



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«МИРЭА – Российский технологический университет»

РТУ МИРЭА

Институт информационных технологий (ИИТ)

Кафедра прикладной математики (ПМ)

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №4

по дисциплине «Языки программирования для статистической обработки
данных»

Студент группы

ИМБО-11-23 Журавлев Ф. А.

(подпись)

Преподаватель

Трушин С. М.

(подпись)

Москва 2025 г.

1 ЦЕЛЬ И ЗАДАЧИ

Цель практической работы:

Научиться рассчитывать основные статистические показатели и проводить проверку гипотез с использованием Python, R.

Задачи практической работы:

1. Рассчитать основные статистические показатели:

- Среднее, медиана, мода, дисперсия, стандартное отклонение.
- Python: `scipy.stats`, R: `summary`, `stats`.

2. Выполнить проверку гипотез:

- t-тест (для сравнения двух выборок). • U-критерий Манна-Уитни (для несвязанных выборок).
- Хи-квадрат тест (для проверки независимости).

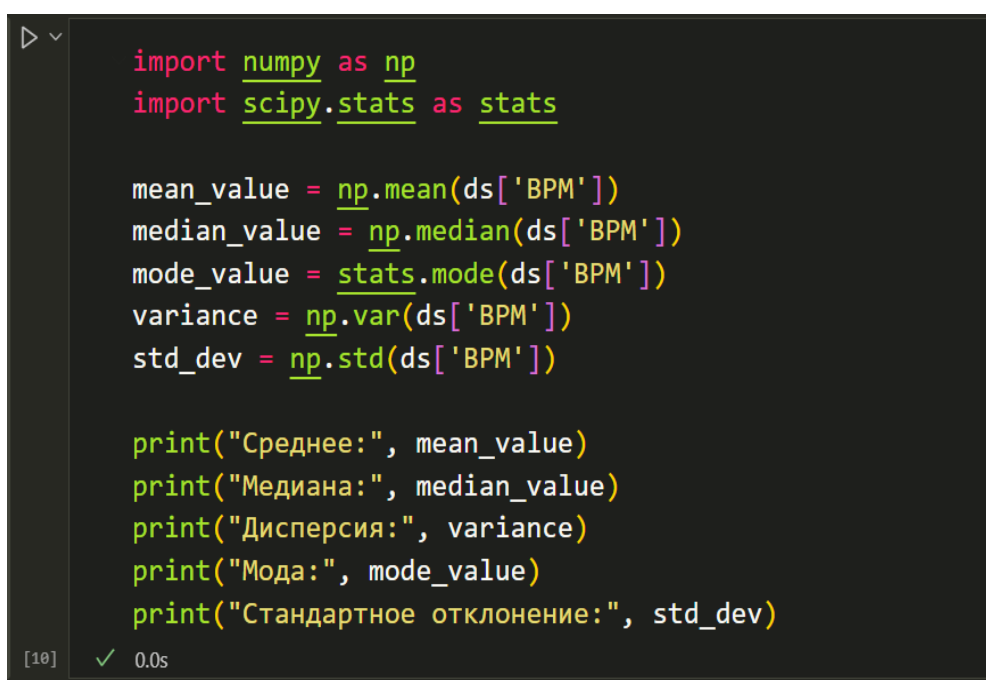
3. Сравнить результаты между Python, R.

2. РЕЗУЛЬТАТЫ ПРАКТИЧЕСКОЙ РАБОТЫ

2.1 Расчет статистических показателей в Python

После загрузки данных в Python, напомним коды, которые помогут рассчитать различные статистические показатели.

рисунок 2.1.1 – статистических показателей



```
import numpy as np
import scipy.stats as stats

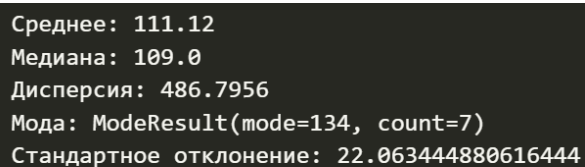
mean_value = np.mean(ds['BPM'])
median_value = np.median(ds['BPM'])
mode_value = stats.mode(ds['BPM'])
variance = np.var(ds['BPM'])
std_dev = np.std(ds['BPM'])

print("Среднее:", mean_value)
print("Медиана:", median_value)
print("Дисперсия:", variance)
print("Мода:", mode_value)
print("Стандартное отклонение:", std_dev)
```

[10] ✓ 0.0s

Далее посмотрим какие мы получили значения среднего, медианы, дисперсии, моды и стандартного отклонения.

рисунок 2.1.2 – значения показателей.



```
Среднее: 111.12
Медиана: 109.0
Дисперсия: 486.7956
Мода: ModeResult(mode=134, count=7)
Стандартное отклонение: 22.063444880616444
```

Далее рассмотрим код, благодаря которому сможем проанализировать показатели с помощью критерия Мана-Уитни и Т-теста.

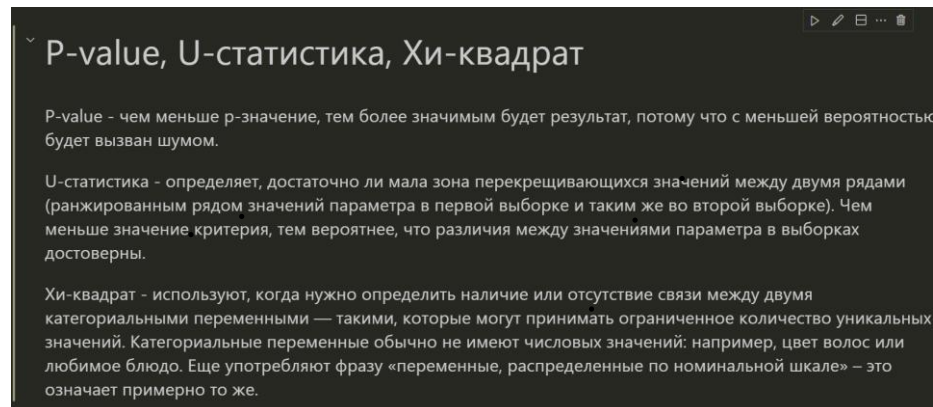


Рисунок 2.1.3 – код T-test

```
group1 = df[df['Жанр'] == 'рок']['BPM']
group2 = df[df['Жанр'] == 'хип-хоп']['BPM']
t_stat, p_value = stats.ttest_ind(group1, group2)
print("t-статистика:", t_stat, "p-значение:", p_value)
```

[14] ✓ 0.0s

... t-статистика: 16.036541965221023 p-значение: 1.3304053172087315e-32

Рисунок 2.1.4 – код Ман-Уитни.

```
u_stat, p_value = stats.mannwhitneyu(group1, group2)
print("U-статистика:", u_stat, "p-значение:", p_value)
```

[15] ✓ 0.0s

... U-статистика: 4268.5 p-значение: 1.8398245520485363e-21

Рисунок 2.1.5 – Хи-квадрат тест.

```
contingency_table = pd.crosstab(df['Жанр'], df['Настроение'])
chi2_stat, p_value, dof, expected = stats.chi2_contingency(contingency_table)
print("Хи-квадрат:", chi2_stat, "p-значение:", p_value)
```

✓ 0.0s

Хи-квадрат: 4.412879687057155 p-значение: 0.6209874571848064

2.2 Статистические данные в R.

Теперь рассмотрим реализацию подобного кода в R.

Рисунок 2.2.1 – Статистические данные в R

```
1 library(readr)
2 ds <- read.csv("D:/Documents/Learning/3/R/music_genre_dataset.csv")
3 summary(ds$BPM)
4 library(modeest)
5 mode_value <- mfv(ds$BPM)
6 print(paste("Мода: ", mode_value))
7 variance <- var(ds$BPM)
8 std_dev <- sd(ds$BPM)
9 print(paste("Дисперсия", variance))
10 print(paste("Отклонение", std_dev))
```

Рисунок 2.2.2– Вывод статистических данных в R

```
The downloaded binary packages are in
  C:\Users\fedor\AppData\Local\Temp\RtmpGyQqce\downloaded_packages
> source("d:\\Documents\\Learning\\3\\R\\4\\4.r", encoding = "UTF-8")
[1] "Мода: 134"
[1] "Дисперсия 489.241809045226"
[1] "Отклонение 22.1188112032547"
> 
```

Теперь рассмотрим код, который показывает нам значения всех тестов, что мы использовали ранее. Статистические характеристики t-test, и статистику нельзя выполнить на моих данных в R, так как у меня более 2 уникальных категориальных данных (3). Если я запущу код, то получу ошибку:

Рисунок 2.2.4 – Ошибка

```
Error in t.test.formula(ds$Жанр ~ ds$BPM) :  
  grouping factor must have exactly 2 levels  
> 
```

Рисунок 2.2.4 – Хи-квадрат тест.

```
13  ctable <- table(ds$Жанр, ds$`Настроение`)  
14  ctable1 <- chisq.test(ctable)  
15  print(ctable1)  
16
```

Рисунок 2.2.5 — Хи-квадрат тест, вывод

```
Pearson's Chi-squared test  
  
data:  ctable  
X-squared = 4.4129, df = 6, p-value = 0.621
```

ИТОГИ И ВЫВОДЫ:

В результате практической работы были проведены поиски статистических значений с помощью различных функций в обоих языках программирования, а также с помощью разных тестов таких, как t-test, Хи-квадрат тест и критерий Мана-Уитни. Работа со всеми этими характеристиками в разы удобнее на языке программирования Python. Так как в R нельзя анализировать любые данные произвольного размера для нахождения статистических характеристик (t-test, u-статистика).