|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

|  |  |
| --- | --- |
| **ФАКУЛЬТЕТ** | **ИУК «Информатика и управление»** |
| **КАФЕДРА** | **ИУК4 «Программное обеспечение ЭВМ,** |
| **информационные технологии»** | |

**Практическое занятие №7**

**«Критерии согласия»**

**ДИСЦИПЛИНА: «Методы обработки информации»**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Выполнил: студент гр. ИУК4-72Б | |  |  | ( | Сафронов Н.С. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |
| Проверил: | |  |  | ( | Никитенко У.В. | ) |
|  |  |  | (подпись) |  | (Ф.И.О.) |  |

|  |  |
| --- | --- |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | |
|  | - Балльная оценка:  - Оценка: |

Калуга, 2023

**Постановка задачи**

Пусть проверяется простая гипотеза относительно закона распределения для выборки из ПЗ№2 : при различных уровнях значимости α и для различных объемов выборки.

1. объем (любые значений из заданной выборки);

2. объем (полный объем исходной выборки) Используя критерии согласия Колмогорова, (Крамера – фон Мизеса), Пирсона принять или опровергнуть основную гипотезу.

**Результаты выполнения работы**

Используемые критерии значимости: 0.05,0.01,0.001.

Используется полная выборка и 11 случайных значений из выборки.

Статистика критерия согласия Колмогорова рассчитывается по формуле:

Статистика критерия согласия Крамера-фон-Мизеса рассчитывается по формуле:

Статистика критерия согласия Пирсона рассчитывается по формуле:

Для расчета критических значений используется библиотека scipy.stats

**Листинг программы:**

import argparse

import csv

import random

import typing

import numpy as np

import prettytable as pt

import scipy.stats as st

from scipy.stats import norm

def ft(x):

mean\_value = np.mean(x)

std\_deviation = np.std(x, ddof=1)

cdf\_value = norm.cdf(x, loc=mean\_value, scale=std\_deviation)

return cdf\_value

def chi\_squared\_test(points: np.array, expected\_frequencies=None,

bins=None):

if bins is None:

bins = int(np.sqrt(len(points)))

observed\_frequencies, bin\_edges = np.histogram(points, bins=bins)

if expected\_frequencies is None:

expected\_frequencies = (np.ones\_like(observed\_frequencies) \*

len(points) / bins)

chi2 = np.sum(

(observed\_frequencies - expected\_frequencies) \*\* 2

/ expected\_frequencies

)

dof = bins - 1

return chi2, dof

def test\_hypothesis(sample: np.array, alpha: float) -> pt.PrettyTable:

sample = np.sort(sample)

F\_emp = np.arange(1, len(sample) + 1) / len(sample)

F\_theor = ft(sample)

D = np.max(np.abs(F\_emp - F\_theor))

W2 = np.sum((F\_emp - F\_theor) \*\* 2) + 1 / (

12 \* len(sample))

chi2, dof = chi\_squared\_test(sample)

D\_crit = st.kstwobign.ppf(1 - alpha) / np.sqrt(len(sample))

W2\_crit = st.chi2.ppf(1 - alpha, df=1)

chi2\_crit = st.chi2.ppf(1 - alpha, df=len(sample) - 1)

result = pt.PrettyTable()

result.field\_names = ["Критерий", "Статистика", "Критическое значение",

"Вывод"]

result.add\_row(

["Колмогоров", round(D, 4), round(D\_crit, 4),

"Принять" if D < D\_crit else "Отклонить"]

)

result.add\_row(

["Крамер-фон Мизес", round(W2, 4), round(W2\_crit, 4),

"Принять" if W2 < W2\_crit else "Отклонить"]

)

result.add\_row(

["Пирсон", round(chi2, 4), round(chi2\_crit, 4),

"Принять" if chi2 < chi2\_crit else "Отклонить"]

)

return result

if \_\_name\_\_ == "\_\_main\_\_":

parser = argparse.ArgumentParser()

parser.add\_argument("-file")

args = parser.parse\_args()

file = args.file or "./data/Test14.csv"

points = []

with open(file, newline='') as csvfile:

reader = csv.reader(csvfile, delimiter=' ', quotechar='|')

for row in reader:

points.append(float("".join(row)))

points = np.array(points)

alpha = np.array([0.01, 0.05, 0.001])

sample\_k = 11

sample = random.sample(list(points), 11)

sample\_result = []

for a in alpha:

sample\_result.append(test\_hypothesis(sample, a))

all\_points\_k = len(points)

all\_points = points

all\_points\_result = []

for a in alpha:

all\_points\_result.append(test\_hypothesis(all\_points, a))

for i in range(len(alpha)):

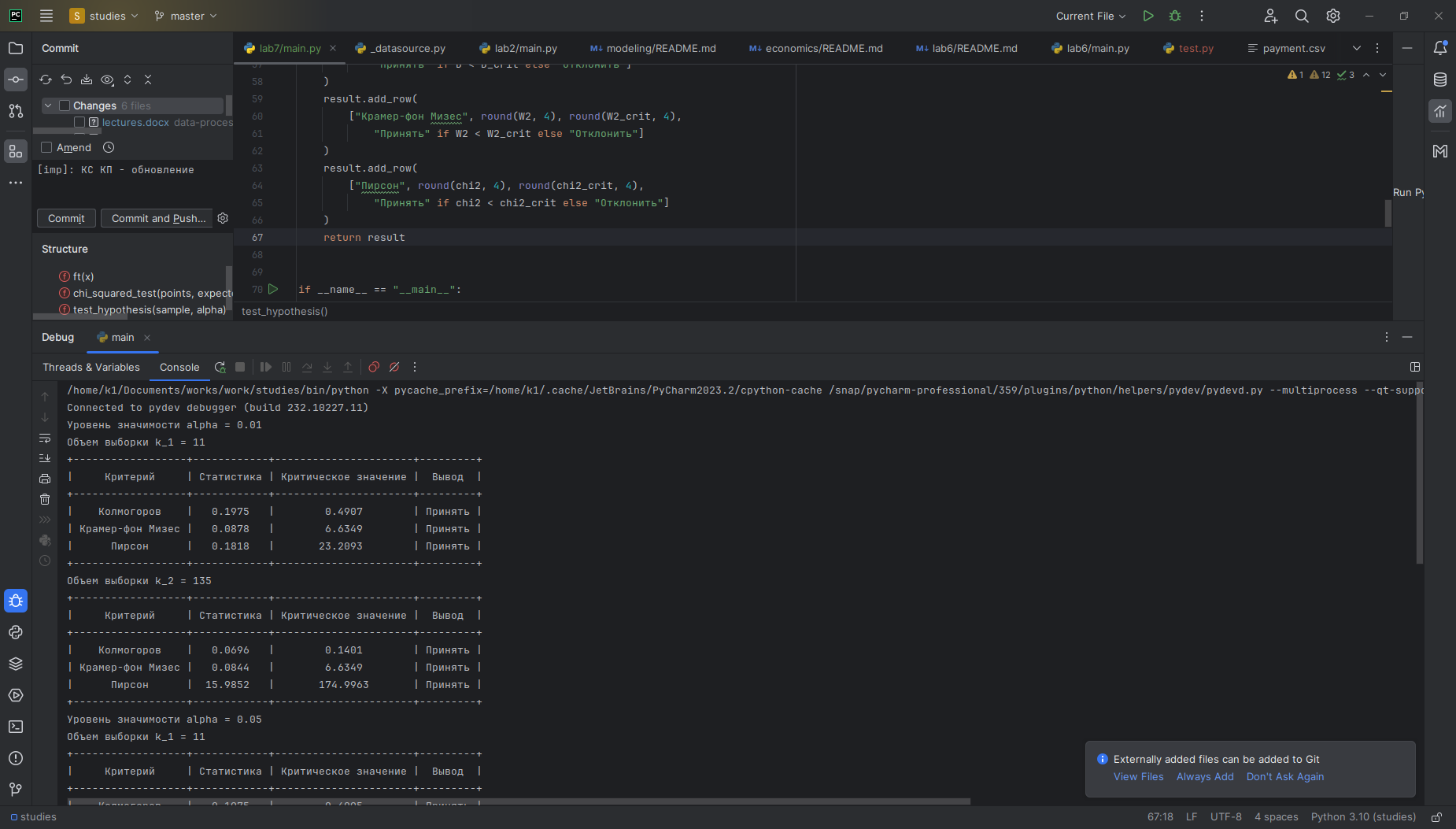
print(f"Уровень значимости alpha = {alpha[i]}")

print(f"Объем выборки k\_1 = {sample\_k}")

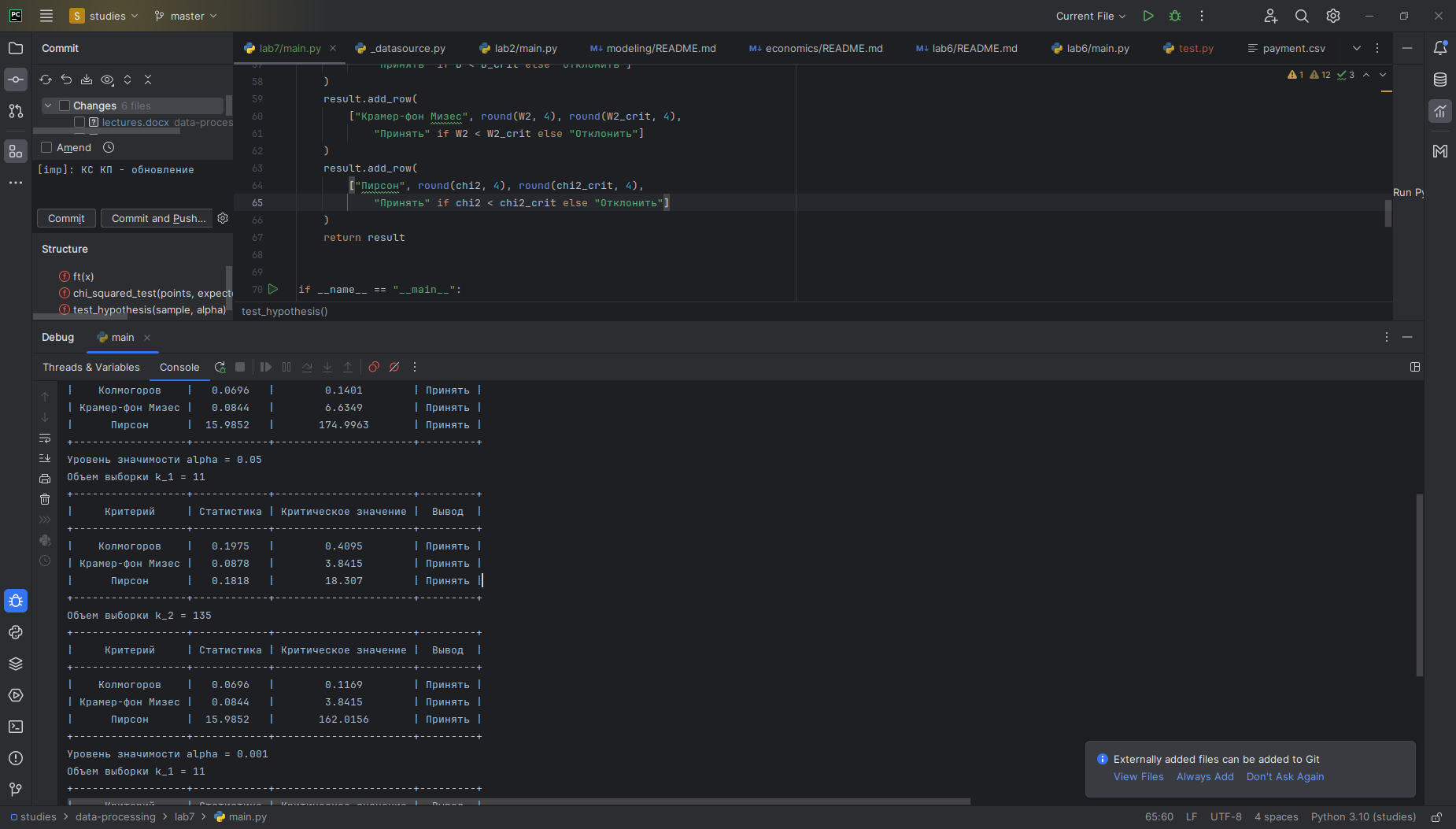
print(sample\_result[i])

print(f"Объем выборки k\_2 = {all\_points\_k}")

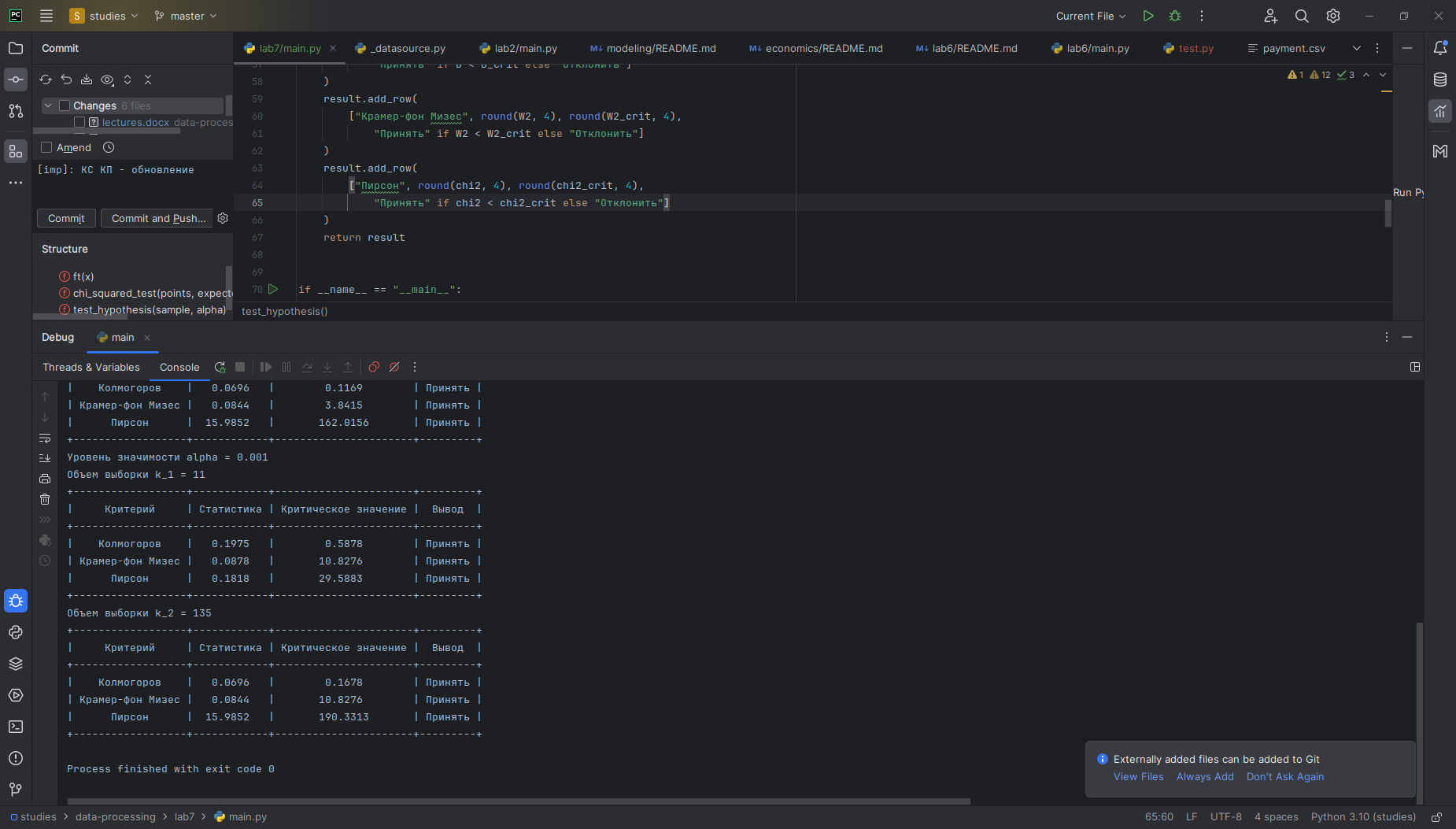
print(all\_points\_result[i])



**Рисунок 1 –** Значения статистики для уровня значимости



**Рисунок 2 –** Значения статистики для уровня значимости



**Рисунок 3 –** Значения статистики для уровня значимости