

```
Ввод [1]: 1 from PIL import ImageGrab
2 from IPython.display import display, Image
3 def ins(ratio=1.0):
4     im_data = ImageGrab.grabclipboard()
5     new_size = tuple([int(i*ratio) for i in im_data.size])
6     thumb = im_data.resize(new_size)
7     fn = "temp.PNG"
8     thumb.save(fn)
9     img = Image(filename=fn)
10    display(img)
```

```
Ввод [3]: 1 ins(1)
```

При подбрасывании N раз симметричной монеты наблюдались следующие статистические закономерности (описываемые в литературе) появления герба $\hat{p}(A) = \frac{N(A)}{N}$:

Автор эксперимента	N	$\hat{p}(A) = \frac{N(A)}{N}$
Бюффон (1707–1788)	4040	0,507
Де Морган (1806-1871)	4090	0,5005
Джевонс (1835-1882)	20 480	0,5068
Пирсон К. (1857-1936)	24 000	0,5005
Студент ПМ2024	100 000	???

```
Ввод [2]: 1 from scipy.stats import *
2 import numpy as np
```

```
Ввод [3]: 1 X=randint(0,2) # Задаем равномерное дискретное распределение, "0" -- появление "
```

```
Ввод [4]: 1 X.rvs(size=10)
```

```
Out[4]: array([1, 1, 1, 1, 1, 1, 0, 0, 0, 1], dtype=int64)
```

```
Ввод [5]: 1 N=1000000
2 sample=X.rvs(size=N)
3 np.array(sample,dtype=int)
```

```
Out[5]: array([1, 0, 0, ..., 0, 1, 1])
```

```
Ввод [6]: 1 N_A=len(sample[sample==0])
2 N_A
```

```
Out[6]: 500565
```

```
Ввод [7]: 1 hat_p=N_A/N
2 hat_p
```

```
Out[7]: 0.500565
```

Ввод [13]:

```
1 ins(1)
```

Примеры

1. Пусть симметричная игральная кость бросается дважды. Найти вероятность события A , состоящего в том, что сумма выпавших очков равна 5.

Решение. Пространство элементарных исходов

$\Omega = \{\omega_{ij} = (i, j) : i = \overline{1, 6}; j = \overline{1, 6}\}$, $n = |\Omega| = 6^2 = 36$, причём

$\mathbb{P}(\omega_{ij}) = \frac{1}{36}$ (все исходы равновероятны). Событие

$A = \{\omega_{14} = (1, 4), \omega_{23} = (2, 3), \omega_{32} = (3, 2), \omega_{41} = (4, 1)\}$, $m = |A| = 4$.

Следовательно, $\mathbb{P}(A) = \frac{4}{36} = \frac{1}{9}$.

Ввод [8]:

```
1 def union(A, B):
2     "Соединяем элемент A с каждым элементом из B."
3     return {a + b for a in A for b in B}
4
5 Omega = np.array(list(union('123456', '123456')))
6 Omega
```

Out[8]: array(['61', '65', '51', '32', '12', '22', '26', '46', '52', '45', '21',
'33', '24', '66', '54', '35', '55', '14', '23', '13', '15', '34',
'63', '25', '42', '16', '44', '31', '64', '62', '36', '41', '56',
'43', '53', '11'], dtype='<U2')

Ввод [9]:

```
1 len(Omega)
```

Out[9]: 36

Ввод [10]:

```
1 Omega = np.array(list(union('123456', '123456')))
2 sum_Omega=np.array([int(a[0])+int(a[1]) for a in Omega])
3 A=Omega[sum_Omega ==5]
4 n_A=len(A)
5 n_Omega=len(Omega)
6 P_A=n_A/n_Omega
7 P_A # Классическое определение вероятности
```

Out[10]: 0.1111111111111111

Ввод [11]:

```
1 N=1000000
2 sample=np.random.choice(Omega,size=N,replace=True)
3 sum_sample=np.array([int(a[0])+int(a[1]) for a in sample])
4 N_A=len(sample[sum_sample==5])
5 pstat=N_A/N
6 pstat # Статистическое определение вероятности
```

Out[11]: 0.112252

Ввод [12]:

```
1 from sympy import *
2 import locale as loc
3 loc.setlocale(loc.LC_ALL, 'ru')
4 init_printing(use_unicode=True,use_latex=True)
```

Ввод [13]:

```
1 from IPython.display import display, Math, Latex
```

```

Ввод [14]: 1 import matplotlib.pyplot as plt
2 import matplotlib.ticker as ticker
3 from matplotlib import rcParams
4 #####
5 import locale
6 locale.setlocale(locale.LC_NUMERIC, 'russian')
7 plt.rcParams['axes.formatter.use_locale'] = True
8 #####
9 #####
10 plt.rcParams['font.size'] = 36
11 plt.rcParams['font.family'] = "Times New Roman"
12 plt.rcParams['mathtext.fontset'] = 'cm'
13 #####

```

```

Ввод [15]: 1 sample=list(np.random.choice(Omega,1,replace=True))
2 sum_sample=np.array([int(a[0])+int(a[1]) for a in sample])
3 sample, sum_sample

```

```
Out[15]: (['11'], array([2]))
```

```

Ввод [16]: 1 sum_sample ==5

```

```
Out[16]: array([False])
```

```

Ввод [17]: 1 np.count_nonzero(sum_sample == 5)

```

```
Out[17]: 0
```

```

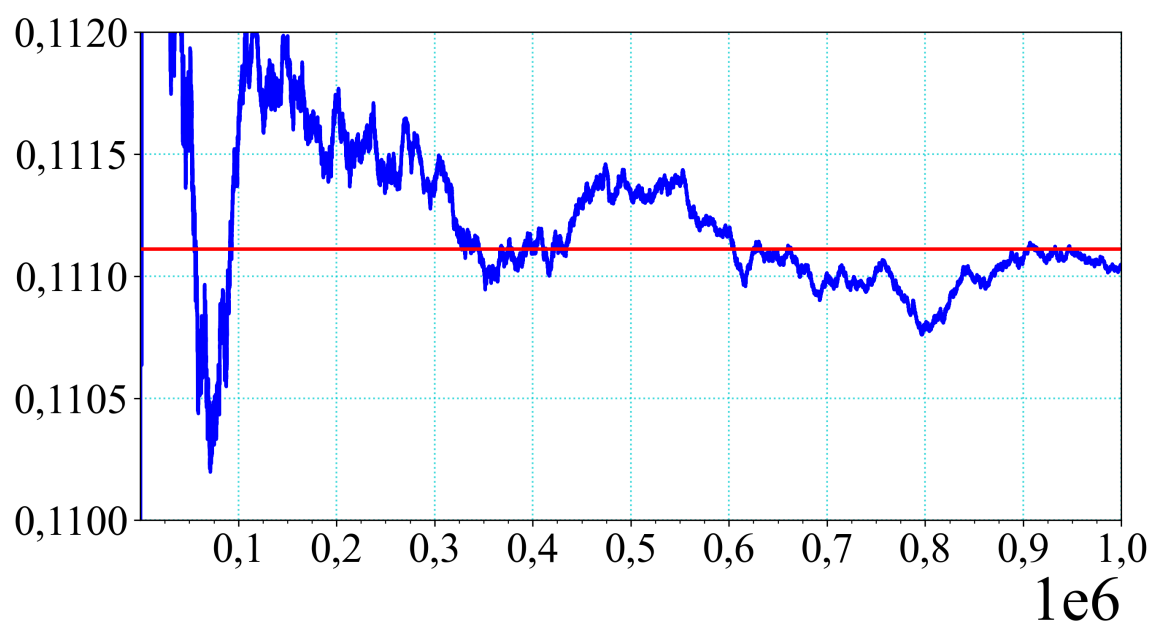
Ввод [18]: 1 count=0
2 N=1000000
3 x=[]
4 y=[]
5 for i in range(N):
6     x.append(i+1)
7     #omega=random.sample(Omega, k=6)
8     #omega=list(np.random.choice(Omega,6,replace=False)) #(без возвращения)
9     #omega=list(np.random.choice(Omega,1,replace=True)) #(с возвращением)
10    sample=list(np.random.choice(Omega,1,replace=True))
11    sum_sample=np.array([int(a[0])+int(a[1]) for a in sample])
12    if sum_sample == 5:
13        count+=1
14    y.append(count/(i+1))
15 pstatA=count/N
16 print(pstatA)

```

```
0.110773
```

Ввод [19]:

```
1 rcParams['figure.figsize'] = (30, 20)
2 rcParams['figure.dpi'] = 300
3 import matplotlib.pyplot as plt
4 fig,ax =plt.subplots(figsize=(10, 5))
5 plt.tick_params(labelsize = 24)
6 plt.grid(color='DarkTurquoise', alpha=0.75, linestyle=':', linewidth=1)
7 #ax.xaxis.set_major_locator(ticker.MaxNLocator(10))
8 #ax.xaxis.set_minor_locator(ticker.MaxNLocator(10))
9 #ax.xaxis.set_major_locator(ticker.IndexLocator(base=10, offset=10))
10 #ax.xaxis.set_major_locator(ticker.LogLocator(base=2, numticks=1000))
11 #ax.yaxis.set_major_locator(ticker.MaxNLocator(8))
12 ax.xaxis.set_major_locator(ticker.MaxNLocator(10))
13 ax.xaxis.set_minor_locator(ticker.MaxNLocator(40))
14 plt.ylim(0.11,0.112)
15 plt.xlim(10**2, 10**6)
16 #####
17 plt.plot(x,y,color='b',lw=2)
18 plt.plot(x,[1/9]*len(x),color='r',lw=2)
19 plt.show()
20 fig.savefig("pstatA.pdf", bbox_inches='tight')
```



Ввод []:

1