

Преамбула

Ноутбук: HUAWEI MateBook D 16
Модель: 53013ESY
Процессор: 12th Gen Intel Core i7-12700H Processor
Оперативная память: 16 ГБ
ОС: Windows 11 Домашняя
Версия: 22H2
Входные данные для SATSolvera
Результаты замеров

Подготовка

Минимальную и максимальную частоту процессора выставила на 100%, выбрала режим питания "максимальная производительность кэш очистила, закрыла все приложения, подключила ноутбук к питанию и держала на весу, чтобы обеспечить лучшее охлаждение.

Импортировала библиотеки для дальнейшей работы:

```
import matplotlib.pyplot as plt
from scipy.stats import normaltest, norm, sem
import statistics
```

Тест на нормальность

```
with open('time1.txt', 'r') as file:
    time1 = list(map(float, file.readlines()))
print(normaltest(time1))
```

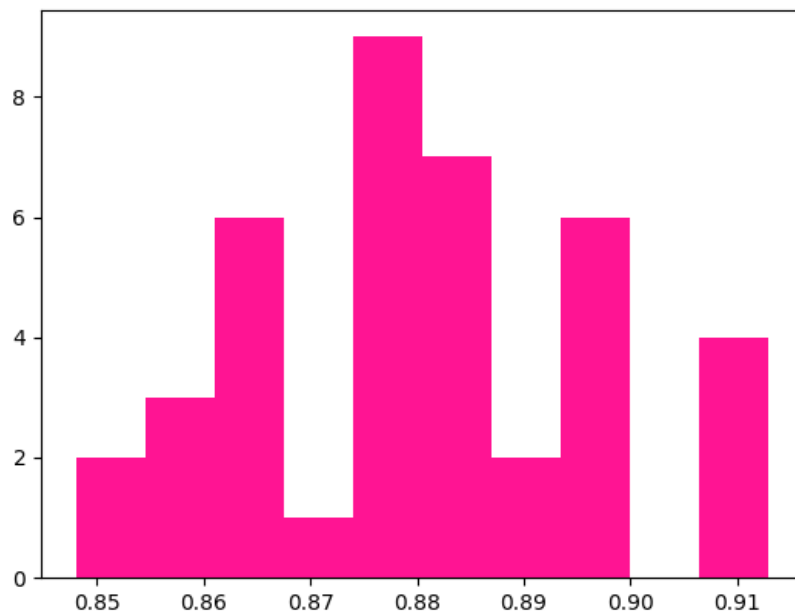
Output:

```
NormaltestResult(statistic=0.3913489524734149,
pvalue=0.822279862751577)
```

$pvalue > 0.05$, то есть тест на нормальность пройден

График

```
plt.hist(time1, color='#FF1493')
plt.show()
```



Используемые формулы

Среднее арифметическое: $\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$

Стандартное отклонение: $\sigma_{c.o.} = \sqrt{\frac{\sum_{i=1}^n (\bar{x} - x_i)^2}{n-1}}$

Стандартное отклонение среднего: $\sigma_{c.o.c.} = \frac{\sigma_{c.o.}}{\sqrt{n}}$

n — размер выборки, x_i — значения выборки

Подсчёты

```
avg = statistics.mean(time1)
print('Среднее арифметическое: ', avg)

st_dev = statistics.stdev(time1)
print('Стандартное отклонение: ', st_dev)

st_dev_sem = sem(time1)
print('Стандартное отклонение среднего: ',
      st_dev_sem)

ratio = float('%s' % float('%0.1g' % (st_dev_sem /
    avg * 100)))
print(f'Отношение среднего отклонения к среднему:
    ~{ratio}%')

conf_interval = norm.interval(confidence=0.95, loc
    =avg, scale=st_dev_sem)
len_conf = conf_interval[1] - conf_interval[0]
print('Половина длины доверительного интервала: ',
      len_conf/2)
print(f'95%-доверительный интервал: {"%s" % float
    ("%0.3g" % avg)} ± '
      f'{"%s" % float("%0.1g" % (len_conf / 2))}')

print('Половина длины предсказывающего интервала:
    ', 2 * st_dev)
```

```
print(f'95%-предсказывающий интервал: {"%s" %  
      float("%.2g" % avg)} ± '  
      f'{"%s" % float("%.g" % (2 * st_dev))}'')
```

Output:

```
Среднее арифметическое: 0.8806  
Стандартное отклонение: 0.016266687683887237  
Стандартное отклонение среднего:  
0.002571989153384636  
Отношение среднего отклонения к среднему: ~2.0%  
Половина длины доверительного интервала:  
0.005041006109261503  
95%-доверительный интервал: 0.881 ± 0.005  
Половина длины предсказывающего интервала:  
0.032533375367774474  
95%-предсказывающий интервал: 0.88 ± 0.03
```

Отношение стандартного отклонения к среднему $< 15\%$, то есть стандартное отклонение не слишком большое, то есть хорошо