

In [69]:

```
import numpy as np
import pandas as pd
from itertools import combinations
```

Задача 1

Даны значения величины заработной платы заемщиков банка (salary) и значения их поведенческого кредитного скоринга (scoring): salary = [35, 45, 190, 200, 40, 70, 54, 150, 120, 110] scoring = [401, 574, 874, 919, 459, 739, 653, 902, 746, 832] Используя только встроенные питоновские функции и структуры данных (т.е. без библиотек numpy, pandas и др.) найдите: ковариацию (смещённую и несмещённую) этих двух величин, коэффициент корреляции Пирсона, коэффициент корреляции Кендалла. Можно затем посчитать те же значения с использованием библиотек, чтобы проверить себя.

In [56]:

```
salary = [35, 45, 190, 200, 40, 70, 54, 150, 120, 110]
scoring = [401, 574, 874, 919, 459, 739, 653, 902, 746, 832]
XY = []
XY1 = 0
Sal = 0
Scr = 0
i = 0
while i < len(salary):
    XY.append(salary[i] * scoring[i])
    XY1 += (salary[i] * scoring[i])
    Sal += salary[i]
    Scr += scoring[i]
    i+=1
XYmean = XY1/len(XY)
MX = Sal/len(salary)
MY = Scr/len(scoring)
COV1 = XYmean - Xmean*Ymean
COV2 = np.mean(XY) - np.mean(salary)*np.mean(scoring)
COV1, COV2
```

Out [56]:

(9157.839999999997, 9157.839999999997)

In [64]:

```
SX = 0
SY = 0
k = 0
while k < len(salary):
    SX += (salary[k] - MX)**2
    SY += (scoring[k] - MY)**2
    k+=1
sigmaX = (SX/len(salary))**0.5
sigmaY = (SY/len(scoring))**0.5
sigmaX, sigmaY
```

Out [64]:

```
(59.115480206118605, 174.55340157098058)
```

In [65]:

```
RXY = COV1/(sigmaX*sigmaY)
RXY
```

Out [65]:

```
0.8874900920739158
```

In [66]:

```
np.corrcoef(salary, scoring)
```

Out [66]:

```
array([[1., 0.88749009],
       [0.88749009, 1.]])
```

Коэффициент корреляции методом Кендала (ранговый) tau:

In [70]:

```
def is_concordant(pair1: tuple, pair2: tuple) -> bool:
    return (pair1[0] - pair2[0]) * (pair1[1] - pair2[1]) > 0
```

In [71]:

```
P = 0
Q = 0
for pair1, pair2 in combinations(zip(salary, scoring), r=2):
    if is_concordant(pair1, pair2):
        P+=1
    else:
        Q+=1
```

P,Q

Out[71]:

(43, 2)

In [72]:

```
tau = (P-Q)/(P+Q)
tau
```

Out[72]:

0.9111111111111111

Задача 2

Измерены значения IQ выборки студентов, обучающихся в местных технических вузах:

131, 125, 115, 122, 131, 115, 107, 99, 125, 111

Известно, что в генеральной совокупности IQ распределен нормально. Найдите доверительный интервал для математического ожидания с надежностью 0.99.

In [73]:

```
IQ = [131, 125, 115, 122, 131, 115, 107, 99, 125, 111]
M = np.mean(IQ)
M
```

Out[73]:

118.1

In [75]:

```
(M - 3 * np.sqrt(M/len(IQ)), M + 3 * np.sqrt(M/len(IQ)))
```

Out[75]:

```
(107.79029583353625, 128.40970416646374)
```