Теория вероятностей и математическая статистика

Задание 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]. Найдите ковариацию этих двух величин с помощью элементарных действий, а затем с помощью функции cov из numpy Полученные значения должны быть равны. Найдите коэффициент корреляции Пирсона с помощью ковариации и среднеквадратичных отклонений двух признаков, а затем с использованием функций из библиотек numpy и pandas.

Подсказка № 1

Рассчитайте средние значения обеих переменных перед вычислением ковариации. Для этого найдите среднее значение заработной платы (`zp`) и среднее значение кредитного скоринга (`ks`). Эти значения будут использоваться для центровки данных.

Подсказка № 2

При вычислении ковариации вручную используйте формулу ковариации. Формула ковариации: cov(X,Y)=$\frac{\sum(xi-\bar{x})(yi-\bar{y})}{n-1}$, где xii и yi — значения переменных, $\bar{x}$ и $\bar{y}$ — их средние значения, и n — количество наблюдений.

Подсказка № 3

Функция `np.cov` возвращает матрицу ковариаций, где элемент [0,1] представляет собой ковариацию между двумя переменными.

Подсказка № 4

Рассчитайте стандартные отклонения двух переменных перед вычислением коэффициента корреляции. Стандартное отклонение можно найти с помощью функции `np.std` с параметром `ddof=1` для вычисления выборочного стандартного отклонения.

Подсказка № 5

Проверьте расчет коэффициента корреляции Пирсона вручную. Используйте формулу: corr(X,Y)=$\frac{cov(X,Y)}{\sigma x\ \sigma y}$, где σx и σy — стандартные отклонения переменных. Сравните полученное значение с результатами, полученными через `np.corrcoef` и `pd.Series.corr`.

Эталонное решение:

```
import numpy as np

import pandas as pd



# Данные

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])



# 1. Ковариация вручную

mean_zp = np.mean(zp)

mean_ks = np.mean(ks)



cov_manual = np.sum((zp - mean_zp) * (ks - mean_ks)) / (len(zp) - 1)

print(f"Ковариация вручную: {cov_manual}")



# 2. Ковариация с помощью numpy

cov_numpy = np.cov(zp, ks)[0, 1]

print(f"Ковариация с помощью numpy: {cov_numpy}")



# 3. Коэффициент корреляции Пирсона вручную

std_zp = np.std(zp, ddof=1)

std_ks = np.std(ks, ddof=1)



correlation_manual = cov_manual / (std_zp * std_ks)

print(f"Коэффициент корреляции Пирсона вручную:
{correlation_manual}")



# 4. Коэффициент корреляции Пирсона с помощью numpy
```

```
correlation_numpy = np.corrcoef(zp, ks)[0, 1]

print(f"Коэффициент корреляции Пирсона с помощью numpy:
{correlation_numpy}")



# 5. Коэффициент корреляции Пирсона с помощью pandas

correlation_pandas = pd.Series(zp).corr(pd.Series(ks))

print(f"Коэффициент корреляции Пирсона с помощью pandas:
{correlation_pandas}")
```

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.

Подсказка № 1

Определите объем выборки и рассчитайте среднее значение и стандартное отклонение. Найдите среднее значение IQ и его стандартное отклонение. Обратите внимание, что стандартное отклонение нужно рассчитать как выборочное (`ddof=1`), так как это небольшая выборка.

Подсказка № 2

Выберите правильное распределение для расчета критического значения. Поскольку мы имеем небольшую выборку (менее 30), используйте t-распределение. Найдите критическое значение t для 95% доверительного интервала, используя функцию `stats.t.ppf`.

Подсказка № 3

Рассчитайте стандартную ошибку среднего. Стандартная ошибка среднего вычисляется как выборочное стандартное отклонение, деленное на квадратный корень из объема выборки: SE=$\frac{std\_IQ}{\sqrt{n}}$.

Подсказка № 4

Вычислите погрешность интервала. Умножьте критическое значение t на стандартную ошибку среднего, чтобы получить погрешность интервала. Это значение определяет, насколько можно отклоняться от выборочного среднего.

Подсказка № 5

Сформируйте доверительный интервал. Сложите и вычтите погрешность интервала из выборочного среднего, чтобы получить нижнюю и верхнюю границы доверительного интервала. Убедитесь, что ваши вычисления и формулы точны и соответствуют методике расчета доверительного интервала.

Эталонное решение:

```
import numpy as np

import scipy.stats as stats



# Данные

IQ = [131, 125, 115, 122, 131, 115, 107, 99, 125, 111]

n = len(IQ)

mean_IQ = np.mean(IQ)

std_IQ = np.std(IQ, ddof=1)



# Уровень значимости

alpha = 0.05



# Критическое значение t для 95% доверительного интервала

t_crit = stats.t.ppf(1 - alpha / 2, df=n - 1)



# Доверительный интервал

margin_of_error = t_crit * (std_IQ / np.sqrt(n))

confidence_interval = (mean_IQ - margin_of_error, mean_IQ +
margin_of_error)



print(f"Доверительный интервал для среднего IQ:
{confidence_interval}")
```

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.

Подсказка № 1

Проверьте данные и преобразуйте дисперсию в стандартное отклонение. Дисперсия равна 25, но для расчета доверительного интервала нужно использовать стандартное отклонение. Найдите стандартное отклонение, взяв квадратный корень из дисперсии

Подсказка № 2

Используйте нормальное распределение для расчета критического значения. Поскольку дисперсия генеральной совокупности известна, используйте нормальное распределение для нахождения критического значения Z. В Python это можно сделать с помощью функции `stats.norm.ppf`.

Подсказка № 3

Рассчитайте стандартную ошибку среднего. Стандартная ошибка среднего определяется как стандартное отклонение, деленное на квадратный корень из объема выборки: $SE=\frac{\delta}{\sqrt{n}}$. Это значение будет использовано для расчета погрешности интервала.

Подсказка № 4

Определите погрешность интервала. Умножьте критическое значение Z на стандартную ошибку среднего, чтобы найти погрешность интервала: Margin of Error=z_crit×SE.

Подсказка № 5

Формируйте доверительный интервал. Добавьте и вычтите погрешность интервала от выборочного среднего, чтобы получить нижнюю и верхнюю границы доверительного интервала: (mean_sample−Margin of Error,mean_sample+Margin of Error). Убедитесь, что ваши вычисления точны и формулы корректны.

Эталонное решение:

```
import numpy as np

import scipy.stats as stats
```

```
# Данные

sigma = 5  # Стандартное отклонение (корень из дисперсии)

n = 27

mean_sample = 174.2



# Уровень значимости

alpha = 0.05



# Критическое значение Z для 95% доверительного интервала

z_crit = stats.norm.ppf(1 - alpha / 2)



# Доверительный интервал

margin_of_error = z_crit * (sigma / np.sqrt(n))

confidence_interval = (mean_sample - margin_of_error, mean_sample +
margin_of_error)



print(f"Доверительный интервал для среднего роста:
{confidence_interval}")
```