

```
In [1]: %run Mod_sv_var22_Denisov_Kod.ipynb
```

В сбербанк внесли деньги $N \gg 1$ человек.

Сумма вклада каждого из них есть с.в. η со средним A и дисперсией B .

С.в. η — общая сумма вклада.

```
In [2]: N = int(input("Введите количество людей, внесших деньги в сбербанк: "))
A = float(input("Введите среднее значение: "))
B = float(input("Введите дисперсию: "))
size = int(input("Введите количество реализаций случайной величины: "))
```

```
Введите количество людей, внесших деньги в сбербанк: 100
Введите среднее значение: 1
Введите дисперсию: 1
Введите количество реализаций случайной величины: 1000
```

"Розыгрыш" значений случайной величины.

```
In [3]: %%time
#pd.set_option("display.max_rows", 3000)
theta = Worker(N, A, B, size)
theta.gauss_distribution = theta.gauss_distribution()
x = pd.DataFrame({"Реализации случайной величины": theta.gauss_distribution})
x
```

```
CPU times: total: 93.8 ms
Wall time: 93.5 ms
```

```
Out[3]:
```

Реализации случайной величины	
0	68.779943
1	71.318734
2	72.139788
3	73.935113
4	75.073770
...	...
995	124.977418
996	125.409326
997	125.452353
998	126.954078
999	134.857377

1000 rows × 1 columns

Определим теоретические и выборочные числовые характеристики:

```
In [4]: N, A, B, size
```

```
Out[4]: (100, 1.0, 1.0, 1000)
```

```
In [5]: theta.statistical_characteristics()
```

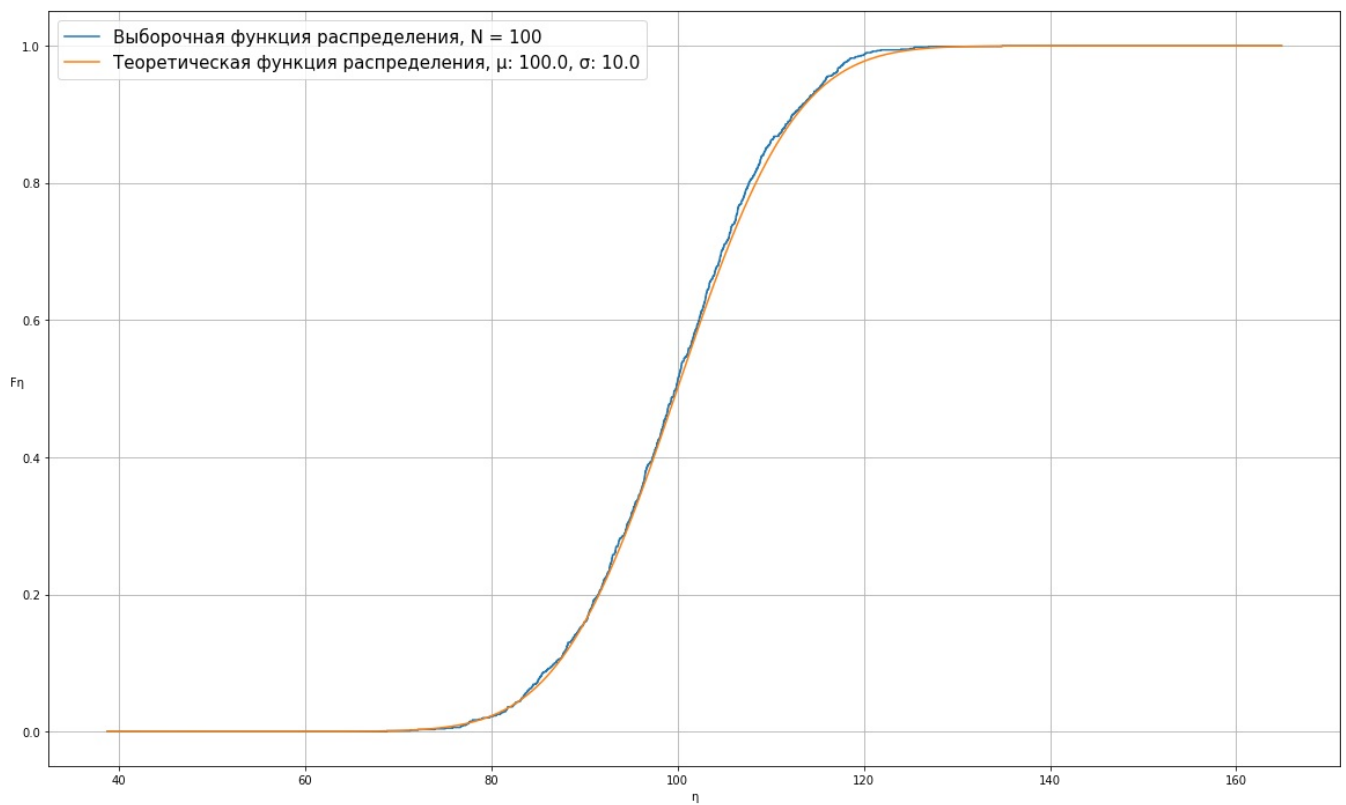
```
Out[5]:
```

	$E\eta$	\bar{x}	$ E\eta - \bar{x} $	$D\eta$	S^2	$ D\eta - S^2 $	\hat{M}_e	\hat{R}	D
1	100.0	99.654385	0.345615	100.0	95.039654	4.960346	99.845549	66.077434	0.022128

```
In [6]: D = str(theta.statistical_characteristics()['D']).split('\nName')[0].split(" ")[-1]
print(f'Размер выборки: {theta.size}, ')
print(f'Мера расхождения теоретической и эмпирической плотности распределения: {D}')
```

```
Размер выборки: 1000,
Мера расхождения теоретической и эмпирической плотности распределения: 0.022128
```

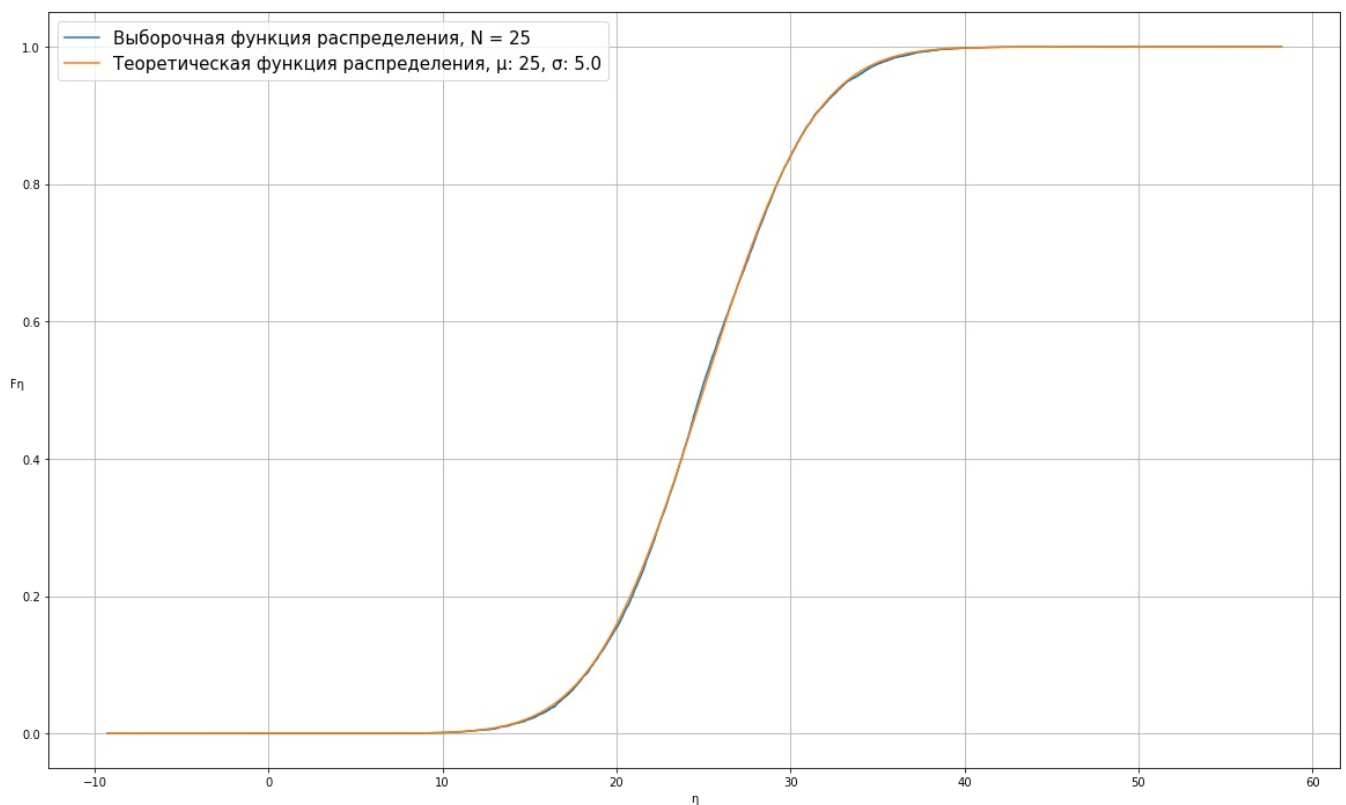
```
In [7]: theta.check_distr()
```



```
In [8]: # Сходимость D
N, A, B, size = 30, 1, 1, 30000
theta = Worker(N, A, B, size)
theta.gauss_distribution = theta.gauss_distribution()
display(theta.statistical_characteristics())
```

	$E\eta$	\bar{x}	$ \bar{E\eta} - \bar{x} $	$D\eta$	S^2	$ D\eta - S^2 $	\hat{Me}	\hat{R}	D
1	30	29.97299	0.02701	30	30.040527	0.040527	29.968945	45.5372	0.004232

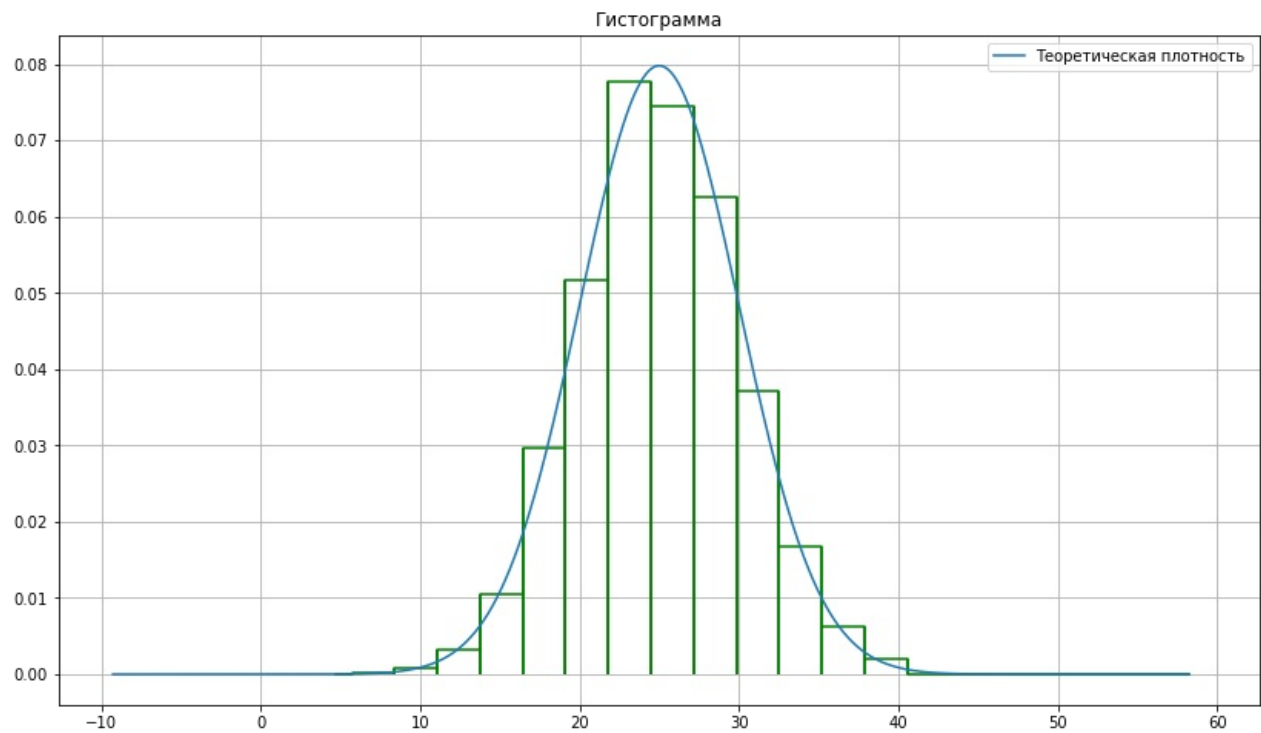
```
In [9]: # Сходимость графиков
N, A, B, size = 25, 1, 1, 10000
theta = Worker(N, A, B, size)
theta.gauss_distribution = theta.gauss_distribution()
display(theta.check_distr())
```



None

Построим гистограмму и таблицу

```
In [10]: # Гистограмма
theta.set_intervals() # True - ручное задание интервалов
theta.hist()
```



```
In [11]: # Таблица
theta.create_table()
```

Out[11]:

	info	1	2	3	4	5	6	7	8	9	10	11	12	
0	delta	$(-\infty, 5.7)$	[5.7,8.4)	[8.4,11.1)	[11.1,13.8)	[13.8,16.4)	[16.4,19.1)	[19.1,21.8)	[21.8,24.5)	[24.5,27.2)	[27.2,29.8)	[29.8,32.5)	[32.5,35.2)	[35.2,37.9)
1	zk	-inf	7.069395	9.747365	12.425335	15.103305	17.781275	20.459245	23.137215	25.815185	28.493155	31.171125	33.849095	36.527065
2	fn	0.0	0.000129	0.000761	0.003377	0.011251	0.02814	0.052826	0.074439	0.078735	0.062511	0.037253	0.016664	0.003377
3	h	0.0	0.000112	0.000747	0.003323	0.010418	0.029612	0.051718	0.077671	0.074497	0.062659	0.037155	0.016804	0.003377

Полная информация о данных

```
In [12]: theta.get_all_info()
```

Реализации случайной величины	
0	5.730410
1	7.267023
2	8.198263
3	8.691114
4	8.811187
...	...
9995	42.048551
9996	42.074436
9997	42.372025
9998	42.584983
9999	43.221991

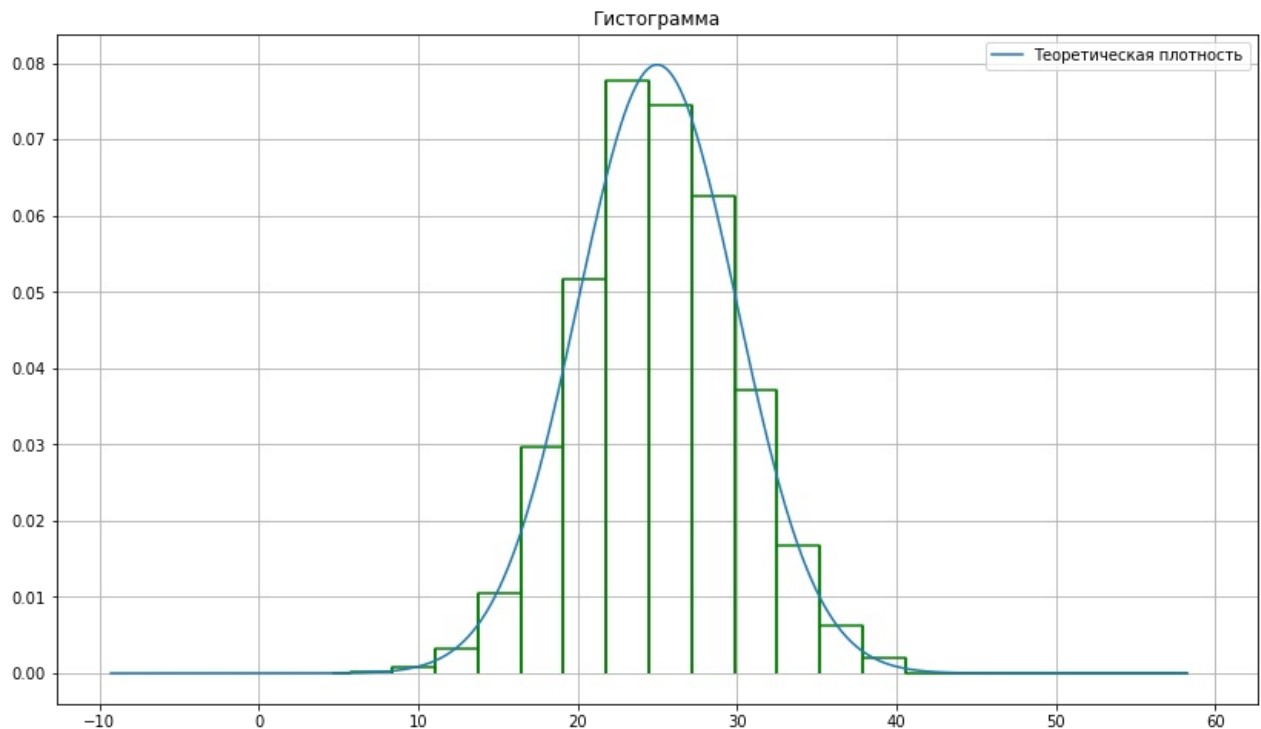
10000 rows × 1 columns

Размер выборки: 10000, Математическое ожидание: 25, Дисперсия: 5.0
Мера расхождения теоретической и эмпирической функции распределения: 0.009882
Мера расхождения теоретической плотности распределения и гистограммы: 0.004238315010302807
Статистические характеристики:

	$E\eta$	\bar{x}	$ E\eta - \bar{x} $	$D\eta$	S^2	$ D\eta - S^2 $	\hat{Me}	\hat{R}	D
1	25	25.0381	0.0381	25	24.977448	0.022552	24.890088	37.491581	0.009882

Характеристики для гистограммы

	info	1	2	3	4	5	6	7	8	9	10	11	12	
0	delta	$(-\inf, 5.7)$	[5.7,8.4)	[8.4,11.1)	[11.1,13.8)	[13.8,16.4)	[16.4,19.1)	[19.1,21.8)	[21.8,24.5)	[24.5,27.2)	[27.2,29.8)	[29.8,32.5)	[32.5,35.2)	[35.2,37.9)
1	zk	-inf	7.069395	9.747365	12.425335	15.103305	17.781275	20.459245	23.137215	25.815185	28.493155	31.171125	33.849095	36.527065
2	fn	0.0	0.000129	0.000761	0.003377	0.011251	0.02814	0.052826	0.074439	0.078735	0.062511	0.037253	0.016664	0.003377
3	h	0.0	0.000112	0.000747	0.003323	0.010418	0.029612	0.051718	0.077671	0.074497	0.062659	0.037155	0.016804	0.003377



```
In [13]: %%time
t = 100
result = [0, 0]
for i in range(t):
    N, A, B, size = 50, 1, 1, 1000
    #pd.set_option("display.max_rows", 3000)
    theta = Worker(N, A, B, size)
    theta.gauss_distribution = theta.gauss_distribution()
    # theta.set_intervals()
    theta.set_significance_level() # alpha

    #x = pd.DataFrame({"Реализации случайной величины": theta.gauss_distribution})
    theta.hypothesis_verdict()

    # theta.hist()
    if theta.nF_R0 > theta.alpha:
        # print(f'!F(R0) = {theta.nF_R0} > {theta.alpha} гипотеза принята')
        result[0] += 1
    else:
        result[1] += 1
        # print(f'!F(R0) = {theta.nF_R0} < {theta.alpha} гипотеза отвергнута')

print(f'alpha = {theta.alpha}\nКоличество запусков: {t}\nГипотеза принята {result[0]} раз\nОтклонена {result[1]} раз')
print(f'\nСтатистики для гипотезы:\nТеоретические вероятности: {theta.q_i}\nИнтервалы: {theta.intervals_q}\nЧастоты: {theta.h_i}')

alpha = 0.5
Количество запусков: 100
Гипотеза принята 55 раз
Отклонена 45 раз

Статистики для гипотезы:
Теоретические вероятности: [0.25, 0.25, 0.25, 0.25]
Интервалы: [45.23066262250723, 49.999932565042386, 54.76933737749277]
Частоты: [258, 271, 233, 238]
CPU times: total: 6.36 s
Wall time: 6.41 s
```

Processing math: 100%