点到类别 A 的中心和类别 B 的中心的马氏距离，新样本点距离哪个类别中心更近，就将其判归到哪一类。
- 这种方法简单直观，适用于类别具有不同均值但相同协方差矩阵的情况。



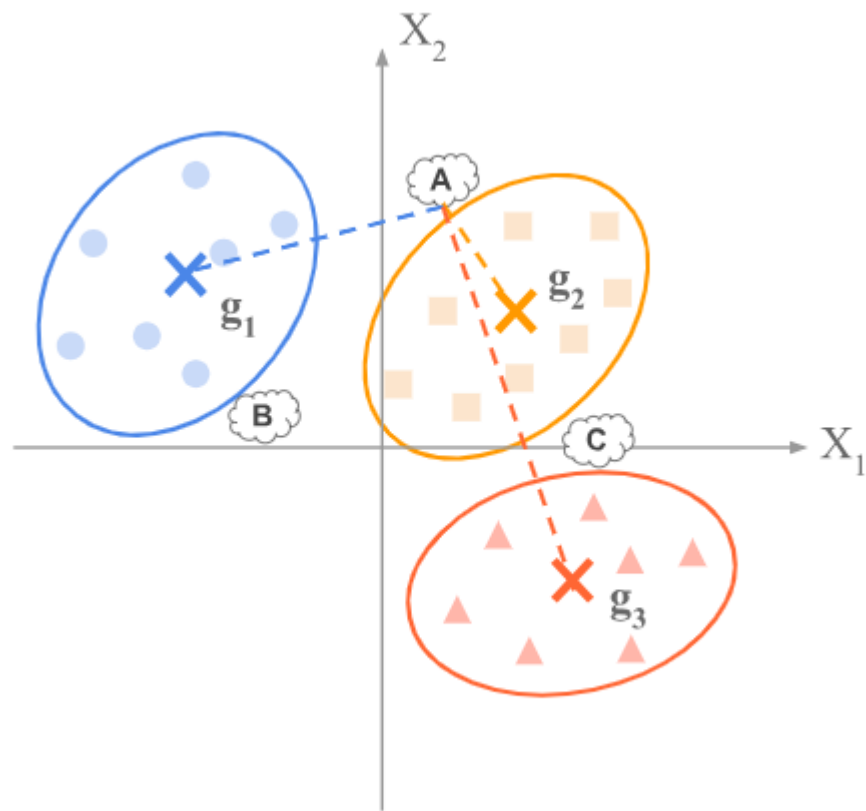
# 马氏距离和等高线

- 马氏距离通过协方差矩阵的逆对数据进行了标准化处理，使得不同尺度和分布的数据可以进行比较。
- 在二维正态分布中，等高线（或等概率线）通常是椭圆形的。马氏距离可以看作是从一个点到分布中心的最短距离，这个距离沿着椭圆的主轴方向。椭圆的形状和方向由协方差矩阵决定。



# 距离判别 (Distance Discriminant)

- 距离判别：基于马氏距离对新样本所属的类别进行判断。
- 两点之间的马氏距离为：
$$d^2(x, y) = (x - y)^T \Sigma^{-1} (x - y)$$
- 新的样本 $x$ 到总体 $G$ 的马氏距离为
$$d^2(x, G) = (x - \mu)^T \Sigma^{-1} (x - \mu)$$



## 示例：两总体距离判别

- 设有两个  $p$  维总体  $G_1$  和  $G_2$ , 分布的均值向量分别为  $\mu_1, \mu_2$ , 协方差矩阵分别为  $\Sigma_1 > 0, \Sigma_2 > 0$ 。现有一未知类别的样品, 记为  $x$ , 试判断  $x$  的归属, 则有以下判别规则
- 如果  $d(X, G_1) < d(X, G_2)$ , 则  $X$  属于  $G_1$ ;
- 如果  $d(X, G_1) > d(X, G_2)$ , 则  $X$  属于  $G_2$ ;
- 如果  $d(X, G_1) = d(X, G_2)$ , 则无法判断  $X$  属于哪个总体。

# 贝叶斯判别 (Bayes Discriminant)

- 距离判别没有考虑人们对研究对象已有的认知，而这种已有的认知可能会对判别的结果产生影响。贝叶斯 (Bayes) 判别则用一个先验概率来描述这种已有的认知，然后通过样本来修正先验概率，得到后验概率，最后基于后验概率进行判别。
- 贝叶斯判别基于贝叶斯定理，它考虑了先验概率和数据的概率密度函数 (PDF)。在给定一个新样本的情况下，贝叶斯判别会计算该样本属于每个类别的后验概率，并将其归类为具有最高后验概率的类别。

# 朴素贝叶斯分类器

- 朴素贝叶斯分类器 (Naive Bayes Classifier) 是一种基于贝叶斯定理的概率分类器，它基于一个非常强的假设：特征之间相互独立。由于这个假设在现实世界中往往不成立，因此被称为“朴素”的。
- 这个“朴素”的假设虽然在实际情况中可能不完全成立，但在很多场景下却能取得很好的分类效果。例如在文本分类和垃圾邮件过滤等领域。

# 朴素贝叶斯分类器

- 对于一个具有特征向量  $x = (x_1, x_2, \dots, x_n)$  的样本，朴素贝叶斯分类器判断它属于类别  $y$  的概率，根据贝叶斯定理有

$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

- 由于  $P(x)$  对于所有类别  $y$  来说是相同的常数，所以在比较不同类别概率大小时可以忽略。因此我们只看  $P(x|y)P(y)$ 。
- 根据朴素贝叶斯的独立性假设，

$$P(x|y) = P(x_1, x_2, \dots, x_n|y) = \prod_{i=1}^n P(x_i|y)$$

- 这意味着可以将联合概率  $P(x|y)$  分解为各个特征的条件概率的乘积。
- 先验概率  $P(y)$  和条件概率  $P(x_i|y)$  可以通过训练数据中各类别的比例来估计。

# 贝叶斯判别 (Bayes Discriminant)

- **似然比：** 类条件概率密度之比。设样本 $x$ ，有两个类别 $G_1$ 和 $G_2$ ，似然比 $\lambda(x) = \frac{P(G_2|x)}{P(G_1|x)}$ 。它表示在给定样本 $x$ 的情况下，类别 $G_2$ 相对于类别 $G_1$ 的相对可能性。
- **判决阈值：** 用于做出最终判别决策的一个界限值。判决阈值可以用先验概率来计算，而且先验概率在确定判决阈值的过程中起着重要的作用。
- 通过设定一个判决阈值 $t = \frac{P(G_2)}{P(G_1)} \alpha$ ，这里的 $\alpha$ 是错误分类的代价，如果 $\lambda(x) > t$ ，则将样本判归为 $G_2$ ；否则判归为 $G_1$ 。

# 示例： 贝叶斯判别

- 对某一地震高发区进行统计，地震以 $G_1$ 类表示，正常以 $G_2$ 类表示统计的时间区间内，每周发生地震的概率为20%，即 $P(G_1) = 0.2$ ，当然 $P(G_2) = 1 - 0.2 = 0.8$  在任意一周，要判断该地区是否会有地震发生。
- 显然，因为 $P(G_2) > P(G_1)$ ，只能说是正常的可能性大。如要进行判断，只能其它观察现象来实现。通常地震与生物异常反应之间有一定的联系。

# 示例： 贝叶斯判别

- 某公司开发一内部电子邮箱垃圾邮件过滤系统，垃圾邮件以 $G_1$ 类表示，正常邮件以 $G_2$ 类表示。统计的邮件样本内，垃圾邮件的概率为20%，即 $P(G_1) = 0.2$ ，当然 $P(G_2) = 1 - 0.2 = 0.8$  在任意一周，要判断该地区是否会有地震发生。
- 显然，因为 $P(G_2) > P(G_1)$ ，只能说是正常的可能性大。如要进行判断，只能其它观察现象来实现。通常地震与生物异常反应之间有一定的联系。

# 示例：贝叶斯判别

- 我们要构建一个简单的贝叶斯分类器来判断一封电子邮件是垃圾邮件还是正常邮件。首先，我们把邮件内容看作是由单词组成的集合，这些单词就是我们用于分类的特征。假设我们只考虑三个单词作为特征：“促销”“免费”“工作”。这里 $x$ 为一维特征，且只有 $x = \text{“垃圾”}$ 和 $x = \text{“正常”}$ 两种结果。假设根据一批已经标记好的邮件作为训练数据，发现这种方法有以下统计结果：
- 在垃圾邮件中，“促销”出现的概率= 0.6，即 $p(x = \text{垃圾} | G_1) = 0.6$
- 在垃圾邮件中，“促销”不出现的概率= 0.4，即 $p(x = \text{正常} | G_1) = 0.4$
- 在正常邮件中，“促销”出现的概率= 0.1，即 $p(x = \text{垃圾} | G_2) = 0.1$
- 在正常邮件中，“促销”不出现的概率= 0.9，即 $p(x = \text{正常} | G_2) = 0.9$
- 收到一封新邮件，“促销”出现了，这封邮件是垃圾邮件的概率为多少，即求 $P(G_1 | x = \text{垃圾}) = ?$

# 示例： 贝叶斯判别

# 示例： 贝叶斯判别

- 已知：  $P(\omega_1) = 0.2$ ,  $P(\omega_2) = 0.8$ ,
- $p(x = \text{垃圾}|\omega_1) = 0.6$ ,  $p(x = \text{正常}|\omega_1) = 0.4$ ,
- $p(x = \text{垃圾}|\omega_2) = 0.1$ ,  $p(x = \text{正常}|\omega_2) = 0.9$
- 利用贝叶斯公式, 有:

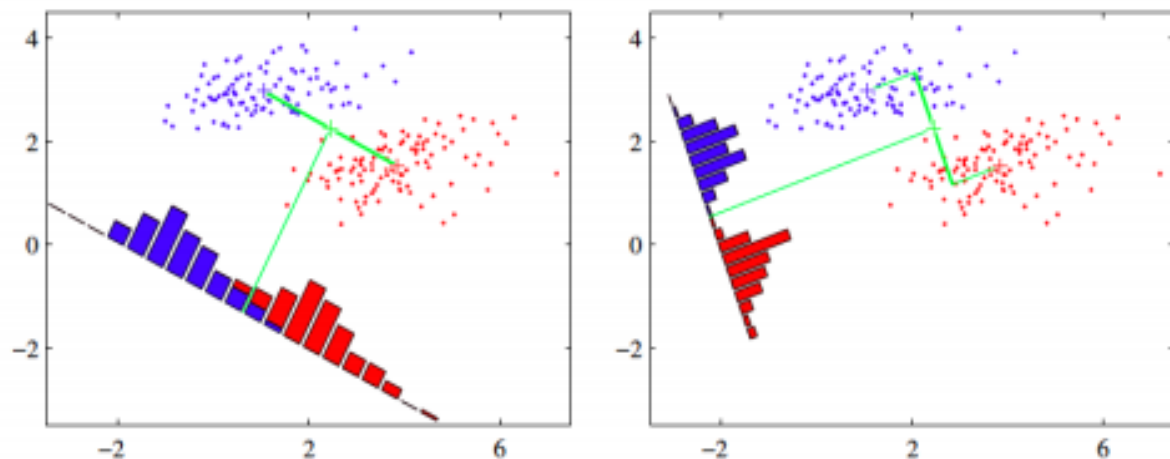
$$\begin{aligned} \bullet P(\omega_1|x = \text{垃圾}) &= \frac{p(x = \text{垃圾}|\omega_1)P(\omega_1)}{p(x = \text{垃圾})} = \frac{p(x = \text{垃圾}|\omega_1)P(\omega_1)}{p(x = \text{垃圾}|\omega_1)P(\omega_1) + p(x = \text{垃圾}|\omega_2)P(\omega_2)} \\ &= \frac{0.6 \times 0.2}{0.6 \times 0.2 + 0.1 \times 0.8} = 0.6 \end{aligned}$$

$$\bullet \text{似然比: } l_{12} = \frac{p(\omega_1|x = \text{垃圾})}{p(\omega_2|x = \text{垃圾})} = \frac{0.6}{0.1} = 6$$

$$\bullet \text{若令 } \alpha = 1.3, \text{ 判决阈值: } \theta_{21} = \frac{P(\omega_2)}{P(\omega_1)} \alpha = \frac{0.8}{0.2} = 5.2$$

# Fisher判别 (Linear Discriminant Analysis)

- Fisher判别 (Fisher's Linear Discriminant Analysis, LDA) 是寻找一个最佳的投影方向，将高维数据投影到低维空间（通常是一维），使得投影后不同类别之间的样本尽可能地分开，同时同一类别内的样本尽可能地聚集。
- LDA的核心思想是寻找一个线性组合的特征，使得不同类别的数据在这个组合特征上具有最大的类别间差异（组间方差）和最小的类别内差异（组内方差）。

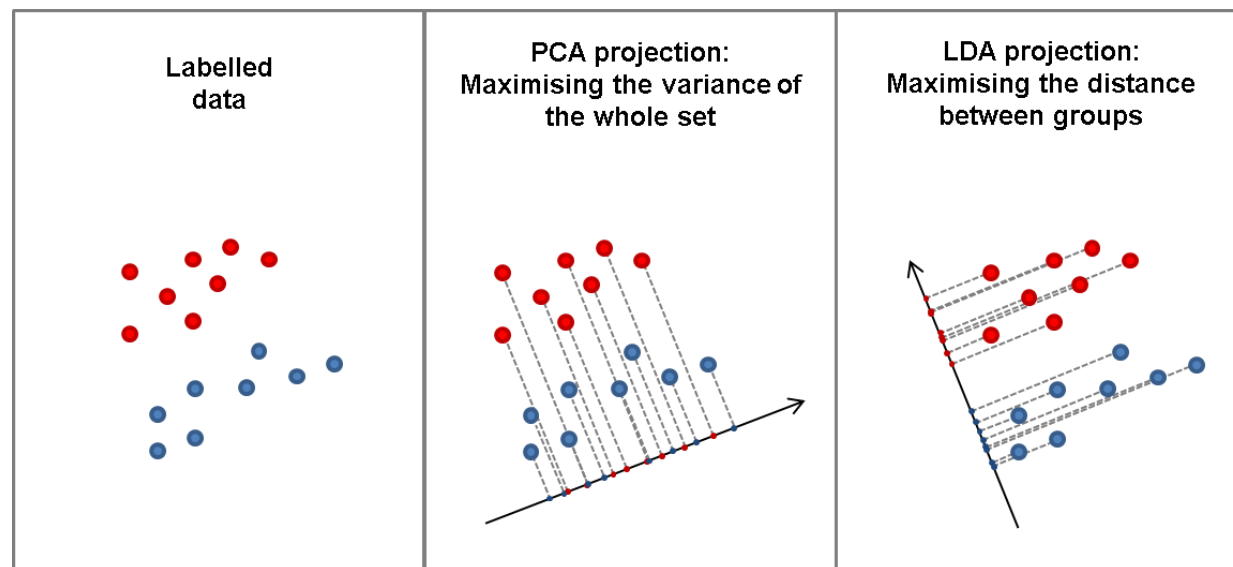


# Fisher判别

- 设样本数据为 $x_i (i = 1, 2, \dots, n)$ ，属于 $k$ 个不同的类别 $C_j (j = 1, 2, \dots, k)$ 。我们构建一个线性判别函数 $y = w^T x$ ，其中 $w$ 是我们要确定的判别系数向量， $x$ 是样本向量。
- **计算类间散度矩阵**： $S_b = \sum_{j=1}^k n_j (\mu_j - \mu)(\mu_j - \mu)^T$ ，其中 $n_j$ 是类别 $C_j$ 中的样本数量， $\mu_j$ 是类别 $C_j$ 的样本均值向量， $\mu$ 是所有样本的总均值向量。 $S_b$ 衡量不同类别中心之间的离散程度。
- **计算类内散度矩阵**： $S_w = \sum_{j=1}^k \sum_{x_i \in C_j} (x_i - \mu_j)(x_i - \mu_j)^T$ ，衡量每个类别内部样本的离散程度。
- **确定判别向量 $w$** ：优化目标是最大化 $J(w) = \frac{w^T S_b w}{w^T S_w w}$ ，通过对 $J(w)$ 求导并令其等于零，可以得到 $S_b w = \lambda S_w w$ （其中 $\lambda$ 是一个常数），求解该广义特征值问题可以得到判别系数向量 $w$ 。

# Fisher判别与PCA

- **LDA:** 找到一个或多个投影方向，使得在投影后的空间中，不同类别之间的样本尽可能地分开，同时同一类别内的样本尽可能地聚集。
- **PCA:** 找到数据中方差最大的方向，将高维数据投影到低维空间，同时尽可能保留原始数据的信息。



# 几种判别分析的区别

- **距离判别：**主要基于样本到各类别中心的距离来判别，如马氏距离判别法考虑了变量间的协方差。
- **贝叶斯判别：**基于概率模型，充分利用先验概率和类条件概率密度来进行分类，对数据的概率分布假设较为严格。
- **LDA：**LDA 在考虑类别中心距离的同时，还兼顾了类别内部的离散程度，在很多情况下能够找到更具判别性的投影方向。LDA 主要关注数据的几何结构，通过寻找最佳投影方向来区分类别，对概率分布假设相对较弱。