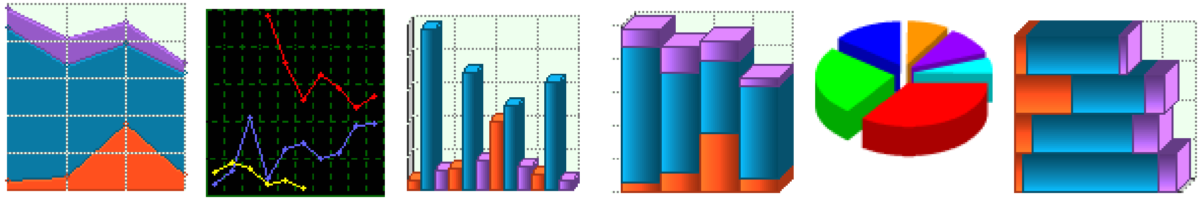
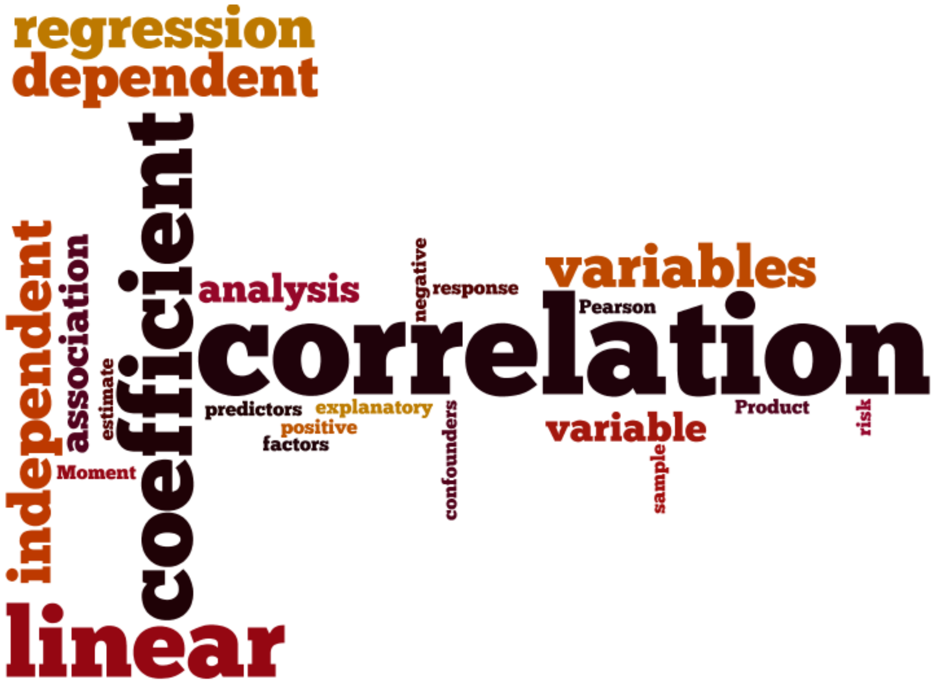





# Unit 4: Covariance, Correlation & Regression

# 回顾：离散趋势（单元2 Measure of dispersion）

- **均值（Mean）**

$$\overline{X} = \frac{\sum_{i=1}^{n} X_i}{n}$$

- **方差（Variance）**

$$S^2 = \frac{\sum_{i=1}^{n} (X_i - \overline{X})^2}{n-1}$$

- **标准差 (SD, STD)**

$$\mathrm{S} = \sqrt{\frac{\sum_{i=1}^{n} (X_i - \overline{X})^2}{n-1}}$$

# 1. 协方差（covariance）

- **用来度量两个随机变量关系的统计量**
  - 可以通俗的理解为：两个变量在变化过程中是同方向变化？还是反方向变化？同向或反向程度如何
    - 你变大，同时我也变大，说明两个变量是同向变化的，这时协方差就是正的
    - 你变大，同时我变小，说明两个变量是反向变化的，这时协方差就是负的
    - 从数值来看，协方差的数值越大，两个变量同向程度也就越大。反之亦然

$$Cov(X,Y)=E[(X-\mu_x)(Y-\mu_y)]=\frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{n-1}$$

**简易版理解：**
有X, Y两个变量，每个时刻的“X值和其均值之差”乘以“Y值和其均值之差”得到一个乘积后，再对这每时刻的乘积求和并求和再均值。

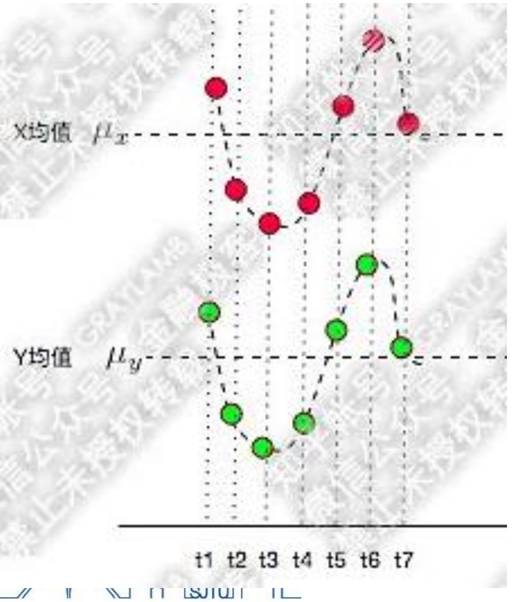
# 1. 协方差（covariance）

- **用来度量两个随机变量关系的统计量**
  - 可以通俗的理解为：两个变量在变化过程中是同方向变化？还是反方向变化？同向或反向程度如何
    - 你变大，同时我也变大，说明两个变量是同向变化的，这时协方差就是正的
    - 你变大，同时我变小，说明两个变量是反向变化的，这时协方差就是负的
    - 从数值来看，协方差的数值越大，两个变量同向程度也就越大。反之亦然

$$Cov(X,Y) = E[(X-\mu_x)(Y-\mu_y)] = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{n-1}$$

**简易版理解：**
有X, Y两个变量，每个时刻的“X值和其均值之差”乘以“Y值和其均值之差”得到一个乘积后，再对这每时刻的乘积求和并求和再均值。

**举个例子**：两个变量X,Y，观察 t1-t7（7个时刻）他们的变化情况




分别用红点和绿点表示X、Y，横轴是时间。可以看到X，Y均围绕各自的均值运动，并且很明显是同向变化的。

我们看到每一时刻 $(X - \mu_x)$ 的值与 $(Y - \mu_y)$ 的值的“正负号”一定相同。

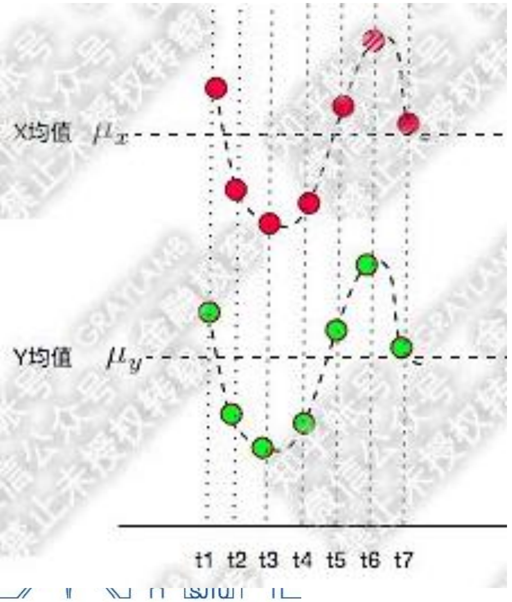
# 1. 协方差（covariance）

- **用来度量两个随机变量关系的统计量**
  - 可以通俗的理解为：两个变量在变化过程中是同方向变化？还是反方向变化？同向或反向程度如何
    - 你变大，同时我也变大，说明两个变量是同向变化的，这时协方差就是正的
    - 你变大，同时我变小，说明两个变量是反向变化的，这时协方差就是负的
    - 从数值来看，协方差的数值越大，两个变量同向程度也就越大。反之亦然

$$Cov(X,Y)=E[(X-\mu_x)(Y-\mu_y)]=\frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{n-1}$$

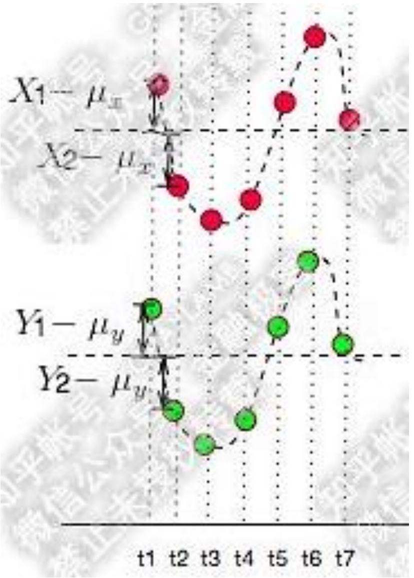
**简易版理解：**
有X, Y两个变量，每个时刻的“X值和其均值之差”乘以“Y值和其均值之差”得到一个乘积后，再对这每时刻的乘积求和并求和再均值。

**举个例子**：两个变量X,Y，观察 t1-t7（7个时刻）他们的变化情况




分别用红点和绿点表示X、Y，横轴是时间。可以看到X，Y均围绕各自的均值运动，并且很明显是同向变化的。

我们看到每一时刻 $(X-\mu_x)$ 的值与 $(Y-\mu_y)$ 的值的“正负号”一定相同。




比如t1时刻，他们同为正，t2时刻他们同为负

当他们同向变化时，$(X-\mu_x)$ 与 $(Y-\mu_y)$ 的乘积为正。这样，当你把 t1-t7 时刻 $(X-\mu_x)$ 与 $(Y-\mu_y)$ 的乘积加在一起，求平均也是**正数**了。

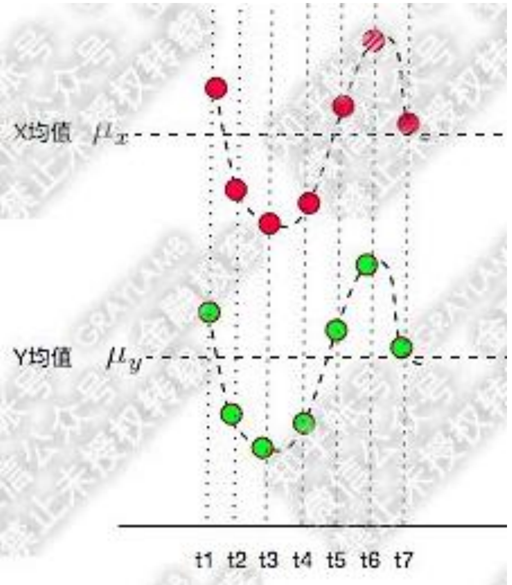
# 1. 协方差（covariance）

- **用来度量两个随机变量关系的统计量**
  - 可以通俗的理解为：两个变量在变化过程中是同方向变化？还是反方向变化？同向或反向程度如何
    - 你变大，同时我也变大，说明两个变量是同向变化的，这时协方差就是正的
    - 你变大，同时我变小，说明两个变量是反向变化的，这时协方差就是负的
    - 从数值来看，协方差的数值越大，两个变量同向程度也就越大。反之亦然

$$Cov(X,Y) = E[(X-\mu_x)(Y-\mu_y)] = \frac{\sum_{i=1}^{n}(X_i-\bar{X})(Y_i-\bar{Y})}{n-1}$$

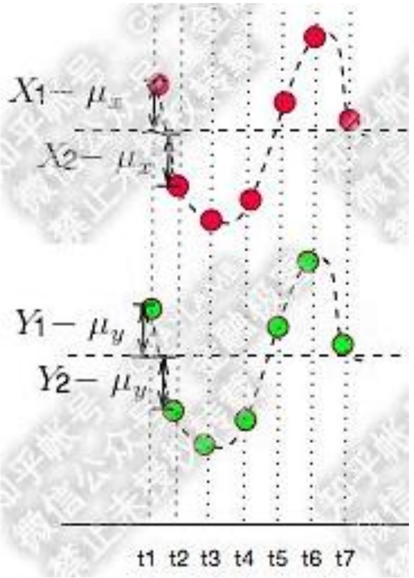
**简易版理解：**
有X, Y两个变量，每个时刻的“X值和其均值之差”乘以“Y值和其均值之差”得到一个乘积后，再对这每时刻的乘积求和并求和再均值。

**举个例子**：两个变量X,Y，观察 t1-t7（7个时刻）他们的变化情况




分别用红点和绿点表示X、Y，横轴是时间。可以看到X，Y均围绕各自的均值运动，并且很明显是同向变化的。

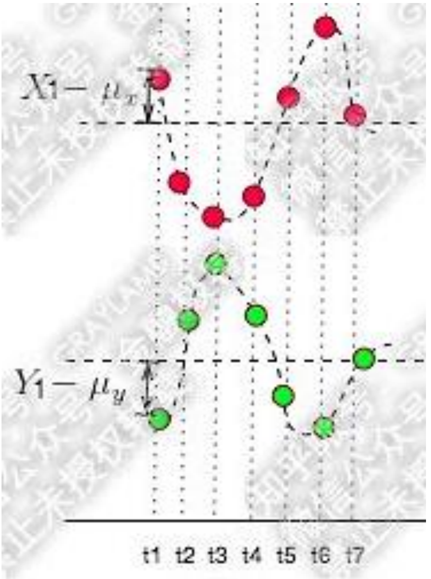
我们看到每一时刻 $(X-\mu_x)$ 的值与 $(Y-\mu_y)$ 的值的“正负号”一定相同。




比如t1时刻，他们同为正，t2时刻他们同为负

当他们同向变化时，$(X-\mu_x)$ 与 $(Y-\mu_y)$ 的乘积为正。这样，当你把 t1-t7 时刻 $(X-\mu_x)$ 与 $(Y-\mu_y)$ 的乘积加在一起，求平均也是**正数**了。

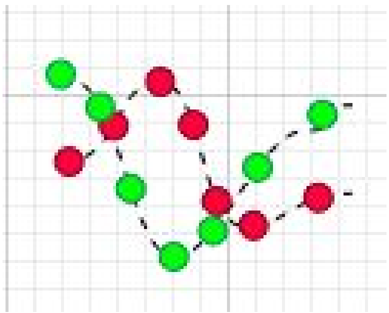
如果反向运动呢？

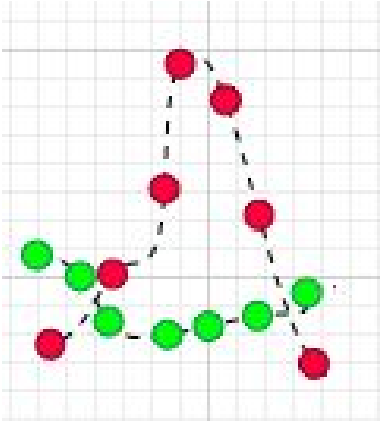



很明显，$(X-\mu_x)$ 的值与 $(Y-\mu_y)$ 的值的“正负号”一定相反了，$(X-\mu_x)$ 与 $(Y-\mu_y)$ 的乘积就是负值了。

这样，当你把 t1-t7 时刻 $(X-\mu_x)$ 与 $(Y-\mu_y)$ 的乘积加在一起，求平均也就是**负数**了。

但很多时候 X，Y 的运动是不规律的，比如：





这时， 很可能某一时$(X - \mu_x)$ 的值与$(Y - \mu_y)$ 的乘积为正，另一时刻$(X - \mu_x)$ 的值与$(Y - \mu_y)$ 的乘积为负。

将每一时刻 $(X - \mu_x)$ 与$(Y - \mu_y)$ 的乘积加在一起，其中的正负项就会抵消，最后求平均得出的就是**协方差**，然后通过协方差的数值大小，就可以判断这两个变量同向或反向的程度了。

所以，如果 例子里的 t1-t7时刻中， $(X - \mu_x)$ 与$(Y - \mu_y)$ 的乘积为正的越多，说明同向变化的次数越多，也即同向程度越高。反之亦然。

**总结一下，如果协方差为正，说明X, Y 同向变化，协方差越大说明同向程度越高；如果协方差为负，说明X, Y 反向运动，协方差越小说明反向越高。**

# wait ...

那如果X，Y同向变化，但X大于均值，Y小于均值，那$(X - \mu_x)$ 与$(Y - \mu_y)$ 的乘积为负值啊？这不是矛盾了吗？

# wait ...

那如果X，Y同向变化，但X大于均值，Y小于均值，那 $(X - \mu_x)$ 与$(Y - \mu_y)$ 的乘积为负值啊？这不是矛盾了吗？

再来，如果t1, t2, t3...t7 时刻X, Y 都在增大，而且X都比均值大，Y都比均值小，这种情况协方差不就是负的了？但是X, Y都是增大的，都是同向变化的，这又矛盾了？

这个怎么解释呢？

# Python for covariance

numpy.cov

```
def cov(a, b):

    if len(a) != len(b):
        return

    a_mean = np.mean(a)
    b_mean = np.mean(b)

    sum = 0

    for i in range(0, len(a)):
        sum += ((a[i] - a_mean) * (b[i] - b_mean))

    return sum/(len(a)-1)
```

## numpy.cov¶

numpy.cov(*m, y=None, rowvar=True, bias=False, ddof=None, fweights=None, aweights=None*) [source]

Estimate a covariance matrix, given data and weights.

Covariance indicates the level to which two variables vary together. If we examine N-dimensional samples, $X = [x_1, x_2, ... x_N]^T$, then the covariance matrix element $C_{ij}$ is the covariance of $x_i$ and $x_j$. The element $C_{ii}$ is the variance of $x_i$.

See the notes for an outline of the algorithm.

**Parameters:**

m : *array_like*
A 1-D or 2-D array containing multiple variables and observations. Each row of *m* represents a variable, and each column a single observation of all those variables. Also see *rowvar* below.

y : *array_like, optional*
An additional set of variables and observations. *y* has the same form as that of *m*.

rowvar : *bool, optional*
If *rowvar* is True (default), then each row represents a variable, with observations in the columns. Otherwise, the relationship is transposed: each column represents a variable, while the rows contain observations.

bias : *bool, optional*
Default normalization (False) is by `(N - 1)`, where `N` is the number of observations given (unbiased estimate). If *bias* is True, then normalization is by `N`. These values can be overridden by using the keyword `ddof` in numpy versions >= 1.5.

ddof : *int, optional*
If not `None` the default value implied by *bias* is overridden. Note that `ddof=1` will return the unbiased estimate, even if both *fweights* and *aweights* are specified, and `ddof=0` will return the simple average. See the notes for the details. The default value is `None`.
*New in version 1.5.*

fweights : *array_like, int, optional*
1-D array of integer frequency weights; the number of times each observation vector should be repeated.
*New in version 1.10.*

aweights : *array_like, optional*
1-D array of observation vector weights. These relative weights are typically large for observations considered "important" and smaller for observations considered less "important". If `ddof=0` the array of weights can be used to assign probabilities to observation vectors.
*New in version 1.10.*

**Returns:**

out : *ndarray*
The covariance matrix of the variables.

# 2. 相关系数（correlation coefficient）

- 相关系数的公式为：

$$\rho = \frac{Cov(X,Y)}{\sigma_X \sigma_Y} \quad r(X,Y) = \frac{Cov(X,Y)}{\sqrt{Var[X]\,Var[Y]}}$$

  - **$X$，$Y$ 的协方差除以 $X$ 的标准差和 $Y$ 的标准差**

**Population versus Sample**

| | Parameter | | Statistic | |
|---|---|---|---|---|
| Mean | μ | m | $\bar{x}$ | x-bar |
| Proportion | p | u | $\hat{p}$ | p-hat |
| Std. Dev. | σ | sigma | $s$ | |
| Correlation | ρ | rho | $r$ | |

- 所以，相关系数也可以看成：**一种**剔除了两个变量量纲影响、标准化后的**特殊协方差（如同变异系数是标准化后的标准差）**

- 是一种标准化后的协方差
  - 可以反映两个变量变化时是同向还是反向，如果同向变化就为正，反向变化就为负
  - 由于它是标准化后的协方差，因此它还有个重要的特性：**它消除了两个变量变化幅度的影响，只是单纯反映了两个变量每个单位变化时的相似程度，这样不同的实验之间就可以进行比较了**
  - 取值在 -1 到 1 之间
  - 通常绝对值大于0.7时认为两变量之间表现出非常强的相关关系，绝对值大于0.4时认为有着强相关关系，绝对值小于0.2时相关关系较弱。

## **举个例子**：还是用之前的例子，变量X、Y变化的示意图（X为红点，Y为绿点），来看两种情况：

很容易可以看到图一，图二两种情况下的，X，Y都是同向变化的，而这个“同向变化”，有个显著特征：X, Y同向变化的过程，具有极高的相似度！无论是在图一还是图二的情况下，都是

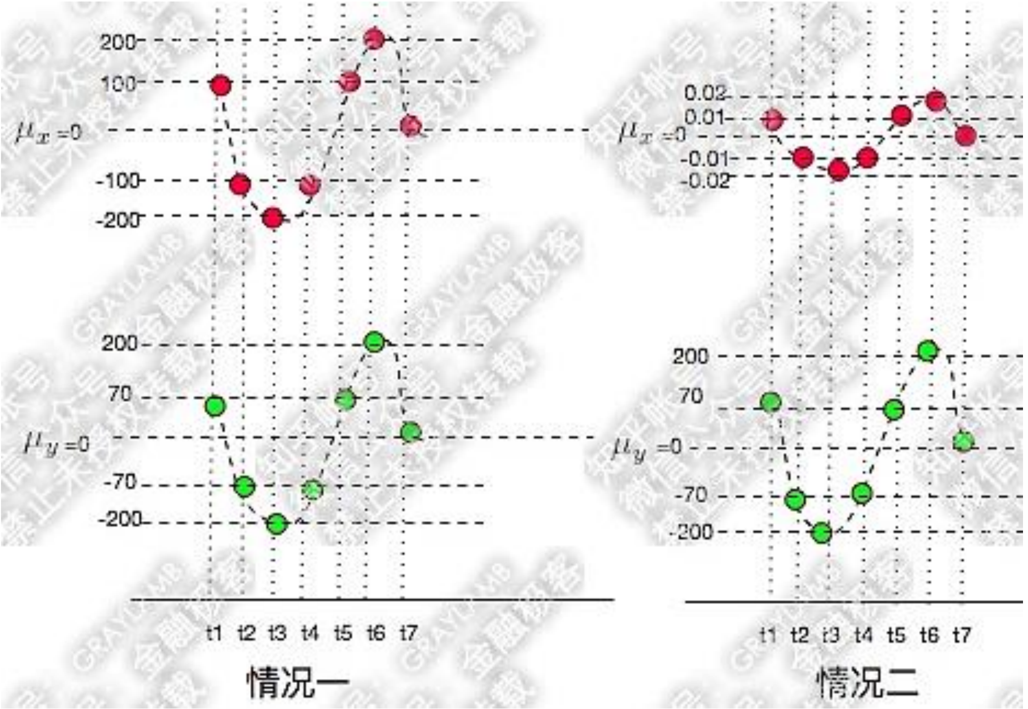
t1 时刻X, Y 都大于均值，

t2 时刻都变小且小于均值，

t3 时刻X, Y 继续变小且小于均值，

t4 时刻X, Y 变大但仍小于均值，

t5 时刻X, Y 变大且大于均值。。。




可是，计算下协方差：

第一种情况下：

$[(100-0)\times(70-0)+(-100-0)\times(-70-0)+(-200-0)\times(-200-0)\ldots]\ \mathrm{div}\ 7\approx 15428.57$

第二种情况下：

$[(0.01-0)\times(70-0)+(-0.01-0)\times(-70-0)+(-0.02-0)\times(-200-0)\ldots]\ \mathrm{div}\ 7\approx 1.542857$

第一种情况下：

X的标准差为

$\sigma_X=\sqrt{E((X-\mu_x)^2)}=\sqrt{[(100-0)^2+(-100-0)^2\ldots]\ \mathrm{div}\ 7}\approx 130.9307$

Y的标准差为

$\sigma_Y=\sqrt{E((Y-\mu_y)^2)}=\sqrt{[(70-0)^2+(-70-0)^2\ldots]\ \mathrm{div}\ 7}\approx 119.2836$

于是相关系数为

$\rho=15428.57\div(130.9307\times 119.2836)\approx 0.9879$

第二种情况下：

X的标准差为

$\sigma_X=\sqrt{E((X-\mu_x)^2)}=\sqrt{[(0.01-0)^2+(-0.01-0)^2\ldots]\ \mathrm{div}\ 7}\approx 0.01309307$

Y的标准差为

$\sigma_Y=\sqrt{E((Y-\mu_y)^2)}=\sqrt{[(70-0)^2+(-70-0)^2\ldots]\ \mathrm{div}\ 7}\approx 119.2836$

于是相关系数为

$\rho=1.542857\ \mathrm{div}(0.01309307\times 119.2836)\approx 0.9879$

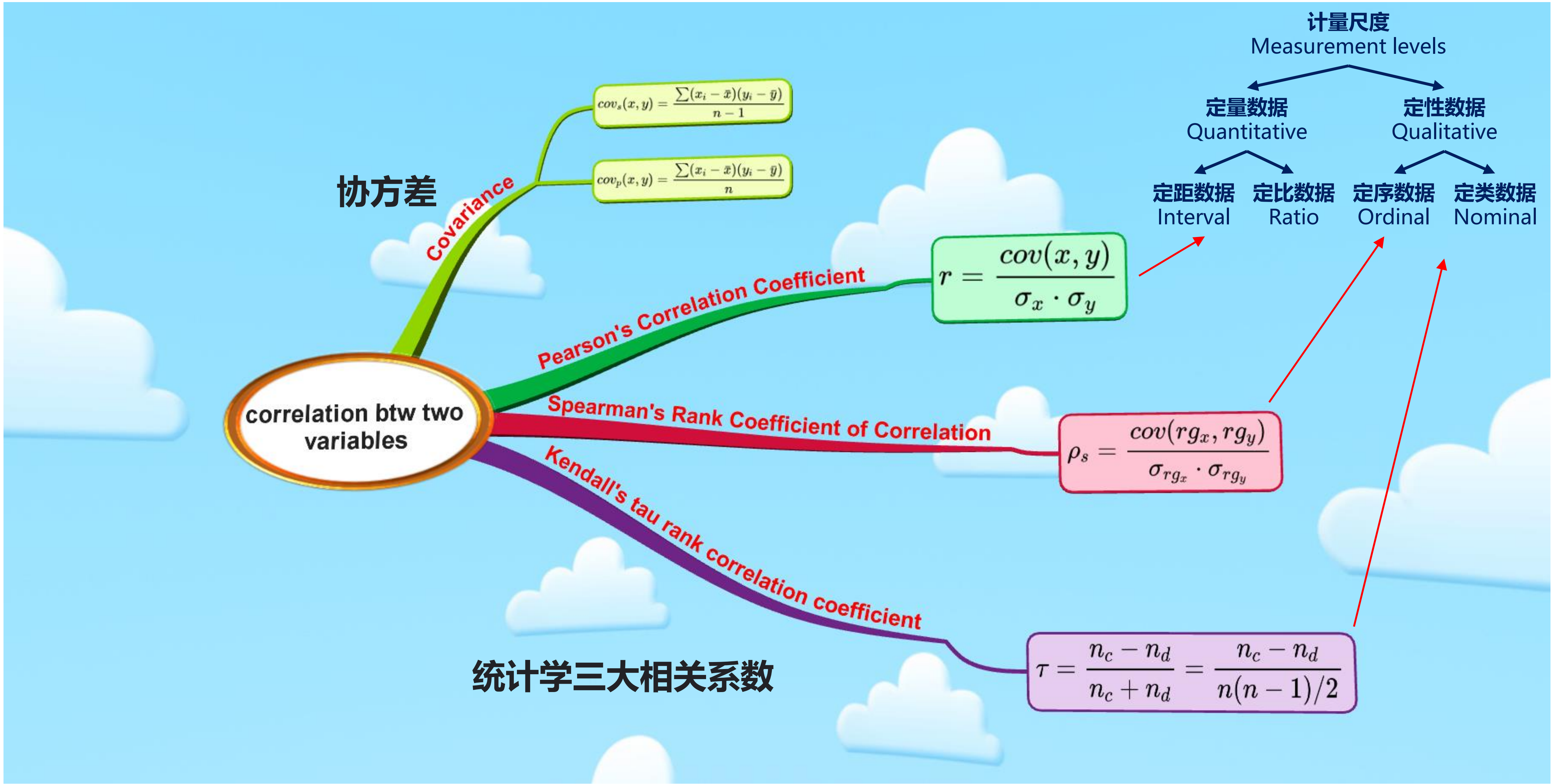
协方差差了一万倍，只能从2个协方差都是正数来判断这两种情况下的X, Y 都是同向变化，但是无法看出两种情况下X, Y 的变化是否具有相似性。

**说明第二种情况下，虽然X的变化幅度比第一张情况X的变化幅度小了10000倍，但是丝毫没有改变“X的变化与Y的变化具有很高的相似度”这个结论。同时这两种情况的相关系数相等，说明有着一样的相似度。**

所以，为了能准确的研究两个变量在变化过程中的相似度，我们需要把变化幅度对协方差的影响，从协方差中剔除掉。于是，就有了相关系数的公式了

$$\rho=\frac{Cov(X,Y)}{\sigma_X\sigma_Y}$$

# Describing the correlation between two variables



# 谢谢，下周见！