**Домашнее задание 8. Корреляционный анализ**

**Задача 1.**

*Даны значения величины заработной платы заемщиков банка (zp) и значения их поведенческого кредитного скоринга (ks):*

*zp = [35, 45, 190, 200, 40, 70, 54, 150, 120, 110],*

*ks = [401, 574, 874, 919, 459, 739, 653, 902, 746, 832].*

**Решение:**

***1.*** *Найдите ковариацию этих двух величин с помощью элементарных действий, а затем с помощью функции cov из numpy. Полученные значения должны быть равны.*

Вычислим ковариацию с помощью элементарных действий:

1.1 Вычисляем среднее значение:

1.2 Вычисляем ковариацию:

Вычислим ковариацию с помощью функции cov из numpy:

**import** numpy **as** np

zp **=** [35, 45, 190, 200, 40, 70, 54, 150, 120, 110]

ks **=** [401, 574, 874, 919, 459, 739, 653, 902, 746, 832]

covariance **=** np**.**cov(zp, ks, ddof**=**0)[0, 1]

print(covariance)

9157.84

***2.*** *Найдите коэффициент корреляции Пирсона с помощью ковариации и среднеквадратичных отклонений двух признаков, а затем с использованием функций из библиотек numpy и pandas.*

Выполним задачу с помощью ковариации и среднеквадратичных отклонений двух признаков:

2.1 Найдем среднее значение каждой из величин (вычисления приведены в п.1.1):

2.2 Найдем ковариацию между величинами: (вычисления приведены в п.1.2):

2.3 Найдем среднеквадратичные отклонения каждой из величин:

2.4 Найдем коэффициент корреляции Пирсона по формуле:

Таким образом, коэффициент корреляции Пирсона между заработной платой и кредитным скорингом равен приблизительно 0.8875.

Для нахождения коэффициента корреляции Пирсона с помощью функций библиотек numpy и pandas необходимо выполнить следующие шаги:

*# Импортировать библиотеку numpy и pandas:*

**import** numpy **as** np

**import** pandas **as** pd

*# Создать два массива numpy с данными:*

zp **=** np**.**array([35, 45, 190, 200, 40, 70, 54, 150, 120, 110])

ks **=** np**.**array([401, 574, 874, 919, 459, 739, 653, 902, 746, 832])

*# Найти ковариацию двух признаков с помощью функции cov из библиотеки numpy:*

covariance **=** np**.**cov(zp, ks, ddof**=**0)[0, 1]

*# Найти среднее значение и стандартное отклонение для каждого признака с помощью функций mean и std из библиотеки numpy:*

mean\_zp **=** np**.**mean(zp)

std\_zp **=** np**.**std(zp, ddof**=**0)

mean\_ks **=** np**.**mean(ks)

std\_ks **=** np**.**std(ks, ddof**=**0)

*# Найти коэффициент корреляции Пирсона с помощью формулы:*

pearson **=** covariance **/** (std\_zp **\*** std\_ks)

*# Найти коэффициент корреляции Пирсона с помощью функции corrcoef из библиотеки numpy:*

pearson\_np **=** np**.**corrcoef(zp, ks)[0, 1]

*# Найти коэффициент корреляции Пирсона с помощью функции corr из библиотеки pandas:*

data **=** pd**.**DataFrame({'zp': zp, 'ks': ks})

pearson\_pd **=** data**.**corr()**.**loc['zp', 'ks']

*# Вывести значения коэффициента корреляции Пирсона, полученные разными способами:*

print("Коэффициент корреляции Пирсона (с помощью ковариации и среднеквадратичных отклонений):", pearson)

print("Коэффициент корреляции Пирсона (с помощью функции corrcoef из numpy):", pearson\_np)

print("Коэффициент корреляции Пирсона (с помощью функции corr из pandas):", pearson\_pd)

Коэффициент корреляции Пирсона (с помощью ковариации и среднеквадратичных отклонений): 0.8874900920739162

Коэффициент корреляции Пирсона (с помощью функции corrcoef из numpy): 0.8874900920739162

Коэффициент корреляции Пирсона (с помощью функции corr из pandas): 0.8874900920739168

**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 0.95.*

**Решение:**

Для нахождения доверительного интервала для математического ожидания нам нужно знать среднее значение выборки и её стандартное отклонение.

Используем стандартное отклонение выборки в качестве оценки стандартного отклонения генеральной совокупности, так как мы не знаем его истинного значения:

**import** numpy **as** np

sample **=** np**.**array([131, 125, 115, 122, 131, 115, 107, 99, 125, 111])

n **=** len(sample)

mean **=** np**.**mean(sample)

std\_dev **=** np**.**std(sample, ddof**=**1) *# исправленное стандартное отклонение*

Здесь **n** - размер выборки, **mean** - среднее значение выборки, **std\_dev** - исправленное стандартное отклонение выборки.

Теперь мы можем найти доверительный интервал для математического ожидания при помощи формулы:

**import** scipy.stats **as** stats

confidence\_level **=** 0.95

alpha **=** 1 **-** confidence\_level

t **=** stats**.**t**.**ppf(1 **-** alpha **/** 2, df**=**n**-**1) *# критическое значение t-статистики*

lower **=** mean **-** t **\*** std\_dev **/** np**.**sqrt(n)

upper **=** mean **+** t **\*** std\_dev **/** np**.**sqrt(n)

print("Доверительный интервал: ({:.2f}, {:.2f})"**.**format(lower, upper))

Доверительный интервал: (110.56, 125.64)

Здесь t - критическое значение t-статистики, которое мы находим с помощью функции stats.t.ppf() из библиотеки scipy.stats.

Итак, доверительный интервал для математического ожидания с надежностью 0.95 для данной выборки равен (110.56, 125.64). То есть мы можем утверждать, что с 95% уверенностью истинное значение математического ожидания находится в этом интервале.

***Задача 3.***

*Известно, что рост футболистов в сборной распределен нормально с дисперсией генеральной совокупности, равной 25 кв.см.*

*Объем выборки равен 27, среднее выборочное составляет 174.2.*

*Найдите доверительный интервал для математического ожидания с надежностью 0.95.*

**Решение:**

1. Для нахождения **доверительного интервала** с надежностью 0.95 воспользуемся формулой:

где - выборочное среднее,

S - выборочное стандартное отклонение,

n - объем выборки,

- критическое значение распределения Стьюдента с (n-1) степенями свободы и уровнем доверия α/2.

Подставим значения в формулу для вычисления **доверительного интервала**:

Итак, с вероятностью 0.95 можно утверждать, что истинное значение математического ожидания a лежит в интервале от 172,31 до 176,09.

2. Также проверим решение с помощью кода

**from** scipy.stats **import** t

**import** math

*# Заданные параметры*

n **=** 27

mean **=** 174.2

stddev\_population **=** math**.**sqrt(25)

alpha **=** 0.05

*# Находим критическое значение t-статистики*

df **=** n **-** 1

t\_critical **=** t**.**ppf(1 **-** alpha**/**2, df)

*# Находим границы доверительного интервала*

margin\_of\_error **=** t\_critical **\*** stddev\_population **/** math**.**sqrt(n)

lower\_bound **=** mean **-** margin\_of\_error

upper\_bound **=** mean **+** margin\_of\_error

*# Выводим результаты*

print(f'Доверительный интервал: [{lower\_bound}, {upper\_bound}]')

Доверительный интервал: [172.2220658754539, 176.17793412454608]

Таким образом, с вероятностью 95% средний рост футболистов в сборной находится в интервале от 172.22 до 176.18 см.