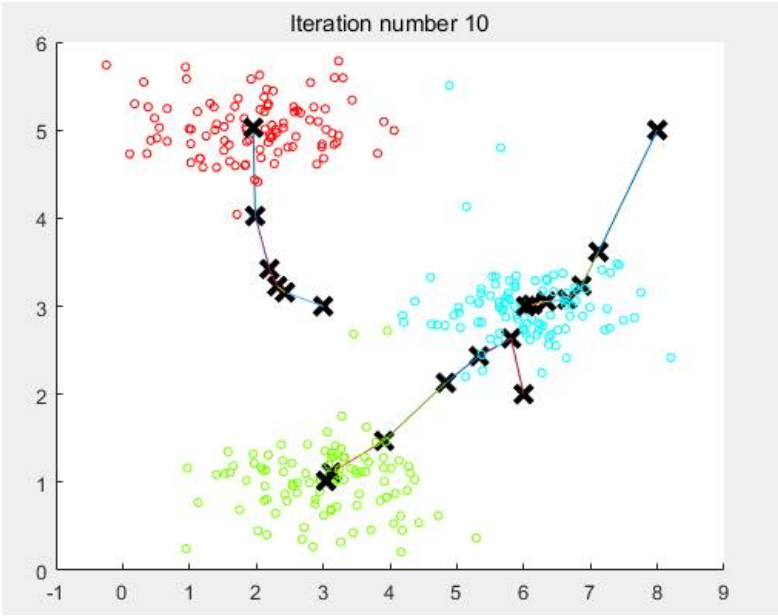
17.5.3（K-Means、PCA、异常检测）

K-Means聚类算法是常用的非监督学习聚类算法。

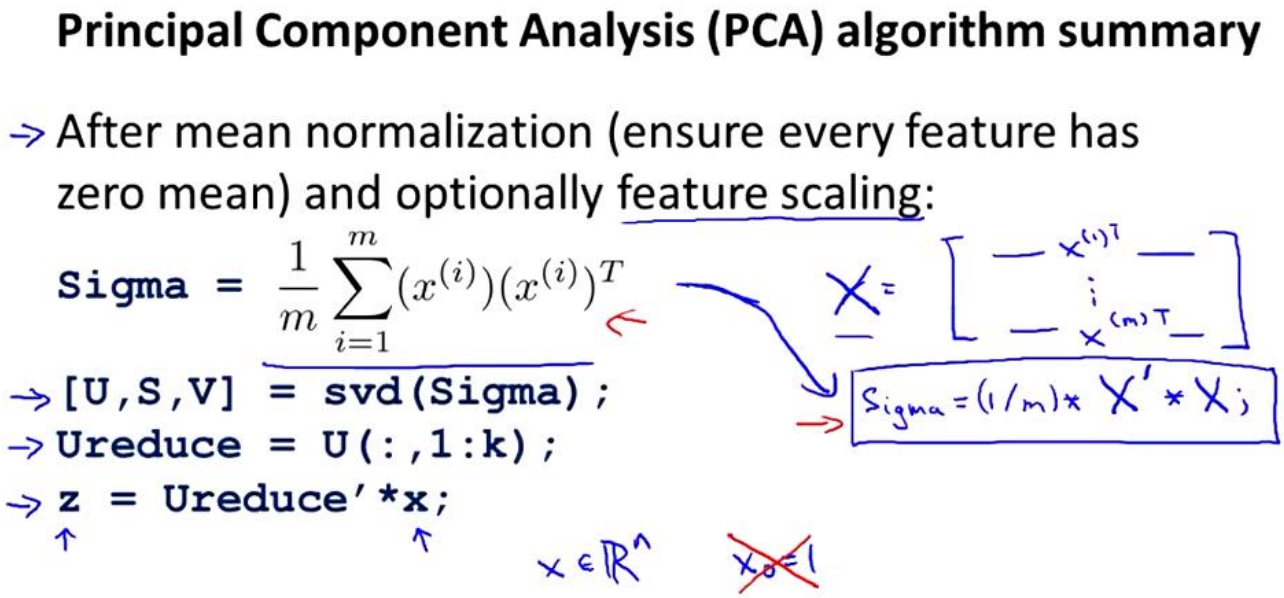


如上图所示：执行算法时，先初始化聚类数（簇的数量）和聚类中心，然后把每个样本进行分类，分好类求出当前类的平均值作为新的聚类中心，然后再分类，再求聚类中心，直到求得的聚类中心不再变化。就是把我们的样本给聚类成功了。在初始化聚类中心的时候聚类中心数要小于样本数，聚类中心可以随机从样本中获取。

$$J(c^{(1)},\ldots,c^{(m)},\mu_1,\ldots,\mu_K)=\frac{1}{m}\sum_{i=1}^{m}||x^{(i)}-\mu_{c^{(i)}}||^2$$

K-Means算法的cost function如上，求得结果是所有样本到其聚类中心的距离之和。上面算法求得最后聚类中心就是最小化cost function。

第二种非监督学习问题是降维，降维可以压缩数据，节省计算机运行内存，加快学习算法。还有助于将数据可视化。
Principal Component Analysis，主成分分析（PCA）是最常见的降维算法。



PCA算法，首先是要进行归一化处理，让每个特征的均值为0，然后用公式求sigma（协方差矩阵），然后用svd函数求sigma的特征向量，我们是从n 维降到k 维，svd函数求得的U 是n*n维的矩阵，我们取n*k维，作为我们的新特征向量，然后用Ureduce（新特征向量）的转置*X ，就得出最后的k * 1 维的特征结果。
PCA算法根据每个特征的权重大小，近似的丢掉一些特征。

异常检测(Anomaly Detection)

异常检测算法：

对于给定的数据集 $x^{(1)},x^{(2)},...,x^{(m)}$ ，我们要针对每一个特征计算 μ 和 σ^2 的估计值。

$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} x^{(i)}$$

$$\sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m} (x_j^{(i)} - \mu_j)^2$$

一旦我们获得了平均值和方差的估计值，给定新的一个训练实例，根据模型计算 p(x)：

$$p(x) = \prod_{j=1}^{n} p(x_j;\mu_j,\sigma_j^2) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(-\frac{(x_j-\mu_j)^2}{2\sigma_j^2}\right)$$

当 $p(x)<\varepsilon$ 时，为异常

p(x)=所有特征的高斯概率密度函数的积。求得预测结果和阈值进行比较，小于阈值为异常。

例如：我们有 10000 台正常引擎的数据，有 20 台异常引擎的数据。 我们这样分配数据：

6000 台正常引擎的数据作为训练集

2000 台正常引擎和 10 台异常引擎的数据作为交叉检验集

2000 台正常引擎和 10 台异常引擎的数据作为测试集

具体的评价方法如下：

1. 根据测试集数据，我们估计特征的平均值和方差并构建 p(x)函数

2. 对交叉检验集，我们尝试使用不同的 ε 值作为阀值，并预测数据是否异常，根据 F_1 值或者查准率与查全率的比例来选择 ε

3. 选出 ε 后，针对测试集进行预测，计算异常检验系统的 F_1 值，或者查准率与查全率之比

查准率（precision）：是在所有预测为1 的结果里面，正确的所占的比例。

查全率（recall）：是在实际结果为1 的里面，预测正确的所占的比例。

F：2（PR/(P+R)）

异常检测VS监督算法

异常检测	监督学习
非常少量的正向类（异常数据 y=1）， 大量的负向类（y=0）	同时有大量的正向类和负向类
许多不同种类的异常，非常难。根据非常少量的正向类数据来训练算法。	有足够多的正向类实例，足够用于训练算法，未来遇到的正向类实例可能与训练集中的非常近似。
未来遇到的异常可能与已掌握的异常、非常的不同。	
例如： 1. 欺诈行为检测 2. 生产（例如飞机引擎） 3. 检测数据中心的计算机运行状况	例如： 1. 邮件过滤器 2. 天气预报 3. 肿瘤分类

在进行异常检测前对特征数据进行处理，让数据呈正态分布。