

BIOINFORMATYKA

1. Wykład wstępny
2. Struktury danych w badaniach bioinformatycznych
3. Bazy danych: projektowanie i struktura
4. Bazy danych: projektowanie i struktura
5. Zastosowanie przykładowych programów do analizy danych
6. Zastosowanie przykładowych programów do analizy danych
7. Powiązania pomiędzy genami: równ. Hardyego-Weinberga, wsp. rekombinacji
8. Analiza sprzężeń - teoria
9. Analiza sprzężeń – przykłady programów
10. Analiza asocjacyjna
11. Analiza asocjacyjna - przykłady programów
12. Symulacje komputerowe, jackknife, bootstrap
- 13. Monte Carlo Markov Models**
14. Metody klasyfikacyjne
15. Wykład podsumowujący

- 1. Łańcuchy Markova**
- 2. Próbkowanie Gibbsa**
- 3. Przykład**

ŁAŃCUCHY MARKOVA

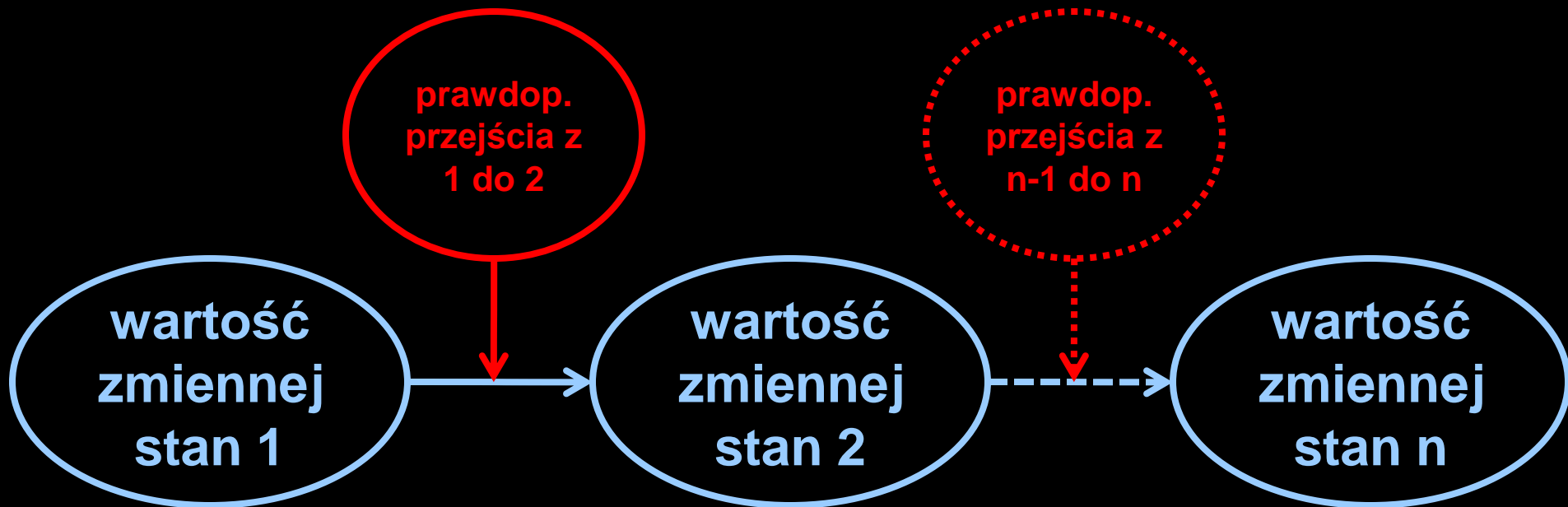


MARKOV CHAIN:

- modeluje prawdopodobieństwo uzyskania poszczególnych wartości zmiennej

ŁAŃCUCHY MARKOVA

- wartość zmiennej (X) np. liczba żywych rekombinantów, średnia wartość cechy
- stan (n) np. struktura populacji, liczebność osobników o danych genotypach, przeżywalność osobników, wsp. rekombinacji

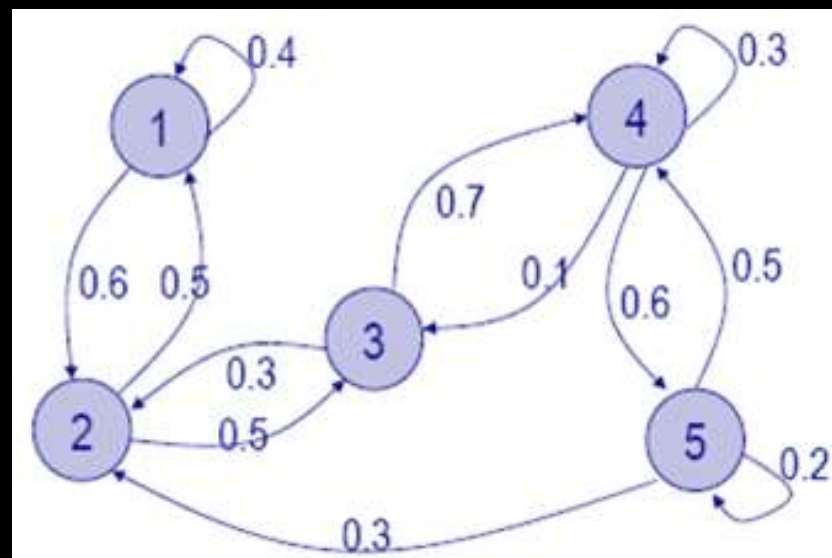


ŁAŃCUCHY MARKOVA

Macierz prawdopodobieństw przejścia pomiędzy poszczególnymi stanami

stan

	1	2	3	4	5
1	0.40	0.60	0.00	0.00	0.00
2	0.60	0.00	0.50	0.00	0.00
3	0.00	0.30	0.00	0.70	0.00
4	0.00	0.00	0.10	0.30	0.60
5	0.00	0.30	0.00	0.50	0.30



PRÓBKOWANIE GIBBSA

GIBBS SAMPLING

- algorytm oparty o metodę Markov Chain
- wykorzystują do generowania danych zasadę symulacji Monte Carlo
- Monte Carlo Markov Chain = MCMC
- umożliwia generowanie danych pochodzących z rozkładu wielowymiarowego $f(x,y,z) \rightarrow$ trudne
- Poprzez generowanie danych z rozkładów warunkowych poszczególnych zmiennych
 - generowanie x z rozkładu $f(x | y, z)$
 - generowanie y z rozkładu $f(y | x, z)$
 - generowanie z z rozkładu $f(z | x, y)$
 - \rightarrow łatwiejsze

PRÓBKOWANIE GIBBSA

PRZYKŁAD

1. dane:

- x - liczba rekombinantów, nieznana
- r - współczynnik rekombinacji, nieznana
- n - liczba osobników, znana

2. Estymacja: x i r

3. Prawdopodobieństwo warunkowe

- $P(r | x, n) = \text{Beta}(x, n-x)$

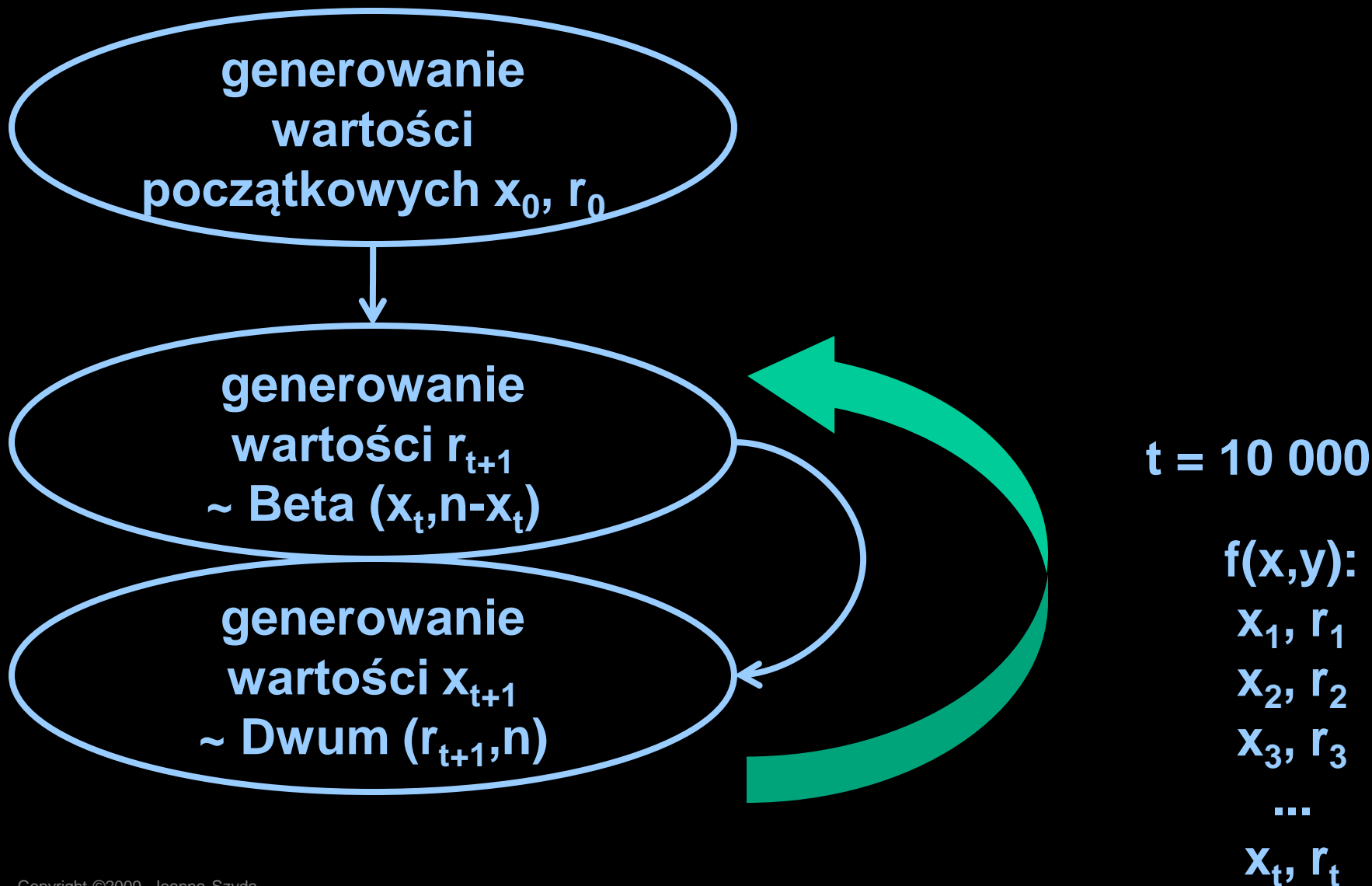
$$P(r | x, n-x) = \frac{\Gamma(x+n-x)}{\Gamma(x)\Gamma(n-x)} r^{x-1} (1-r)^{n-x-1}$$

- $P(x | r, n) = \text{Dwumianowy}(r, n)$

$$P(x | r, n) = \binom{n}{x} r^x (1-r)^{n-x} \quad \binom{n}{x} = \frac{n!}{x!(n-x)!}$$

PRÓBKOWANIE GIBBSA

GENEROWANIE DANYCH



PRÓBKOWANIE GIBBSA

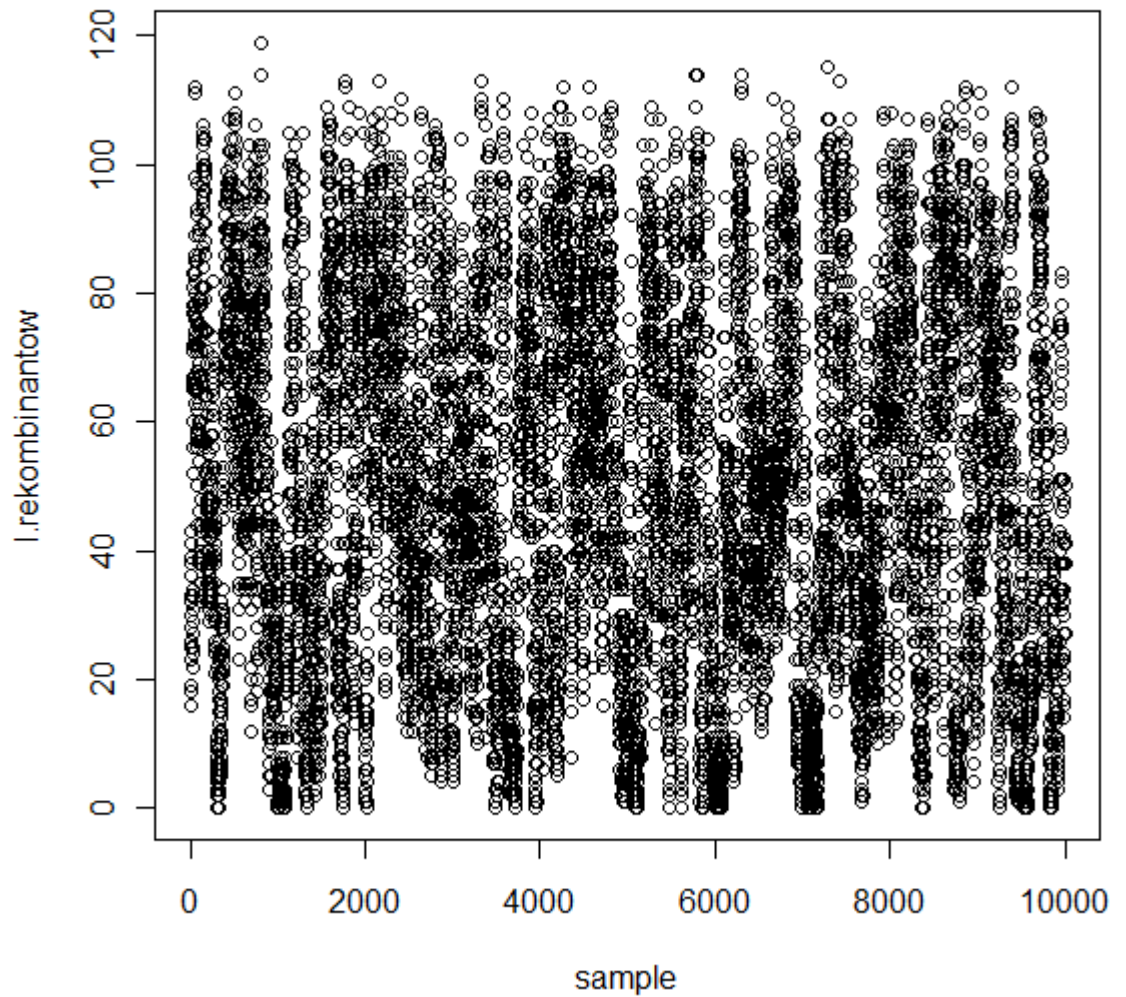
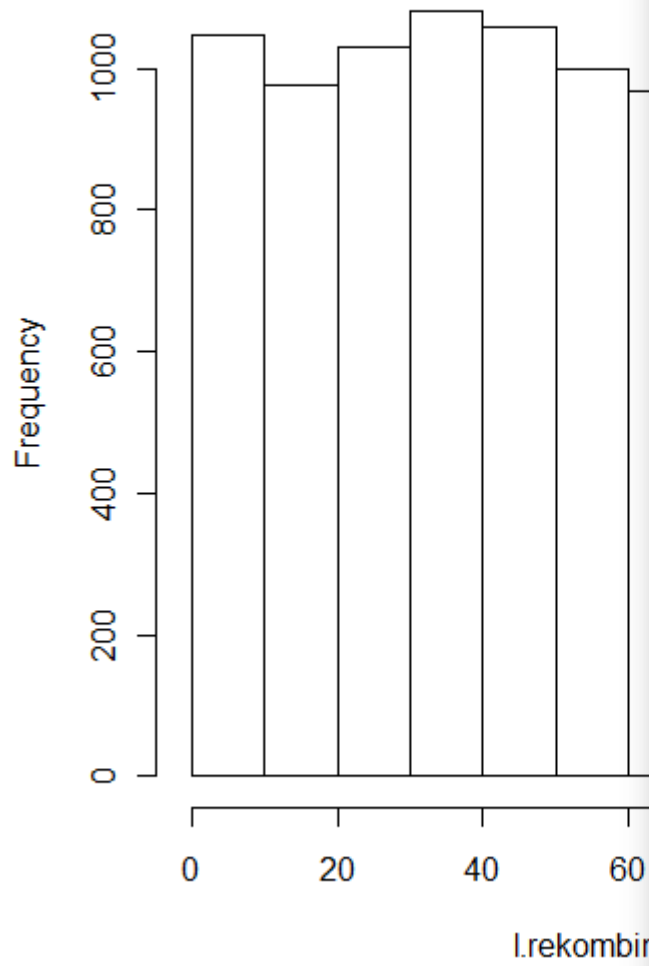
```
NS=10000          # number of generated samples
n=20              # total number of individuals
sample=matrix(0,nrow=NS,ncol=2)
# generate starting values
r=runif(1,0,0.5)  # wsp. rekombinacji
x=rbinom(1,n,r)   # liczba remombinantów
# generate next NS values
for (i in 1:NS) {
  r=rbeta(1,x+1,n-x+1)
  print(r)
  while (r>0.5) {
    r=rbeta(1,x+1,n-x+1)
  }
  x=rbinom(1,n,r)
  sample[i,1]=x
  sample[i,2]=r
}

# results
sample[1:20,1]
hist(sample[,1],main='histogram of generated values',xlab='l.rekombinantow')
plot(sample[,1],xlab='sample',ylab='l.rekombinantow')
meanx=mean(sample[,1])
sdx=sd(sample[,1])
meanx
sdx

sample[1:20,2]
hist(sample[,2],main='histogram of generated values',xlab='wsp.rekombinacji')
plot(sample[,2],xlab='sample',ylab='w.rekombinacji')
meanr=mean(sample[,2])
sdr=sd(sample[,2])
meanr
sdr
```

PRÓBKOWANIE GIBBSA

histogram of generated values



1. Łańcuchy Markova

2. Próbkowanie Gibbsa

3. Przykład