# 数据挖掘中的距离测度及其应用

在数据挖掘中常常用到一些基础的组件。比如最小化是绝大部分算法的一个共同的组件。比如定义内积是常用的核方法的重要环节。比如定义距离和相似度，是常用的聚类、相关性分析、推荐系统算法的基本组件。这里重点介绍距离和相似度的定义。

距离和相似度的主要作用是判定两个同一属性对象之间的相似或相异程度。通常的模式是dist(X, Y) => R。

这里X,Y可以为两个一维或者多维数组，也可以是两个函数（比如两个分布）。还可以是两个群体。

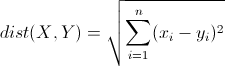
X=（x1, x2, x3, … xn），Y=（y1, y2, y3, … yn）。下面来看看主要可以用哪些方法来衡量两者的差异，主要分为距离度量和相似度度量。

## 数值型向量的距离度量

　　距离度量（Distance）用于衡量个体在空间上存在的距离，距离越远说明个体间的差异越大。

### 欧几里得距离(Euclidean Distance)

　　欧氏距离是最常见的距离度量，衡量的是多维空间中各个点之间的绝对距离。公式如下：



　　因为计算是基于各维度特征的绝对数值，所以会受到量纲的影响。另外对异常值敏感。

### 明可夫斯基距离(Minkowski Distance)

　　明氏距离是欧氏距离的推广，是对多个距离度量公式的概括性的表述。公式如下：

Minkowski Distance

　　这里的p值是一个变量，当p=2的时候就得到了上面的欧氏距离。

### 曼哈顿距离(Manhattan Distance)

　　曼哈顿距离来源于城市区块距离，是将多个维度上的距离进行求和后的结果，即当上面的明氏距离中p=1时得到的距离度量公式，如下：

Manhattan Distance

### 切比雪夫距离(Chebyshev Distance)

　　切比雪夫距离起源于国际象棋中国王的走法，我们知道国际象棋国王每次只能往周围的8格中走一步，那么如果要从棋盘中A格(x1, y1)走到B格(x2, y2)最少需要走几步？扩展到多维空间，其实切比雪夫距离就是当p趋向于无穷大时的明氏距离：

Chebyshev Distance

　　其实上面的曼哈顿距离、欧氏距离和切比雪夫距离都是明可夫斯基距离在特殊条件下的应用。

### 马哈拉诺比斯距离(Mahalanobis Distance)

既然欧几里得距离无法忽略指标度量的差异，所以在使用欧氏距离之前需要对底层指标进行[数据的标准化](http://webdataanalysis.net/data-analysis-method/data-normalization/" \t "_blank)，而基于各指标维度进行标准化后再使用欧氏距离就衍生出来另外一个距离度量——马哈拉诺比斯距离（Mahalanobis Distance），简称马氏距离。

### 向量空间余弦相似度(Cosine Similarity)

　　余弦相似度用向量空间中两个向量夹角的余弦值作为衡量两个个体间差异的大小。相比距离度量，余弦相似度更加注重两个向量在方向上的差异，而非距离或长度上。公式如下：

Cosine Similarity

### 皮尔森相关系数(Pearson Correlation Coefficient)

　　即相关分析中的相关系数r，分别对X和Y基于自身总体标准化后计算空间向量的余弦夹角。公式如下：

Pearson Correlation Coefficient

### Jaccard相似系数(Jaccard Coefficient)

　　Jaccard系数主要用于计算符号度量或布尔值度量的个体间的相似度，因为个体的特征属性都是由符号度量或者布尔值标识，因此无法衡量差异具体值的大小，只能获得“是否相同”这个结果，所以Jaccard系数只关心个体间共同具有的特征是否一致这个问题。如果比较X与Y的Jaccard相似系数，只比较xn和yn中相同的个数，公式如下：

Jaccard Coefficient

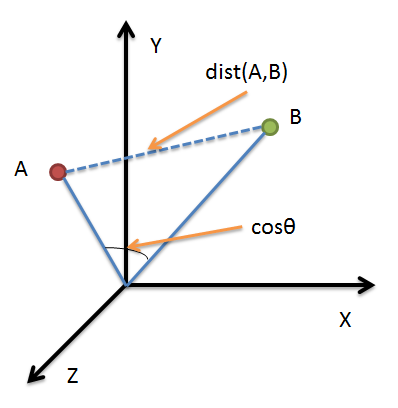
### 调整余弦相似度(Adjusted Cosine Similarity)

　　虽然余弦相似度对个体间存在的偏见可以进行一定的修正，但是因为只能分辨个体在维之间的差异，没法衡量每个维数值的差异，会导致这样一个情况：比如用户对内容评分，5分制，X和Y两个用户对两个内容的评分分别为(1,2)和(4,5)，使用余弦相似度得出的结果是0.98，两者极为相似，但从评分上看X似乎不喜欢这2个内容，而Y比较喜欢，余弦相似度对数值的不敏感导致了结果的误差，需要修正这种不合理性，就出现了调整余弦相似度，即所有维度上的数值都减去一个均值，比如X和Y的评分均值都是3，那么调整后为(-2,-1)和(1,2)，再用余弦相似度计算，得到-0.8，相似度为负值并且差异不小，但显然更加符合现实。

### 欧氏距离与余弦相似度

　　欧氏距离是最常见的距离度量，而余弦相似度则是最常见的相似度度量，很多的距离度量和相似度度量都是基于这两者的变形和衍生，所以下面重点比较下两者在衡量个体差异时实现方式和应用环境上的区别。

　　借助三维坐标系来看下欧氏距离和余弦相似度的区别：

[](http://webdataanalysis.net/wp-content/uploads/2011/10/distance-and-similarity.png)

　　从图上可以看出距离度量衡量的是空间各点间的绝对距离，跟各个点所在的位置坐标（即个体特征维度的数值）直接相关；而余弦相似度衡量的是空间向量的夹角，更加的是体现在方向上的差异，而不是位置。如果保持A点的位置不变，B点朝原方向远离坐标轴原点，那么这个时候余弦相似度cosθ是保持不变的，因为夹角不变，而A、B两点的距离显然在发生改变，这就是欧氏距离和余弦相似度的不同之处。

　　根据欧氏距离和余弦相似度各自的计算方式和衡量特征，分别适用于不同的数据分析模型：欧氏距离能够体现个体数值特征的绝对差异，所以更多的用于需要从维度的数值大小中体现差异的分析，如使用用户行为指标分析用户价值的相似度或差异；而余弦相似度更多的是从方向上区分差异，而对绝对的数值不敏感，更多的用于使用用户对内容评分来区分用户兴趣的相似度和差异，同时修正了用户间可能存在的度量标准不统一的问题（因为余弦相似度对绝对数值不敏感）。

上面都是对距离度量和相似度度量的一些整理和汇总，在现实的使用中选择合适的距离度量或相似度度量可以完成很多的数据分析和数据挖掘的建模，后续会有相关的介绍。

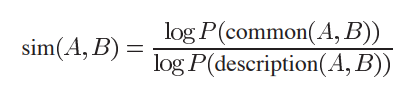
## 离散型向量的距离

其实没有距离，只有等和不等。

### 海明距离

### Lin框架





An Information-Theoretic Definition of Similarity,1998:296-304

## 数值型序列的距离

特点是X的所有数据都是同一属性，因为一些因素产生先后次序。最常见的就是时间序列。这里序列和向量并不是指两种数据类型。而是为区别两者而产生的称谓。当然数值型序列也可以放到向量中，因此数值型序列也可以使用数值型向量的距离测度。但有些场景不合适：比如非等长序列，如两串DNA（XSRA和SA）;非数值类型，比如两个字符串的相似度；比如对x下标不敏感但对形状敏感。

序列类型的相似度区别于一般向量相似度在于：如果把序列的每一项看做一个特征，可以把序列看做是向量。但序列中属性是相同的，因此特征是可以移动或者部分移动的。序列相似度的度量是向量相似度度量的更广义的情况。

集合是序列条件更加放宽的情况。

### DTW

## 离散型序列的距离

### 最长子序列

### 最长公共子序列

### 编辑距离

### simHash + 海明距离

### 映射 + 余弦相似度

## 集合的距离

### [Dice 距离](https://en.wikipedia.org/wiki/S%C3%B8rensen%E2%80%93Dice_coefficient)

### [Jaccard distance](https://en.wikipedia.org/wiki/Jaccard_index)

## 混合数据的距离

### 加权求和

## 特殊侧重点的距离测度

## 求距离前的必要处理：常见的特征转换方式

### 字符转换为数值类型

### 降维

### Trigram（源于ngram）

## 数据挖掘中的其他距离

### SimRank距离

图相似度

### Frechet距离

轨迹相似度

## 场景应用