

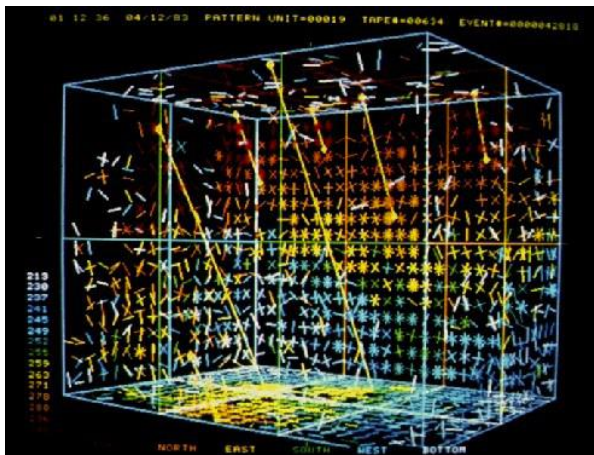
Detekce neutrin

1. Jaká je pravděpodobnost, že v jednom intervalu bude detekováno 8 nebo více neutrin?

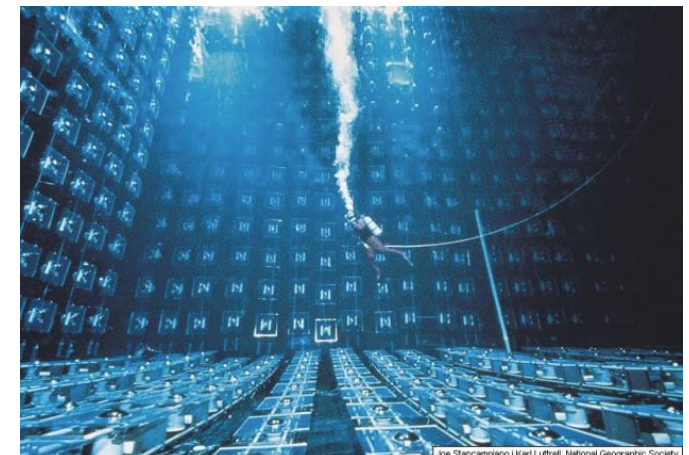
Irvine-Michigan-Brookhaven, 23. 2. 1987

detekce neutrin: interval 10 s

No. of events	0	1	2	3	4	5	6	7	8
No. of intervals	1042	860	307	78	15	3	0	0	1



- detektor Čerenkovova záření
- bazén $17 \times 17.5 \times 23 \text{ m}^3$ (684 000 l) ultra čisté vody
- v solném dolu 600 m pod zemí
- 2048 fotonásobičů



Detekce neutrin

1. Jaká je pravděpodobnost, že v jednom intervalu bude detekováno 8 nebo více neutrin?

Irvine-Michigan-Brookhaven, 23. 2. 1987

detekce neutrin: interval 10 s

No. of events	0	1	2	3	4	5	6	7	8
No. of intervals	1042	860	307	78	15	3	0	0	1
Poisson prediction	1061	824	320	83	16	2	0.3	0.04	0.003

Vážený průměr:

$$(0 \times 1042 + 1 \times 860 + 2 \times 307 + 3 \times 78 + 4 \times 15 + 5 \times 3 + 6 \times 0 + 7 \times 0 + 8 \times 1) / (1042 + 860 + 307 + 78 + 15 + 3 + 1) = 0.777$$

Poissonovo rozdělení: $\nu = n/N = 0.777$

Počet intervalů: $n = 2306$

Počet neutrin: $N = 1791$

$$P = 1.7 \times 10^{-6}$$

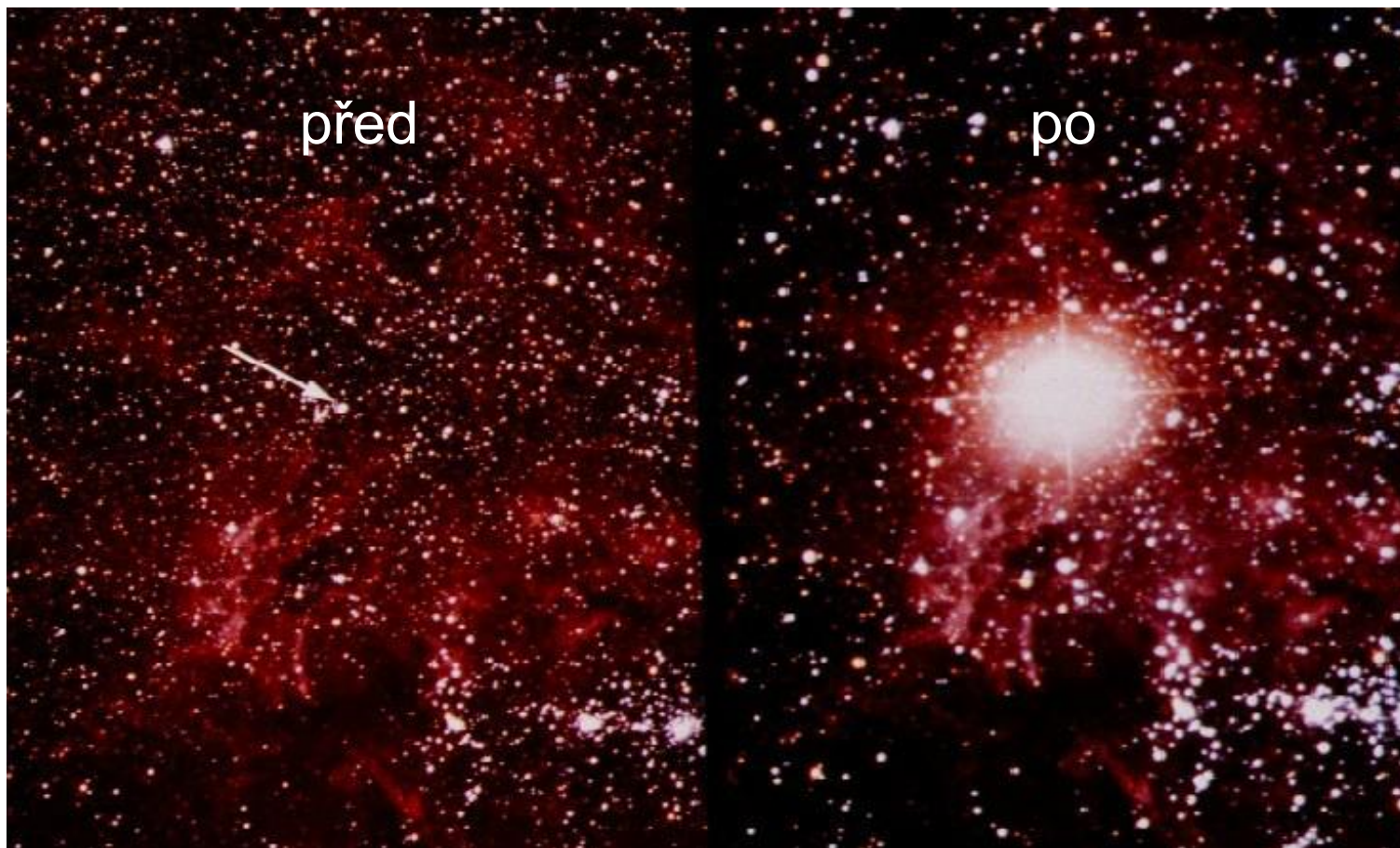
supernova S1987a

Detekce neutrin

1. Jaká je pravděpodobnost, že v jednom intervalu bude detekováno 8 nebo více neutrin?

Irvine-Michigan-Brookhaven, 23. 2. 1987

detekce neutrin: interval 10 s



6	7	8
0	0	1
0.3	0.04	0.003

$P = 1.7 \times 10^{-6}$

supernova S1987a

záblesk neutrin ~ 2.5 h
před světelným zábleskem

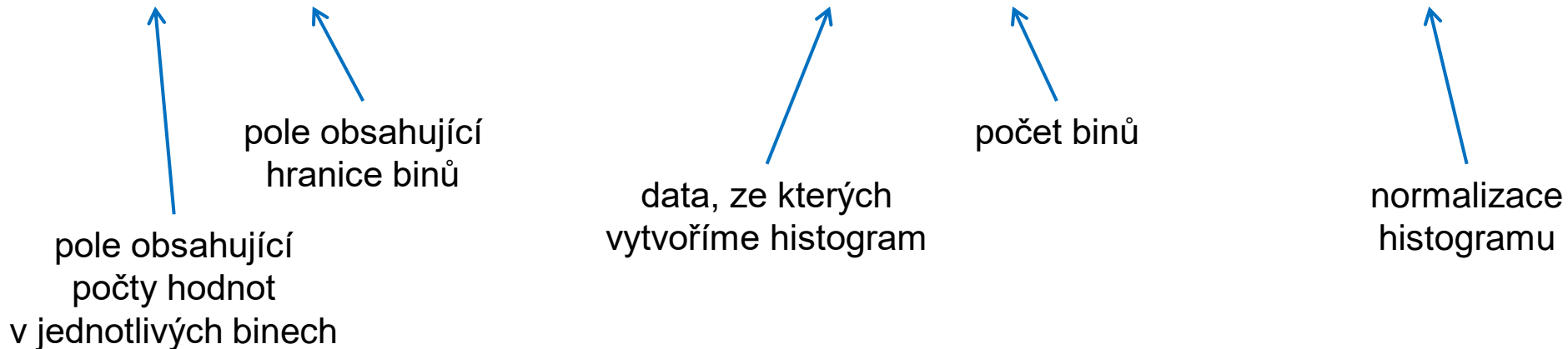
Chyba výšky binu histogramu

2. Zjistěte, jak závisí chyba výšky i -tého binu histogramu na celkovém počtu naměřených hodnot.

Práce s histogramy v Pythonu:

- vytvoření histogramu

```
hist, bin_edges=np.histogram(data, bins=nbins, density='True')
```



2. Zjistěte, jak závisí chyba výšky i -tého binu histogramu na celkovém počtu naměřených hodnot.

- Zvol 50 různých hodnot N_{tot} – celkový počet hodnot v histogramu.

$$\text{např. } N_{tot} = \{100, 200, 300, \dots, 5000\}$$

- Pro každou hodnotu N_{tot} proved' 100 simulací histogramu se 100 biny.

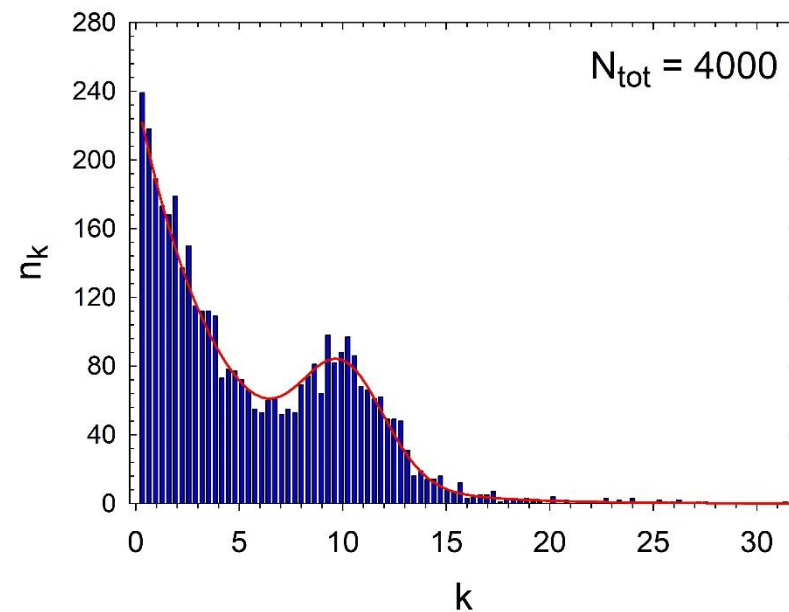
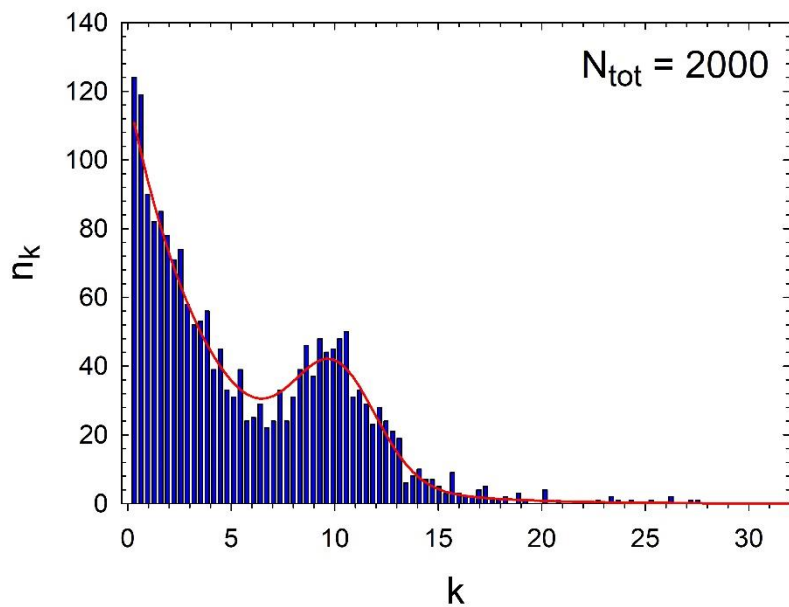
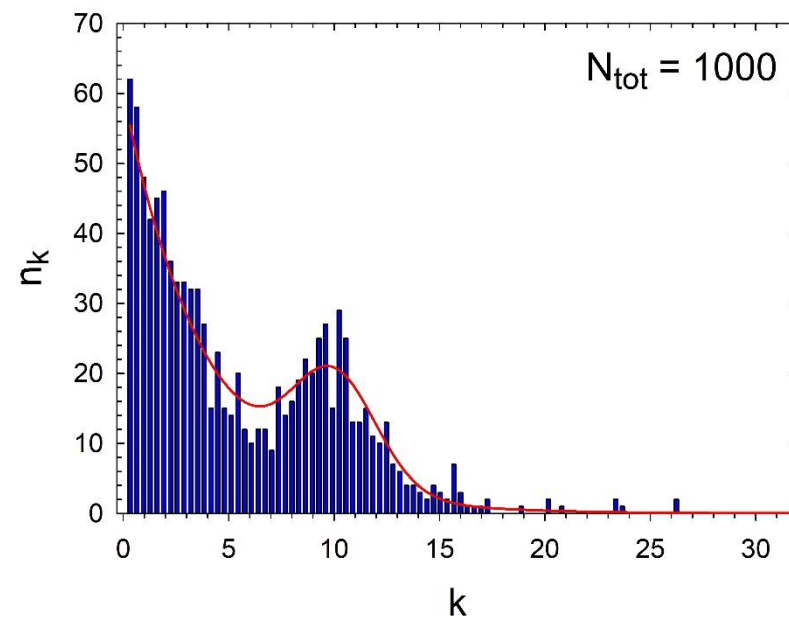
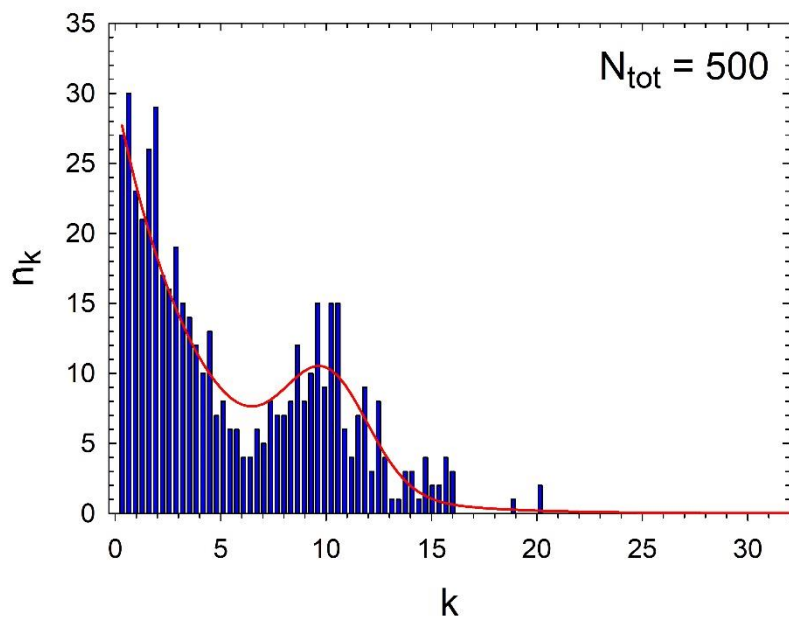
$$N_{sim} = 100$$

- Pro každou simulaci zjisti četnost n_{10} v 10-tém binu histogramu. Vypočítej průměrnou četnost μ_{10} a průměrnou odchylku σ_{10} .

$$\mu_{10} = \sum_{k=1}^{N_{sim}} \frac{1}{N_{sim}} n_{k,10} \qquad \sigma_{10}^2 = \sum_{k=1}^{N_{sim}} \frac{1}{N_{sim} - 1} (n_{k,10} - \mu_{10})^2$$

- Vynes závislost σ_{10} na počtu hodnot N_{tot} .

Chyba výšky binu histogramu



Chyba výšky binu histogramu

hist-bin-err-PH.py

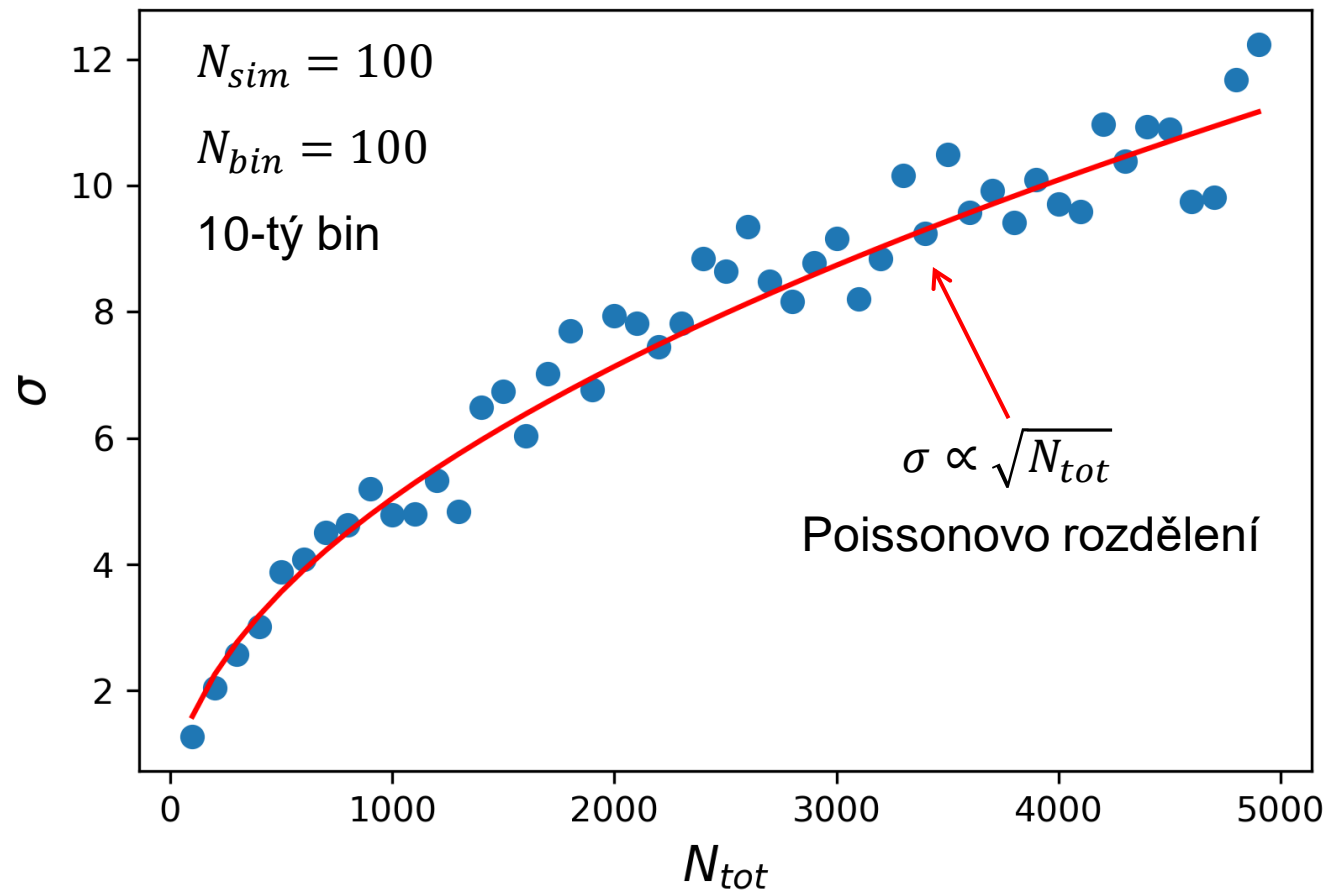
```
1 import numpy as np
2 import matplotlib.pyplot as plt
3
4 P=0.75
5 mu=10
6 sigma=2
7 tau=4
8
9 # simulace nahodne promenne s rozdelenim exp+gaussian 3:1
10 def sim():
11     r=np.random.random_sample()
12     if r>P:
13         return np.random.normal(mu,sigma)
14     else:
15         return np.random.exponential(tau)
16
17 nbins=100 #pocet binu
18 ibin=10 #divame se na pocty hodnot v 10-tem binu
19 Nopakovani=100 #pocet opakovani pro kazde Ntot
20 Ntot=np.arange(100,5000,100) #ruzne pocty Ntot
21 hodnota10=np.empty(Nopakovani) #pole hodnot v 10-tem binu
22 frakce10=np.empty(np.size(Ntot)) #frakce hodnot pripadajici na 10-ty bin
23 odchylka10=np.empty(np.size(Ntot)) #pole standardnich odchylek
24 ik=0
25 for k in Ntot: #cyklus pro postupne narustajici celkove pocty dat Ntot
26     print('počet dat = ',k)
27     for j in range(Nopakovani): #100 opakovani pro kazde Not
28         data=np.empty(k)
29         for i in range(k): #generovani histogramu
30             data[i]=sim()
31             hist,bin_edges=np.histogram(data,bins=nbins)
32             hodnota10[j]=hist[ibin]
33         frakce10[ik]=np.mean(hodnota10)/k #jaka frakce hodnot podne do 10-teho binu
34         odchylka10[ik]=np.std(hodnota10) #standardni odchylka pro pocty v 10-binu
35         ik+=1
36 frakce_mean=np.mean(frakce10) #prumerna frakce pripadajici na 10-ty bin
37
38 fig,ax=plt.subplots()
39 plt.scatter(Ntot,odchylka10)
40 plt.plot(Ntot,np.sqrt(Ntot*frakce_mean),c='red')
41 ax.set_xlabel('$N_{tot}$',fontsize=14)
42 ax.set_ylabel('$\sigma$',fontsize=14)
```

← simulace histogramu

← zaplnění histogramu

výpočet μ_{10} a σ_{10}

2. Zjistěte, jak závisí chyba výšky i -tého binu histogramu na celkovém počtu naměřených hodnot.



Náhodná procházka

3. Procházka opilého námořníka v 1D

- Před každým krokem si hodíme korunou:
padne orel \rightarrow krok vpřed
padne panna \rightarrow krok vzad
- Jaká bude střední vzdálenost l od počátku po N krocích?

$$E[k] = Np = \frac{1}{2}N \quad \text{padne } N/2\text{-krát panna a } N/2\text{-krát orel}$$

$$l = k - (N - k) = 2k - N \quad \Rightarrow \mu_l = E[l] = N - N = 0$$

- Jaká bude standardní odchylka σ_l ?

$$\sigma_l^2 = V[l] = E[l^2] - (E[l])^2 = E[(2k - N)^2] = 4E[k^2] - 4E[k]N + N^2$$

$$E[k^2] = V[k] + (E[k])^2 = Np - Np^2 + N^2p^2 = \frac{1}{4}(N + N^2)$$

$$\sigma_l^2 = N + N^2 - 2N^2 + N^2 = N \quad \sigma_l = \sqrt{N}$$

3. Procházka opilého námořníka ve 2D

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 N=10000
4 x=np.empty(N)
5 y=np.empty(N)
6 x[0]=y[0]=0
7 for i in range(1,N):
8     alpha=np.random.random_sample()*2*np.
9     x[i]=x[i-1]+np.cos(alpha)
10    y[i]=y[i-1]+np.sin(alpha)
11
12 plt.step(x,y)
```

```
1 import numpy as np
2 import matplotlib.pyplot as plt
3 N=10000
4 x=np.empty(N)
5 y=np.empty(N)
6 x[0]=y[0]=0
7 for i in range(1,N):
8     x[i]=x[i-1]+np.random.choice([-1,1],1)
9     y[i]=y[i-1]+np.random.choice([-1,1],1)
10
11 plt.step(x,y)
```

