

Uczenie Maszynowe 2

Modele generatywne

dr hab. Piotr Duda, prof. AGH i PCz



Plan wykładu

Wykład 1

- **Organiczone Maszyny Boltzmann**

Wykład 2

- **Autoenkodery, VAE, WAE**

Wykład 3

- **GAN, Normalizing Flow**

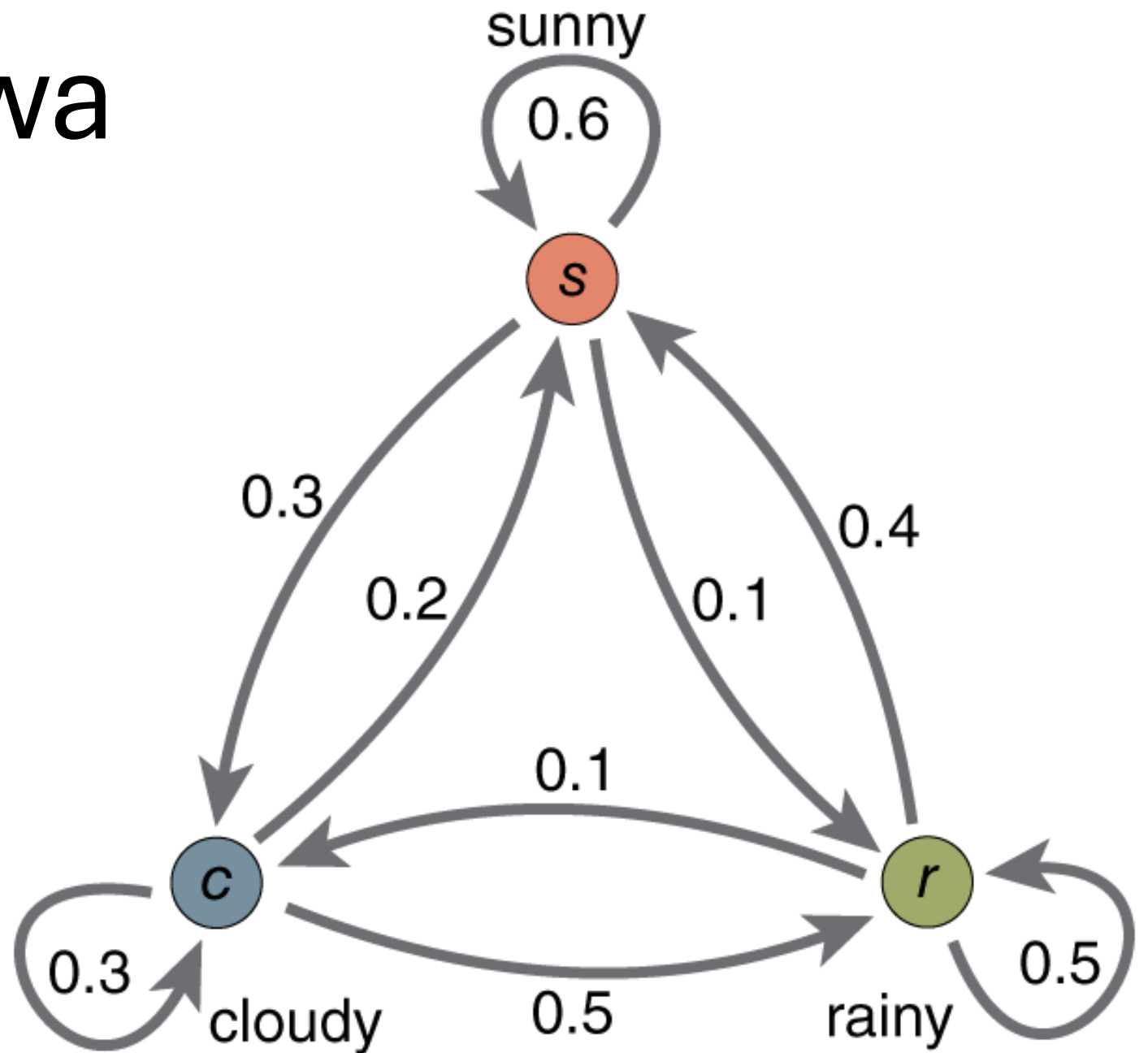
Wykład 4

- **Podejście dyfuzyjne, Przegląd „aktualnych” modeli**

Wykład 1 - Organiczne Maszyny Boltzmannna

- Procesy Markowa
- Metody Monte Carlo
- Próbkowanie Gibbsa
- Probabilistyczne modele nieukierunkowane
- Definicja RBM

Procesy Markowa



Proces Markowa

Proces Markowa (*ang. Markov Process*) – ciąg zdarzeń, w którym prawdopodobieństwo każdego zdarzenia zależy jedynie od wyniku poprzedniego. W ujęciu matematycznym, procesy Markowa to takie procesy stochastyczne, które spełniają własność Markowa.

Łańcuch Markowa

Definicja

Łańcuch Markowa jest ciągiem zmiennych losowych X_1, X_2, \dots .

Dziedzinę tych zmiennych nazywamy przestrzenią stanów, a realizacje X_n stanem w czasie n .

Jeśli rozkład warunkowy X_{n+1} jest funkcją wyłącznie zmiennej X_n

$$P(X_{n+1} \leq y | X_1, X_2, \dots, X_n) = P(X_{n+1} \leq y | X_n)$$

to mówimy, że proces stochastyczny posiada własność Markowa.

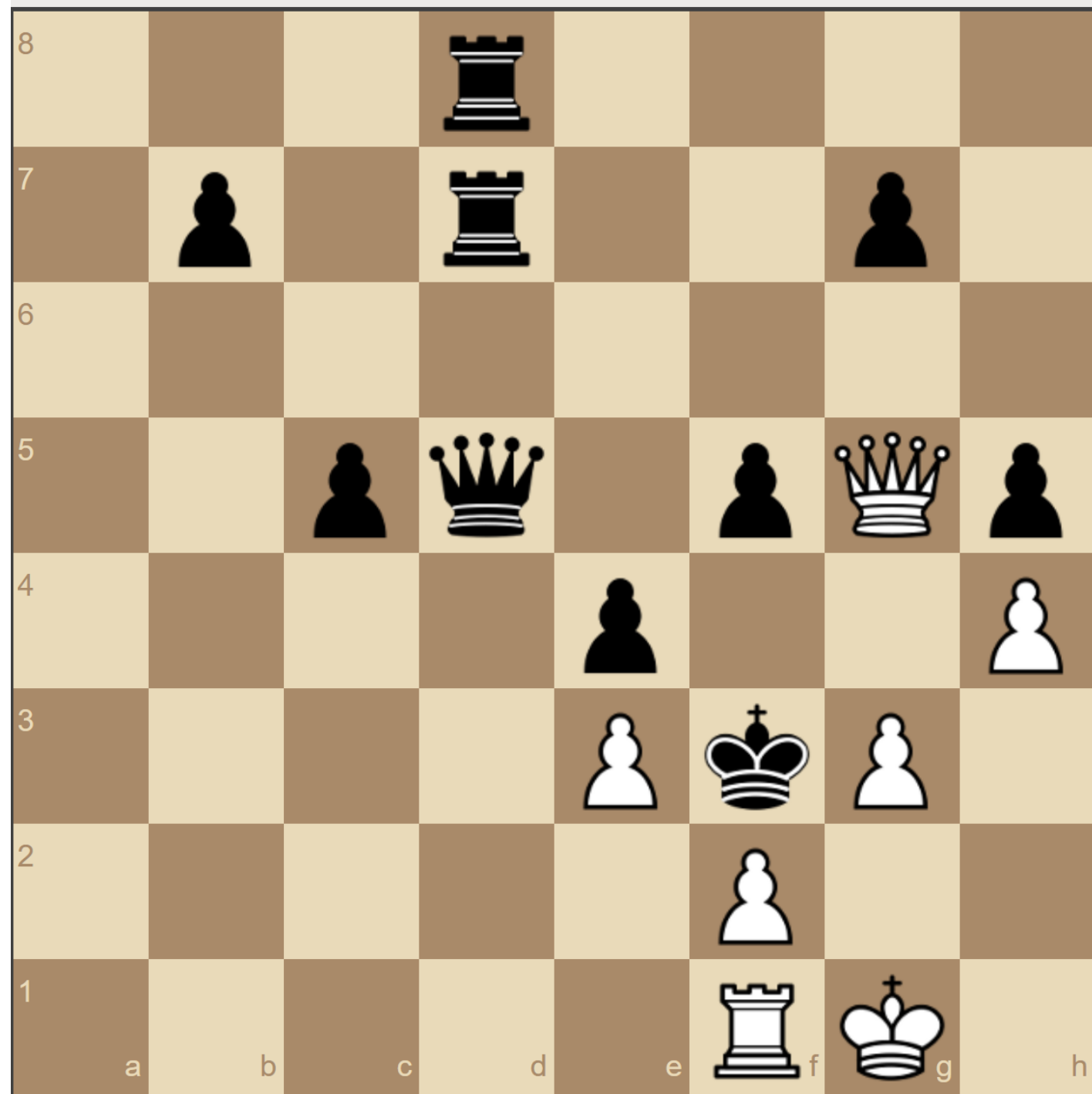
Przykład procesu Markowa

Przestrzeń stanów?

Ile jest możliwych stanów?

Ile jest możliwych stanów z tej pozycji?

Jaki powinien być kolejny ruch?



Przykład procesu, bez własności Markowa

Przestrzeń stanów to dostępne opcje
zakupu akcji?

Widzimy liczbę akcji do kupienia i ich
cenę.

Czy wiedza, że aktualny kurs akcji
PKOBP to 59,3800 wystarczy, żeby
podjąć decyzję o zakupie?



Procesy Markowa

Możliwe stany: $x \in \{1, \dots, N\}$, $X_0, X_1, X_2, \dots, X_n$ - stany przeszłe

$$P(X_{n+1} = x | X_1, X_2, \dots, X_n) = P(X_{n+1} = x | X_n)$$



Procesy Markowa

Możliwe stany: $x \in \{1, \dots, N\}$, $X_0, X_1, X_2, \dots, X_n$ - stany przeszłe

$$P(X_{n+1} = x | X_1, X_2, \dots, X_n) = P(X_{n+1} = x | X_n)$$

Macierz stochastyczna (w języku angielskim zwana: *transition matrix*, *stochastic matrix*, *probability matrix*, *substitution matrix*, oraz *Markov matrix*)

$$\mathbf{P}^{(n+1)} = \begin{bmatrix} p_{1,1}^{(n)} & \cdots & p_{1,n}^{(n)} \\ \vdots & \ddots & \vdots \\ p_{n,1}^{(n)} & \cdots & p_{n,n}^{(n)} \end{bmatrix}, \quad p_{i,j}^{(n)} = P(X_{n+1} = j | X_n = i)$$

Rozkład prawdopodobieństwa stanów (ang. *State Probability Distributions*)

$$\pi^{(0)} = \begin{bmatrix} P(X_0 = 1) \\ \vdots \\ P(X_0 = N) \end{bmatrix} \quad P(X_1 = j) = \sum_{i=1}^N P(X_1 = j | X_0 = i) P(X_0 = i) = \sum_{i=1}^N p_{i,j}^{(0)} P(X_0 = i)$$

$$\pi^{(1)} = \mathbf{P}^{(0)} \pi^{(0)}, \quad \pi^{(2)} = \mathbf{P}^{(0)} \mathbf{P}^{(1)} \pi^{(0)}, \quad \dots, \quad \pi^{(k+1)} = \prod_{t=0}^k \mathbf{P}^{(t)} \pi^{(0)}$$

Rozkład prawdopodobieństwa stanów π nazywamy stacjonarnym jeśli: $\pi = \mathbf{P}\pi$

Akademicki przykład procesu Markowa

Możliwe stany: $x \in \{\text{Słoneczny, Pochmurno, Deszczowo}\}$

$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix}$$

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.3	0.2
Pochmurno	0.1	0.6	0.3
Deszczowo	0.4	0.2	0.4

Akademicki przykład procesu Markowa

Możliwe stany: $x \in \{\text{Słoneczny: 1, Pochmurno: 2, Deszczowo: 3}\}$

$$P = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix}$$

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.3	0.2
Pochmurno	0.1	0.6	0.3
Deszczowo	0.4	0.2	0.4

Akademicki przykład procesu Markowa

Możliwe stany: $x \in \{\text{Słoneczny: 1, Pochmurno: 2, Deszczowo: 3}\}$

$$P = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix}$$

$$\pi^{(0)} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

$$\pi^{(1)} = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0,5 \\ 0,3 \\ 0,2 \end{bmatrix}$$

Proces jednorodny (homogeniczny)

$$\bigwedge_{t=0,1,\dots} P^{(t)} = P$$

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.3	0.2
Pochmurno	0.1	0.6	0.3
Deszczowo	0.4	0.2	0.4

Akademicki przykład procesu Markowa

Możliwe stany: $x \in \{\text{Słoneczny: 1, Pochmurno: 2, Deszczowo: 3}\}$

$$P = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix}$$

$$\pi^{(1)} = \begin{bmatrix} 0,5 \\ 0,3 \\ 0,2 \end{bmatrix}$$

$$\pi^{(2)} = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix} \begin{bmatrix} 0,5 \\ 0,3 \\ 0,2 \end{bmatrix} = \begin{bmatrix} 0,36 \\ 0,23 \\ 0,38 \end{bmatrix}$$

Proces jednorodny (homogeniczny)

$$\bigwedge_{t=0,1,\dots} P^{(t)} = P$$

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.3	0.2
Pochmurno	0.1	0.6	0.3
Deszczowo	0.4	0.2	0.4

Akademicki przykład procesu Markowa

Możliwe stany: $x \in \{\text{Słoneczny: 1, Pochmurno: 2, Deszczowo: 3}\}$

$$P = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix}$$

$$\pi^{(k)} = \begin{bmatrix} 0,31578947 \\ 0,38596491 \\ 0,29824561 \end{bmatrix}$$

$$\pi^{(k+1)} = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix} \begin{bmatrix} 0,31578947 \\ 0,38596491 \\ 0,29824561 \end{bmatrix} =$$

Proces jednorodny (homogeniczny)

$$\bigwedge_{t=0,1,\dots} P^{(t)} = P$$

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.3	0.2
Pochmurno	0.1	0.6	0.3
Deszczowo	0.4	0.2	0.4

Akademicki przykład procesu Markowa

Możliwe stany: $x \in \{\text{Słoneczny: 1, Pochmurno: 2, Deszczowo: 3}\}$

$$P = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix}$$

$$\pi^{(k)} = \begin{bmatrix} 0,31578947 \\ 0,38596491 \\ 0,29824561 \end{bmatrix}$$

$$\pi^{(k+1)} = \begin{bmatrix} 0,5 & 0,3 & 0,2 \\ 0,1 & 0,6 & 0,3 \\ 0,4 & 0,2 & 0,4 \end{bmatrix} \begin{bmatrix} 0,31578947 \\ 0,38596491 \\ 0,29824561 \end{bmatrix} = \begin{bmatrix} 0,31578947 \\ 0,38596491 \\ 0,29824561 \end{bmatrix}$$

Proces jednorodny (homogeniczny)

$$\bigwedge_{t=0,1,\dots} P^{(t)} = P$$

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.3	0.2
Pochmurno	0.1	0.6	0.3
Deszczowo	0.4	0.2	0.4

Procesy Markowa

Stan osiągalny (*ang. accessible*):

Mówimy, że stan ' j ' jest osiągalny ze stanu ' i ' jeżeli $p_{i,j}^{(n)} > 0$, dla $n \geq 0$

Stany skomunikowane (*ang. communicate*):

Mówimy, że stany ' j ' i ' i ' są skomunikowane jeżeli są wzajemnie osiągalne

Mówimy, że proces jest nieredukowalny (*ang. irreducible*):

$$\bigwedge_{i,j} \bigvee_{k \geq 0} P(X_k = j | X_0 = i) > 0$$

Procesy Markowa

Mówimy, że proces jest nieredukowalny (*ang. irreducible*):

$$\bigwedge_{i,j} \bigvee_{k>0} P(X_k = j | X_0 = i) > 0$$

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.3	0.2
Pochmurno	0.1	0.6	0.3
Deszczowo	0.4	0.2	0.4

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.5	0
Pochmurno	0.5	0.1	0.4
Deszczowo	0	0.4	0.6

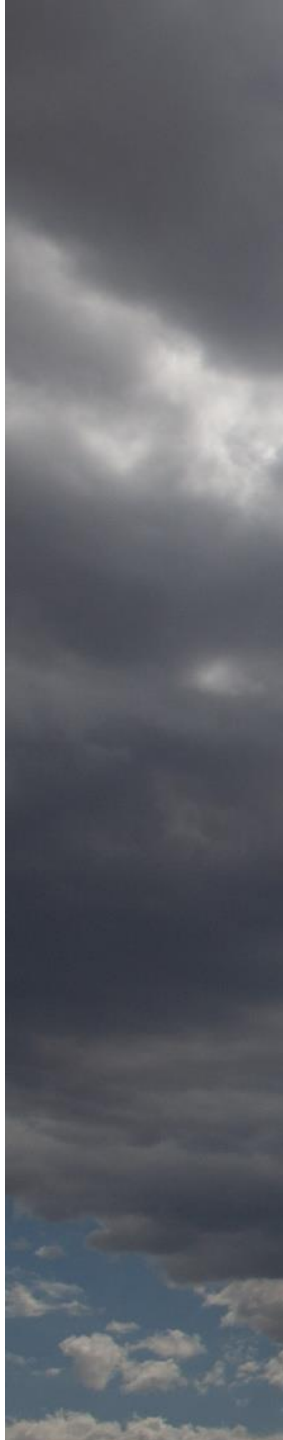
Procesy Markowa

Mówimy, że proces jest nieredukowalny (*ang. irreducible*):

$$\bigwedge_{i,j} \bigvee_{k>0} P(X_k = j | X_0 = i) > 0$$

Kontrprzykład :

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0	1	0
Pochmurno	0	0.5	0.5
Deszczowo	0	0	1



Procesy Markowa

Rozważmy możliwość przejścia procesu ze stanu ' i ' w siebie.

$$P(X_k = i | X_0 = i) > 0 \rightarrow k_i = k$$

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0	0.5	0.5
Pochmurno	0.5	0.1	0.4
Deszczowo	0.4	0.4	0.2

Procesy Markowa

Rozważmy możliwości przejścia procesu ze stanu ' i ' w siebie.

$$\{k : P(X_k = i | X_0 = i) > 0\}$$

Wówczas proces nazywamy aperiodycznym (*ang. aperiodic*) jeżeli:

$$\bigwedge_i NWD(\{k : P(X_k = i | X_0 = i) > 0\}) = 1$$



Procesy Markowa

Rozważmy możliwości przejścia procesu ze stanu ' i ' w siebie.

$$\{k : P(X_k = i | X_0 = i) > 0\}$$

Wówczas proces nazywamy aperiodycznym (*ang. aperiodic*) jeżeli:

$$\bigwedge_i NWD(\{k : P(X_k = i | X_0 = i) > 0\}) = 1$$

Kontrprzykład:

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0	1	0
Pochmurno	0	0	1
Deszczowo	1	0	0

Procesy Markowa

Rozważmy możliwości przejścia procesu ze stanu ' i ' w siebie.

$$\{k : P(X_k = i | X_0 = i) > 0\}$$

Wówczas proces nazywamy aperiodycznym (*ang. aperiodic*) jeżeli:

$$\bigwedge_i NWD(\{k : P(X_k = i | X_0 = i) > 0\}) = 1$$

Przykład:

Stan obecny \ Następny stan	Słonecznie	Pochmurno	Deszczowo
Słonecznie	0.5	0.5	0
Pochmurno	0.3	0.2	0.5
Deszczowo	0.2	0.3	0.5

Procesy Markowa

Jeżeli łańcuch jest **nieredukowalny** i **aperiodyczny** to:

1. Istnieje stacjonarny rozkład prawdopodobieństwa

$$\pi = [\pi_1, \dots, \pi_N]$$

2. Dla każdego stanu ' i ':

$$\pi_i = \lim_{n \rightarrow \infty} P(X_n = i | X_0 = j)$$



Metody Monte Carlo



Monte Carlo

Metoda Monte Carlo (MC) – metoda stosowana do modelowania matematycznego procesów zbyt złożonych (obliczania całek, łańcuchów procesów statystycznych), aby można było przewidzieć ich wyniki za pomocą podejścia analitycznego.

Monte Carlo z łańcuchem Markowa (MCMC)

Próbkowanie Monte Carlo łańcuchami Markowa (ang. Markov Chain Monte Carlo, MCMC) – klasa algorytmów próbkowania z rozkładu prawdopodobieństwa. Poprzez budowę łańcucha Markowa o rozkładzie równowagowym pasującym do pożądanego rozkładu można wydajnie próbować złożone rozkłady prawdopodobieństwa. Im większa liczba kroków w takim algorytmie, tym dokładniej rozkład próbki odpowiada pożądanemu rozkładowi.

Monte Carlo z łańcuchem Markowa (MCMC)

Różne warianty MCMC

- Metropolis-Hastings
- Hamiltonian Monte Carlo
- Slice Sampling
- Reversible Jump MCMC
- Gibbs Sampling

Algorytm Metropolis-Hastings

Procedura **Metropolis-Hastings** jest jednym z podstawowych algorytmów **MCMC** używanych do generowania próbek z rozkładu docelowego $P(X)$, z którego bezpośrednio próbkowanie może być trudne lub niemożliwe.

Metropolis-Hastings to ogólna metoda, która wykorzystuje próbki z prostszego rozkładu, tzw. **rozkładu propozycji**, aby przybliżyć próbkowanie z docelowego rozkładu.

Rozkład propozycji będziemy traktowali jako rozkład założony przez użytkownika.

Algorytm Metropolis-Hastings

1. Wybierz początkową wartość $x^{(0)}$ dla łańcucha.
Może to być dowolna wartość w przestrzeni próbkowania lub próbka z łatwo dostępnego rozkładu.
2. Ustal rozkład propozycji $Q(x'|x)$, który będzie służył do generowania kolejnych stanów
Np. rozkład normalny centrowany wokół aktualnego stanu x , $N(x, 1)$.
3. Powtarzaj poniższe kroki przez określoną liczbę iteracji (lub do osiągnięcia zbieżności):
 - a. Dla bieżącego stanu $x^{(t)}$, wygeneruj nową wartość x' z rozkładu propozycji $Q(x'|x^{(t)})$.
#Jest to potencjalny nowy stan, który może zostać zaakceptowany lub odrzucony.
 - b. Oblicz współczynnik akceptacji:

$$\alpha = \min \left(1, \frac{P(x')Q(x^{(t)}|x')}{P(x^{(t)})Q(x'|x^{(t)})} \right)$$

- c. Wygeneruj losową liczbę z rozkładu jednostajnego $(0,1)$, $u \sim \text{Uniform}(0,1)$
- d. Jeżeli $u \leq \alpha$ to $x^{(t+1)} = x'$
w przeciwnym razie $x^{(t+1)} = x^{(t)}$

Przykład: Modelowanie ryzyka kredytowego

Założmy, że mamy informacje o klientach banku biorących kredyt. Chcemy oszacować jaką „stratę” ryzykujemy, gdyby nasz bank zwiększył pulę kredytów powiedzmy dziesięciokrotnie.



Modelowanie ryzyka kredytowego

Cel

„Naszym celem jest oszacowanie rozkładu strat”.

Chcemy na podstawie dotychczasowych doświadczeń wygenerować dane o rozkładzie takim samym, jak dane które mamy.

Nie chcemy określić rozkład prawdopodobieństwa explicite, a tylko wygenerować dane tak jakbyśmy ten rozkład znali.

Nie chcemy powielać tych samych danych.



Modelowanie ryzyka kredytowego

Założenia modelu

1. Mamy portfel kredytowy składający się z N klientów.
2. Każdy klient jest odpowiedzialny za stratę L_i w przypadku niewypłacalności. Wartości te są znane.
3. Każdy klient ma przypisane prawdopodobieństwo niewypłacalności p_i , które jest nieznane i jest ukrytym parametrem modelu.
4. Chcemy ustalić p_i tak, żeby pochodziły z takiego samego rozkładu jak oryginalne dane, które mamy.



Modelowanie ryzyka kredytowego

Bank chce udzielić 1000 kredytów.

Ma na to przeznaczone 108 010 000\$.

```
#Wartości kredytów ...  
print(L)
```

```
[ 40000 140000  90000  40000  30000 110000 100000  90000 160000 170000  
 30000  40000 200000  30000 120000 180000  20000  80000  50000  60000  
170000  20000 150000  70000  80000 190000 190000 190000 160000  20000  
200000  70000  90000 120000 200000  60000  40000 190000 130000 130000  
 60000 180000 120000 130000  20000 160000 190000 110000 150000 140000  
130000 190000  20000 170000  80000  70000 130000  90000  40000 130000  
 20000 170000  60000  40000 200000 120000 120000 190000 190000 200000  
200000 100000 140000 100000 160000 120000 170000  60000 150000 200000  
120000  50000  70000  10000  80000 140000 170000 170000  40000 190000  
100000  70000  10000 110000 130000 110000 200000  10000 190000 170000  
 30000  60000  20000  30000  70000  40000 160000 160000  40000  90000]
```

Modelowanie ryzyka kredytowego

Bank chce udzielić 1000 kredytów.

Ma na to przeznaczone 108 010 000\$.

Chcemy do każdego kredytu przypisać prawdopodobieństwo poniesienia straty.

```
#Prawdopodobieństwa poniesienia straty  
print(p)
```

```
[3.26665832e-01 4.18899882e-02 8.48811175e-01 7.03079931e-01  
8.59633999e-01 5.03082006e-01 1.68172052e-01 9.08297414e-01  
7.79027342e-01 7.84637080e-01 1.12038868e-01 5.35405829e-01  
7.75493401e-01 9.65969424e-01 7.95958261e-01 2.33503330e-01  
3.70878090e-01 5.80969642e-01 8.49412625e-01 8.34049057e-01  
7.09980359e-01 6.94076618e-01 8.22918682e-01 8.79759435e-01  
1.43823192e-02 3.97149580e-01 6.10251988e-01 1.53773157e-01  
5.54172181e-01 7.64732635e-01 9.81117129e-01 8.36674562e-01  
1.42640693e-01 7.45881403e-01 7.11832235e-02 6.78432294e-01  
6.38592797e-01 7.82834898e-01 3.61729039e-01 8.09816404e-01  
4.28769628e-01 9.44147436e-01 8.90092798e-01 8.29648803e-01  
2.12974841e-01 2.57242194e-01 9.63907423e-01 7.11649865e-01  
4.78508581e-01 8.80286459e-01 6.37751342e-01 6.40509734e-01]
```

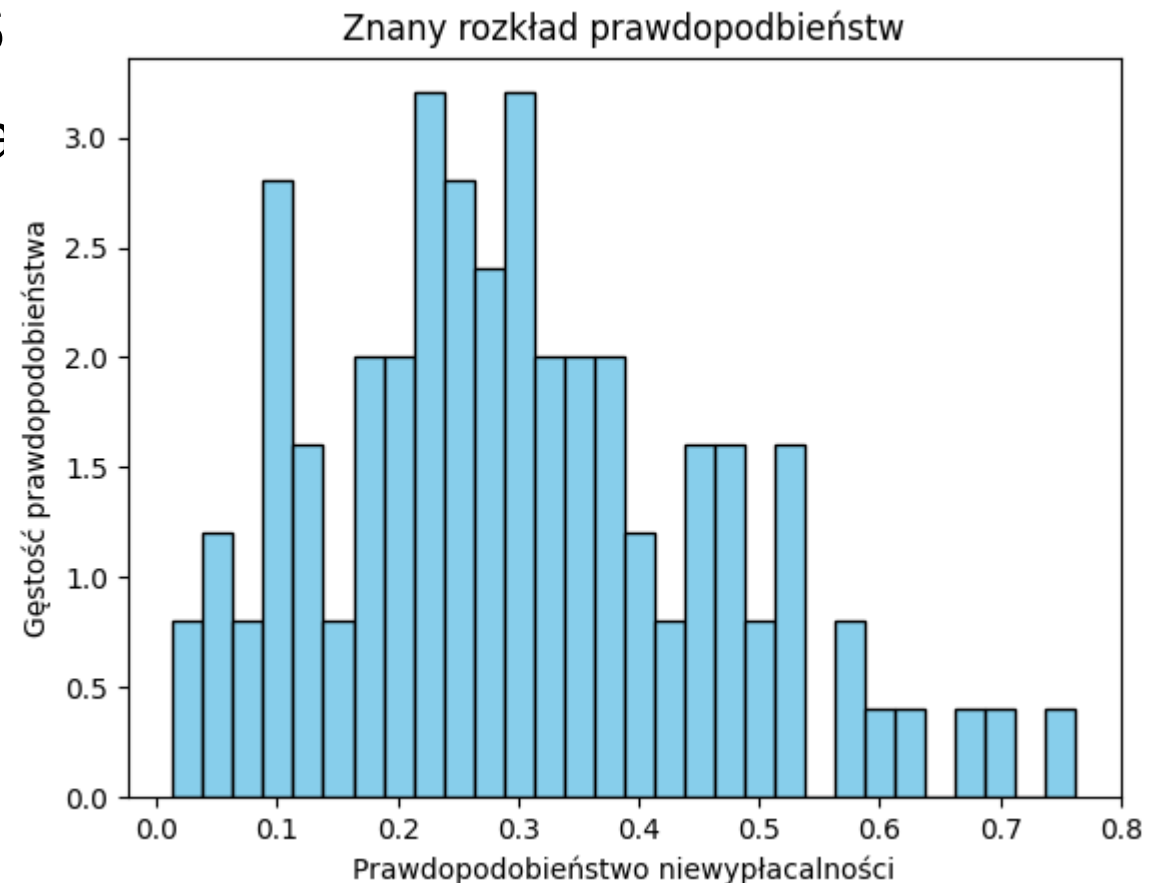
Modelowanie ryzyka kredytowego

Bank chce udzielić 1000 kredytów.

Ma na to przeznaczone 108 010 000\$

Chcemy do każdego kredytu przypisać poniesienia straty.

Mamy takie prawdopodobieństwa ocenione dla 100 kredytów.



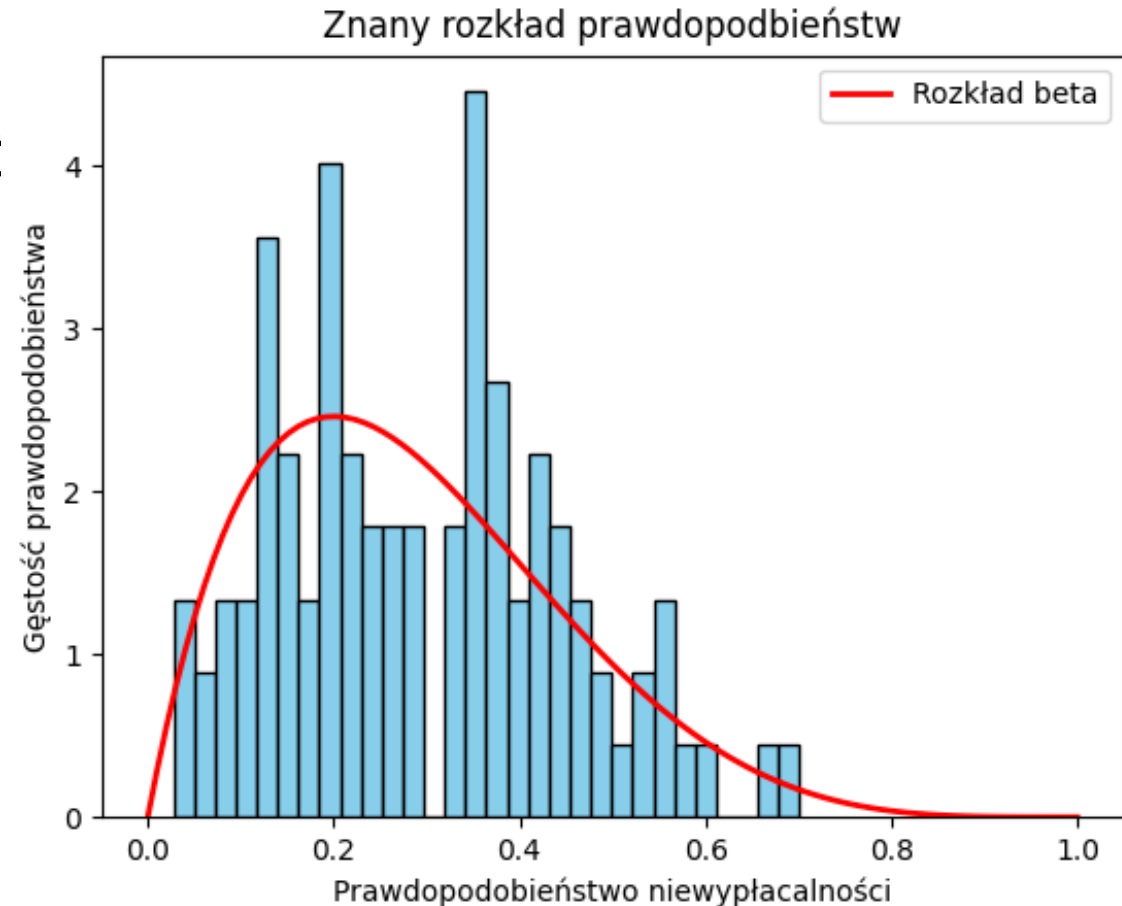
Modelowanie ryzyka kredytowego

Bank chce udzielić 1000 kredytów.

Ma na to przeznaczone 108 010 000\$.

Chcemy do każdego kredytu przypisać poniesienia straty.

Mamy takie prawdopodobieństwa ocenione dla 100 kredytów.



Modelowanie ryzyka kredytowego

Bank chce udzielić 1000 kredytów.

Ma na to przeznaczone 108 010 000\$.

Chcemy do każdego kredytu przypisać prawdopodobieństwo poniesienia straty.

Mamy takie prawdopodobieństwa
ocenione dla 100 kredytów.

Modelowanie ryzyka kredytowego

1. Zainicjalizuj wartości początkowe

2. Iteracyjnie generuj nowe wartości p_i' z rozkładu propozycji, np. normalnego $N(p_i, \sigma)$.

2.1 Oblicz współczynnik akceptacji

$$\alpha = \min \left(1, \frac{P(x')Q(x^{(t)}|x')}{P(x^{(t)})Q(x'|x^{(t)})} \right)$$

2.2 Zdecyduj czy podmienić p_i'

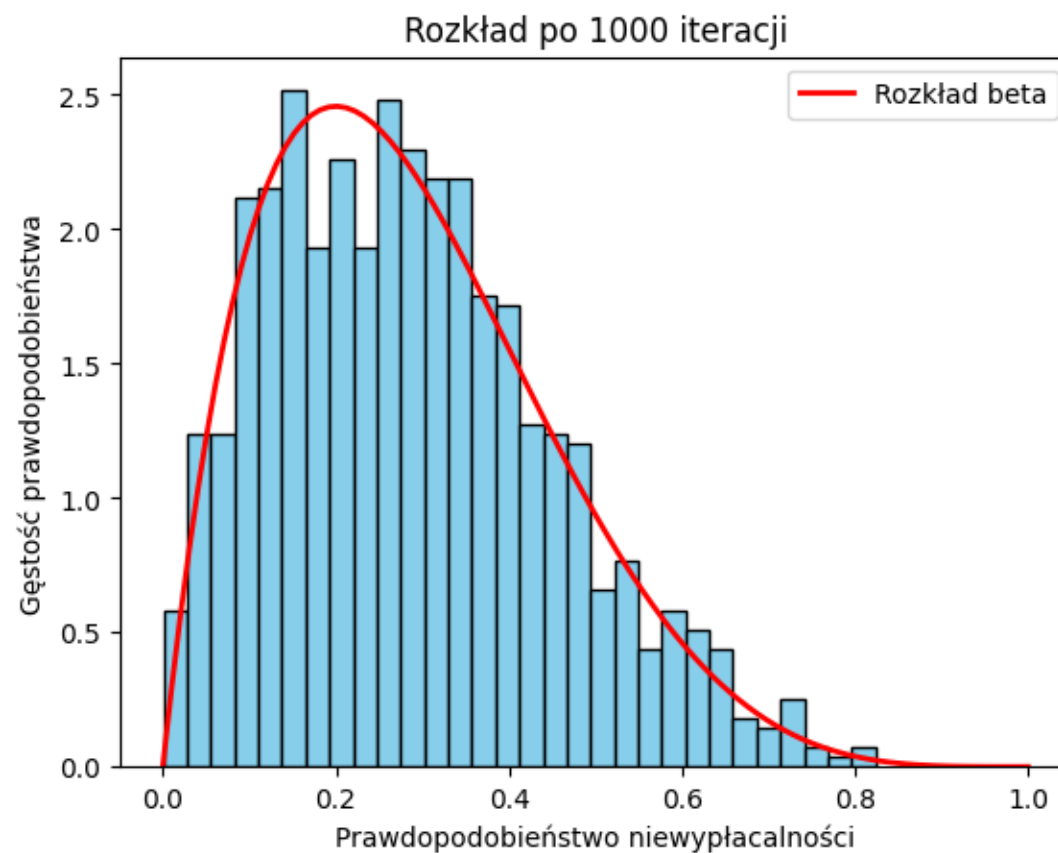
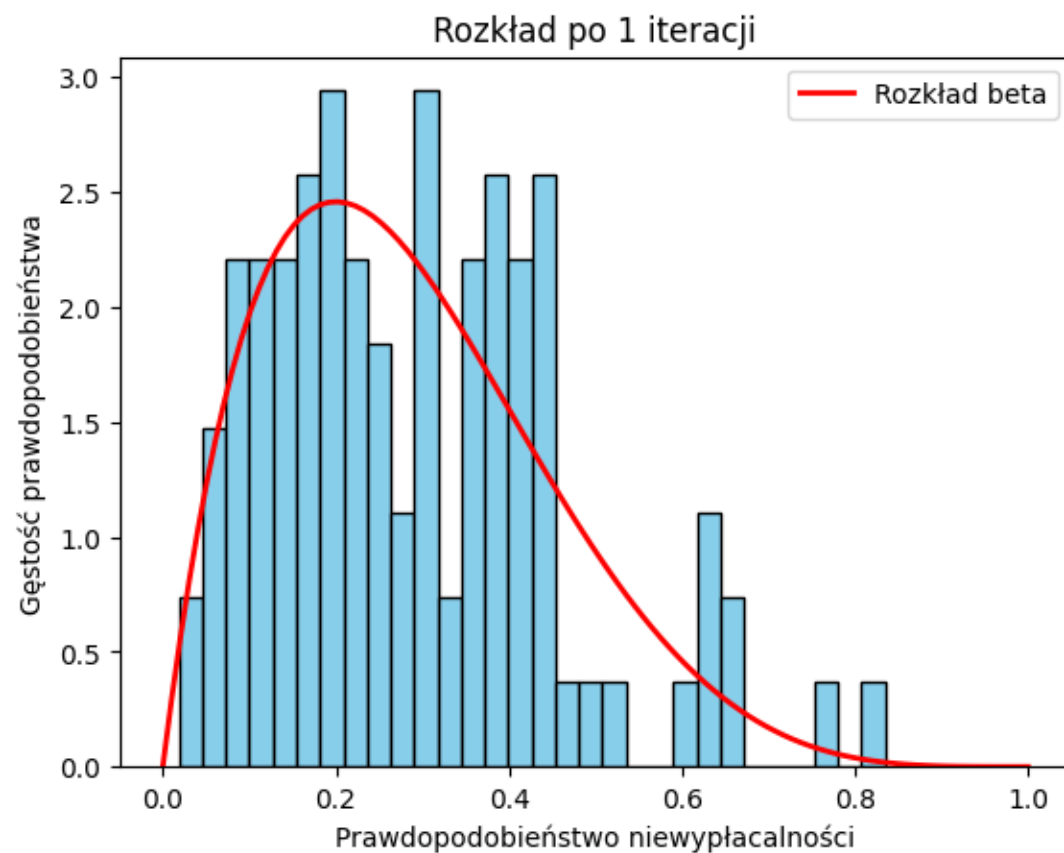
3. Po wielu iteracjach weź ostatnie N wygenerowanych p_i

4. Symulacja strat: Oblicz całkowitą stratę portfela jako

$$S = \sum_{i=1}^N L_i I_i,$$

gdzie I_i jest zmienną losową wskazującą niewypłacalność ($I_i = 1$ z prawdopodobieństwem p_i).

Rozkład wartości na początku i po wielu krokach



Burn-in

W metodzie MCMC, próbki są generowane sekwencyjnie, ale początkowe próbki mogą nie pochodzić jeszcze z rozkładu docelowego, ponieważ łańcuch potrzebuje czasu na osiągnięcie *stanu stacjonarnego*, czyli sytuacji, gdy rozkład generowanych próbek stabilizuje się i dobrze przybliża rozkład docelowy.

Jak dobrać długość burn-in?

Długość burn-in zależy od:

- **Złożoności modelu i rozkładu:** Im bardziej złożony rozkład, tym dłużej może trwać osiągnięcie stanu stacjonarnego.
- **Szybkości konwergencji algorytmu:** W niektórych modelach stan stacjonarny osiąga się szybciej, w innych wolniej.
- **Heurystyk i metod diagnostycznych:** Można oszacować długość burn-in za pomocą różnych metod diagnostycznych, które oceniają, czy próbki osiągnęły stabilny stan.

Praktycznie, długość burn-in może być ustalana eksperymentalnie, np. odrzucając od 10% do 50% początkowych próbek, lub za pomocą bardziej zaawansowanych testów diagnostycznych konwergencji.

Monte Carlo z łańcuchem Markowa (MCMC)

Różne warianty MCMC

- Metropolis-Hastings
- Hamiltonian Monte Carlo
- Slice Sampling
- Reversible Jump MCMC
- Gibbs Sampling

Próbkowanie Gibbsa (*ang. Gibbs sampling*)

- Gibbs sampling to metoda do generowania próbek z trudnych do bezpośredniego próbkowania wielowymiarowych rozkładów prawdopodobieństwa. Jest szczególnym przypadkiem algorytmu Metropolis-Hastings i należy do szerszej klasy metod Monte Carlo z łańcuchem Markowa (MCMC).

Zasada działania

Założmy, że mamy rozkład prawdopodobieństwa $P(X_1, X_2, \dots, X_n)$, który jest rozkładem z którego chcemy wygenerować próbki. Bezpośrednie próbkowanie z tego rozkładu może być trudne, ale możliwe jest wyznaczenie warunkowych rozkładów dla każdej zmiennej X_i przy danym zestawie wartości pozostałych zmiennych.

Główny pomysł Gibbs sampling polega na:

1. **Inicjalizacji:** Losowym wyborze początkowych wartości zmiennych X_1, X_2, \dots, X_n .
2. **Aktualizacji zmiennych po kolei:** Każda zmienna jest próbkowana z rozkładu warunkowego

$$P(X_i | X_1, \dots, X_{i-1}, X_{i+1}, \dots, X_n),$$

gdzie wartości pozostałych zmiennych traktowane są jako ustalone.

3. **Iteracji:** Proces jest powtarzany iteracyjnie dla kolejnych zmiennych, aż do uzyskania stabilnych wartości (tzn. aż łańcuch Markowa osiągnie stan stacjonarny).

Przykład

Założmy, że znamy warunkowe rozkłady prawdopodobieństwa

$$X|Y = y \sim N\left(\mu_X + \rho \frac{\sigma_X}{\sigma_Y} (y - \mu_Y), (1 - \rho^2) \sigma_X^2\right)$$

$$Y|X = x \sim N\left(\mu_Y + \rho \frac{\sigma_Y}{\sigma_X} (x - \mu_X), (1 - \rho^2) \sigma_Y^2\right)$$

gdzie

X i Y to jednowymiarowe zmienne losowe,

μ_X i μ_Y to wartości oczekiwane zmiennych losowych X i Y ,

σ_X^2 i σ_Y^2 to wariancje zmiennych losowych X i Y ,

ρ to współczynnik związany z korelacją.

Przykład

```
# Parametry rozkładu dwuwymiarowego normalnego
mu_X, mu_Y = 5, -10          # Średnie
sigma_X, sigma_Y = 1, 5      # Odchylenia standardowe
rho = 0.8                    # Korelacja

# Liczba iteracji
num_samples = 10000

#Przechowywanie próbek
samples = np.zeros((num_samples, 2))

#Inicjalizacja pierwszej wartości
samples[0] = [0, 0]
```

Przykład

$$X|Y=y \sim N\left(\mu_X + \rho \frac{\sigma_X}{\sigma_Y}(y - \mu_Y), (1 - \rho^2)\sigma_X^2\right)$$
$$Y|X=x \sim N\left(\mu_Y + \rho \frac{\sigma_Y}{\sigma_X}(x - \mu_X), (1 - \rho^2)\sigma_Y^2\right)$$

```
# Parametry rozkładu dwuwymiarowego normalnego
mu_X, mu_Y = 5, -10          # Średnie
sigma_X, sigma_Y = 1, 5      # Odchylenia standardowe
rho = 0.8                    # Korelacja

# Liczba iteracji
num_samples = 10000

#Przechowywanie próbek
samples = np.zeros((num_samples, 2))

#Inicjalizacja pierwszej wartości
samples[0] = [0, 0]
```

```
# Funkcje generujące próbki z rozkładów warunkowych
def sample_X_given_Y(y):
    mean = mu_X + rho * (sigma_X / sigma_Y) * (y - mu_Y)
    variance = (1 - rho ** 2) * sigma_X ** 2
    return np.random.normal(mean, np.sqrt(variance))

def sample_Y_given_X(x):
    mean = mu_Y + rho * (sigma_Y / sigma_X) * (x - mu_X)
    variance = (1 - rho ** 2) * sigma_Y ** 2
    return np.random.normal(mean, np.sqrt(variance))
```

Przykład

```
# Parametry rozkładu dwuwymiarowego normalnego
mu_X, mu_Y = 5, -10          # Średnie
sigma_X, sigma_Y = 1, 5      # Odchylenia standardowe
rho = 0.8                    # Korelacja

# Liczba iteracji
num_samples = 10000

#Przechowywanie próbek
samples = np.zeros((num_samples, 2))

#Inicjalizacja pierwszej wartości
samples[0] = [0, 0]
```

```
# Funkcje generujące próbki z rozkładów warunkowych
def sample_X_given_Y(y):
    mean = mu_X + rho * (sigma_X / sigma_Y) * (y - mu_Y)
    variance = (1 - rho ** 2) * sigma_X ** 2
    return np.random.normal(mean, np.sqrt(variance))

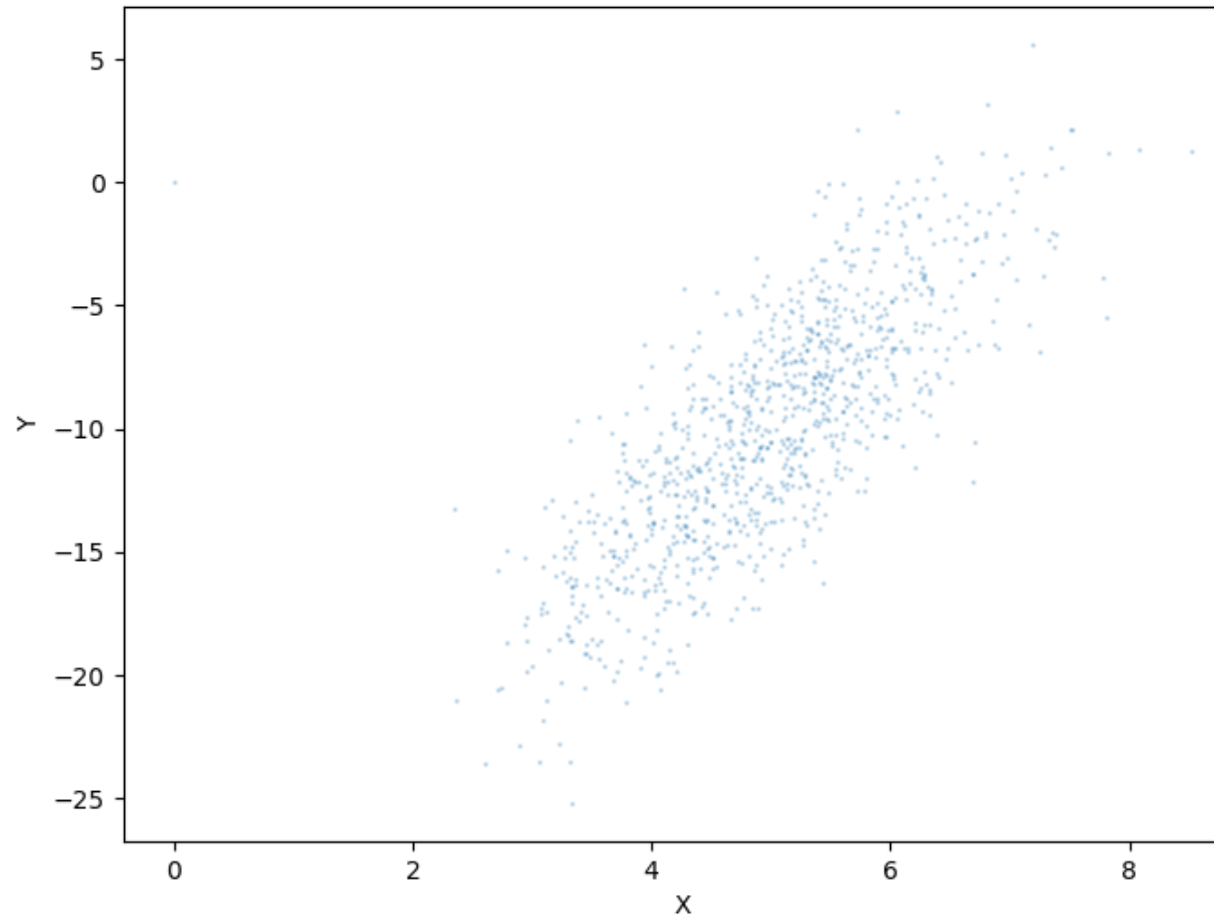
def sample_Y_given_X(x):
    mean = mu_Y + rho * (sigma_Y / sigma_X) * (x - mu_X)
    variance = (1 - rho ** 2) * sigma_Y ** 2
    return np.random.normal(mean, np.sqrt(variance))
```

```
# Wykonaj Gibbs Sampling
for i in range(1, num_samples):
    x_prev, y_prev = samples[i - 1]
    x_new = sample_X_given_Y(y_prev)
    y_new = sample_Y_given_X(x_new)
    samples[i] = [x_new, y_new]
```

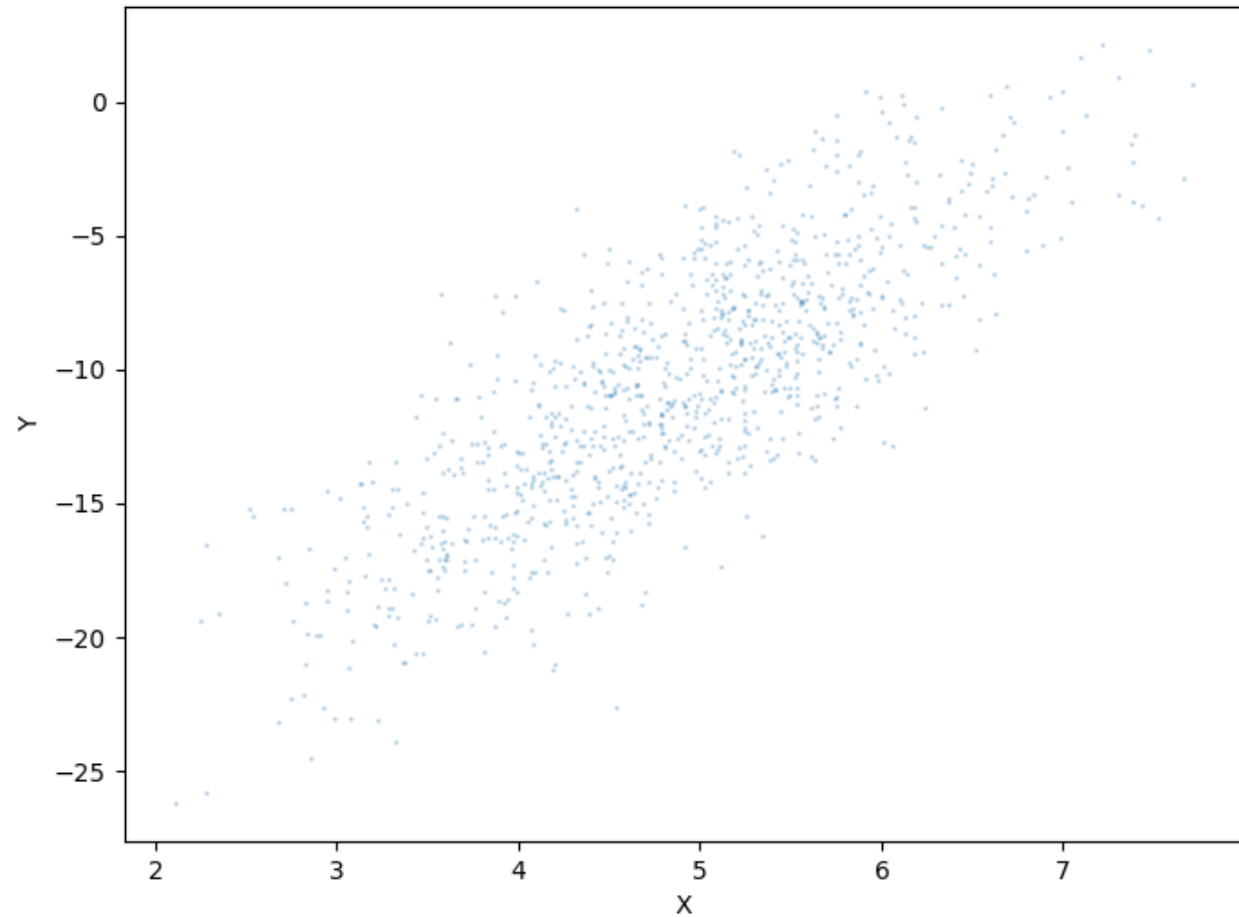
Przykład

```
# Parametry rozkładu dwuwymiarowego normalnego  
mu_X, mu_Y = 5, -10      # Średnie  
sigma_X, sigma_Y = 1, 5  # Odchylenia standardowe  
rho = 0.8                # Korelacja
```

Próbki wygenerowane metodą Gibbs Sampling



Próbki wygenerowane metodą Gibbs Sampling



Gibbs sampling

Można wykazać, że **Gibbs sampling** jest procesem **nieredukowalnym** i **aperiodycznym**, czyli dąży do **rozkładu stacjonarnego**.

Zalety i wady

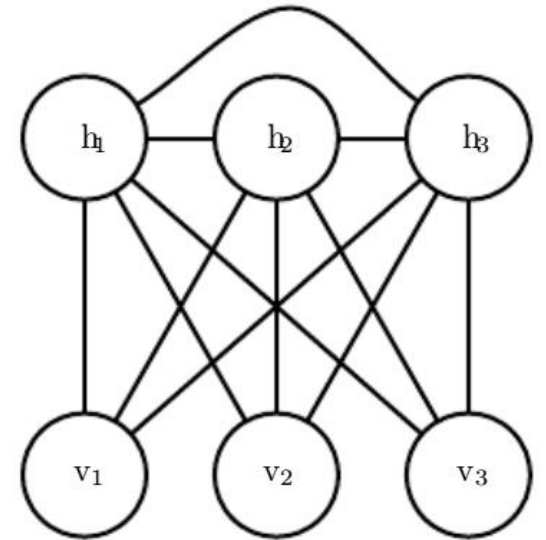
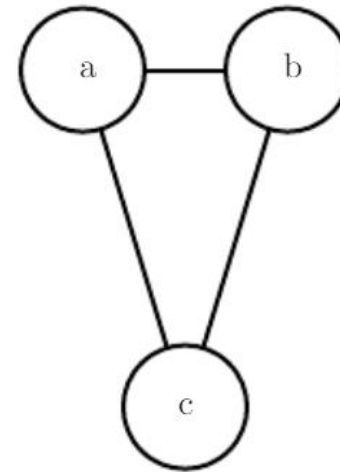
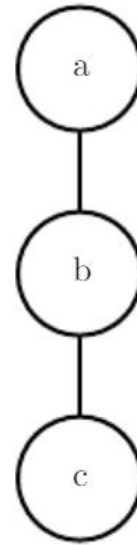
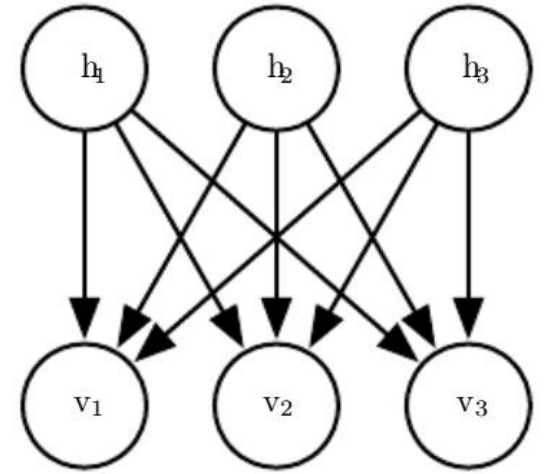
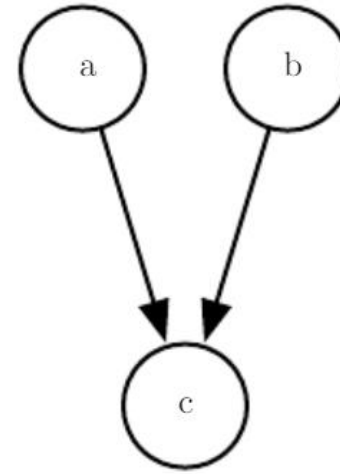
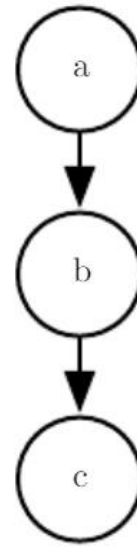
Zalety:

- Prostota implementacji, zwłaszcza jeśli mamy dostęp do rozkładów warunkowych.
- Skuteczność w modelach, gdzie zależności między zmiennymi można łatwo opisać za pomocą rozkładów warunkowych.

Wady:

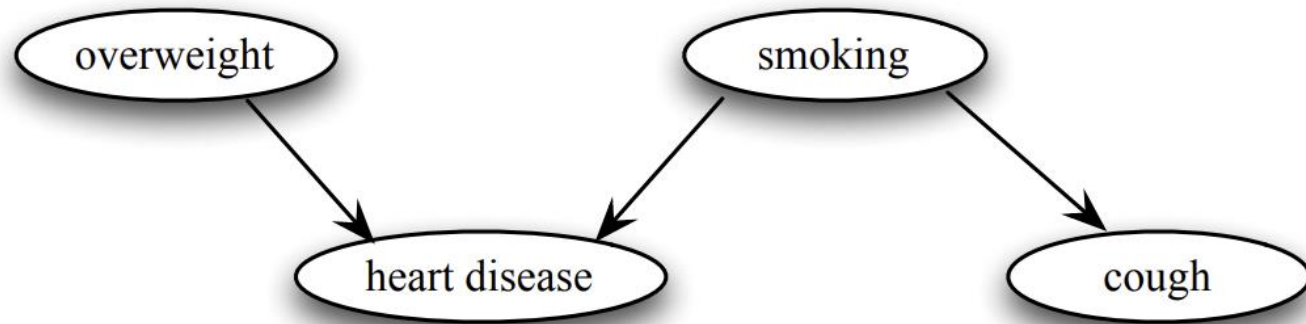
- **Wolna konwergencja:** Algorytm może wymagać wielu iteracji, zanim osiągnie stabilny stan.
- **Korelacja między próbkami:** Próbkki mogą być skorelowane, co wymaga zastosowania dodatkowych metod, np. „burn-in”.

Structured probabilistic models



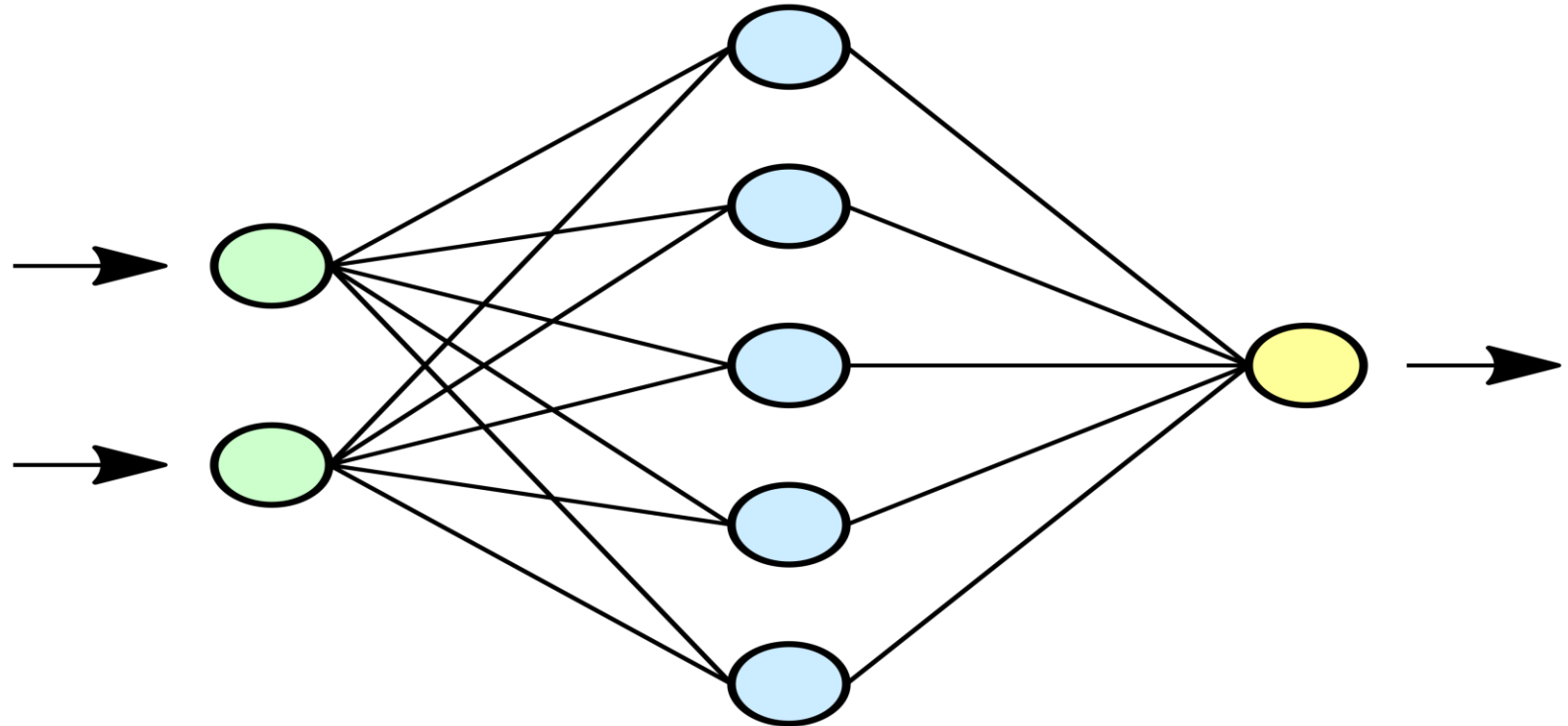
Directed Models

Są to modele stosowane, gdy mam wiedzę, że wartość w węzłach zależy od jednostronnie od innych węzłów. Wiemy, że wartość badanego węzła zależy od innych węzłów, ale w przeciwną stronę już nie.

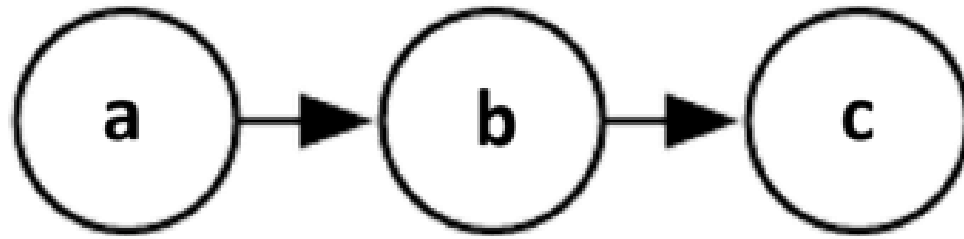


Directed Models

Są to modele stosowane, gdy mam wiedzę, że wartość w węzłach zależy od jednostronnie od innych węzłów. Wiemy, że wartość badanego węzła zależy od innych węzłów, ale w przeciwną stronę już nie.



Directed Models



$$p(a, b, c)$$

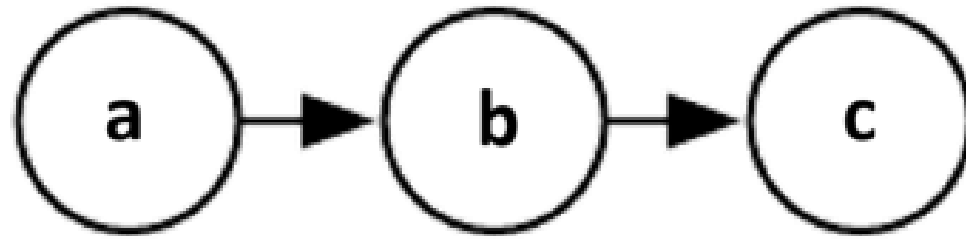
$$\mathbf{x} = (a, b, c)$$

$\boldsymbol{\pi}(x_i)$ - rodzice węzła x_i

$$p(\mathbf{x}) = \prod_i p(x_i | \boldsymbol{\pi}(x_i))$$

$p(x_i | \boldsymbol{\pi}(x_i))$ - *ang. local conditional probability*

Directed Models



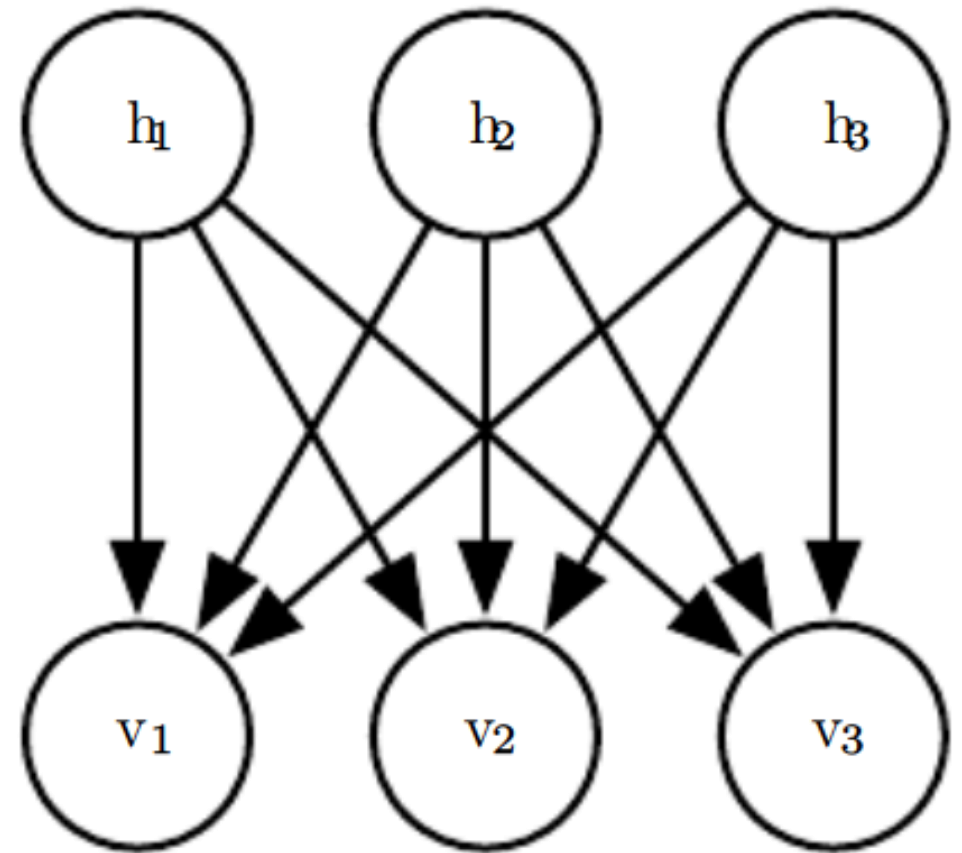
$$p(a, b, c)$$

$$p(\mathbf{x}) = \prod_i p(x_i | \boldsymbol{\pi}(x_i))$$

$$p(a, b, c) = p(a)p(b|a)p(c|b)$$

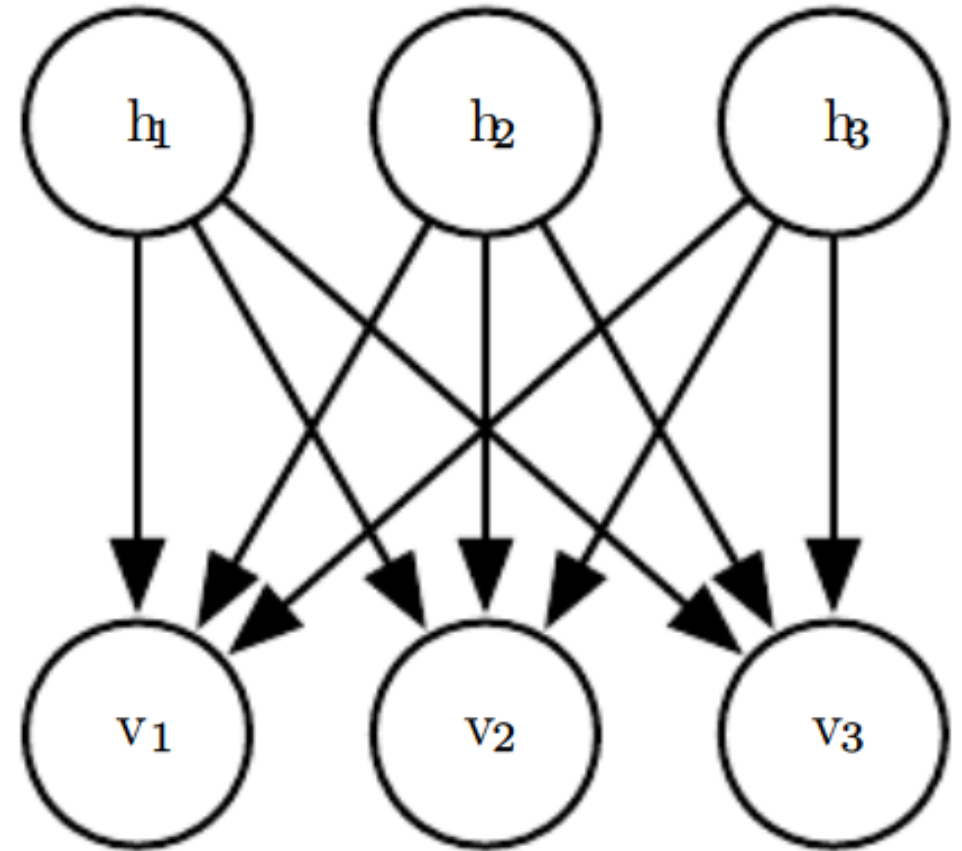
Directed Models

$$p(v_1, v_2, v_3, h_1, h_2, h_3) =$$

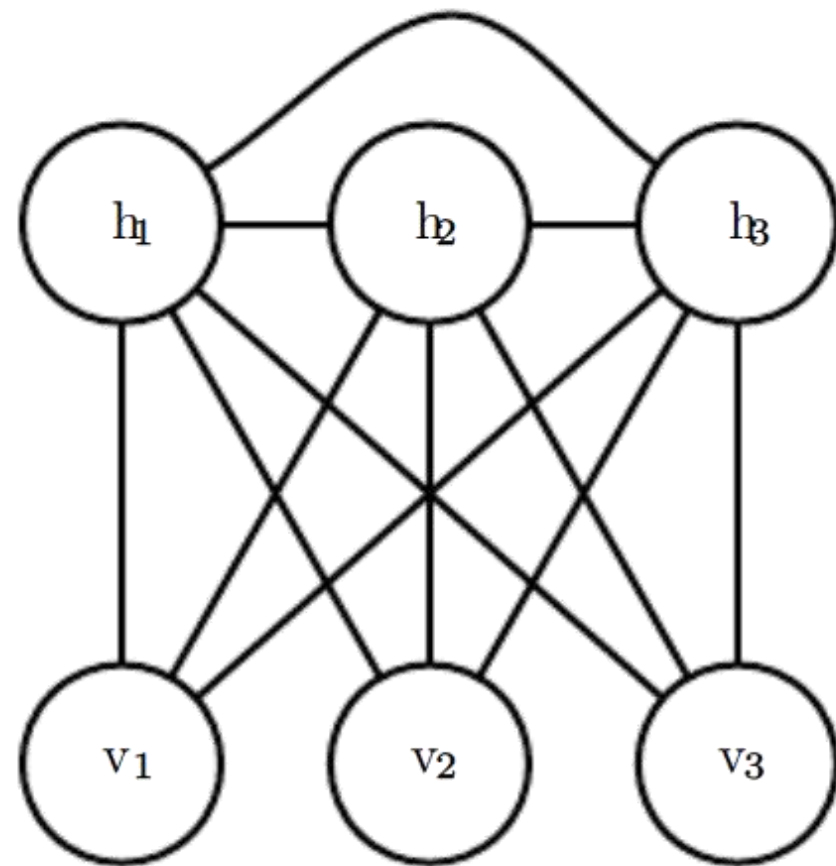
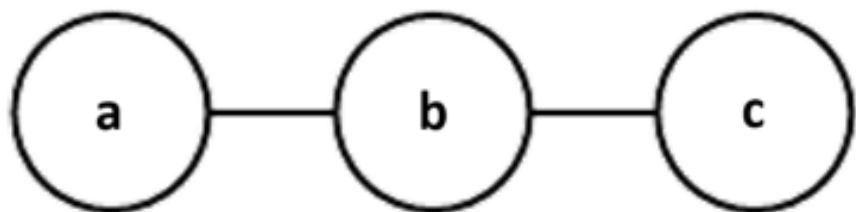


Directed Models

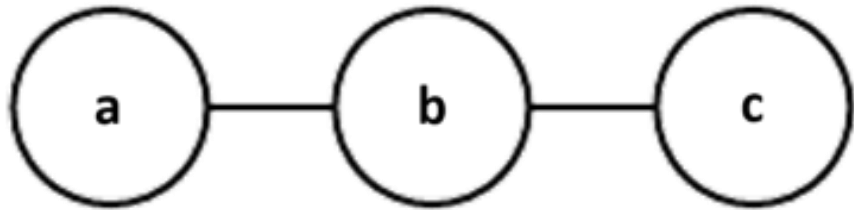
$$\begin{aligned} p(v_1, v_2, v_3, h_1, h_2, h_3) = & \\ p(h_1)p(h_2)p(h_3) \cdot & \\ p(v_1|h_1, h_2, h_3) \cdot & \\ p(v_2|h_1, h_2, h_3) \cdot & \\ p(v_3|h_1, h_2, h_3) & \end{aligned}$$



Undirected Models



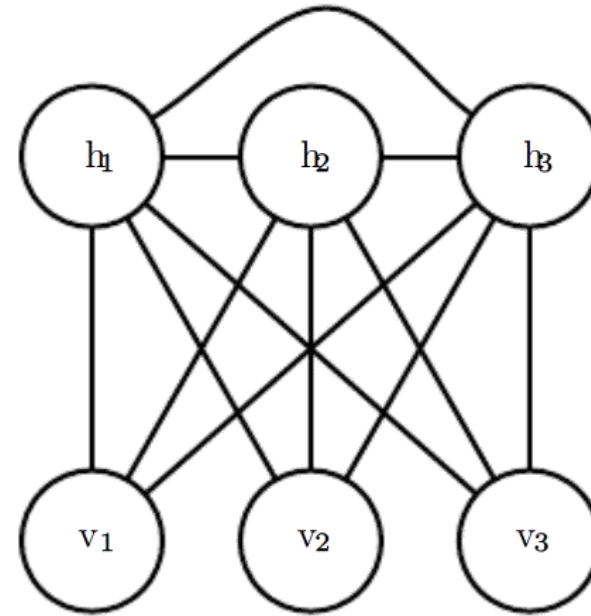
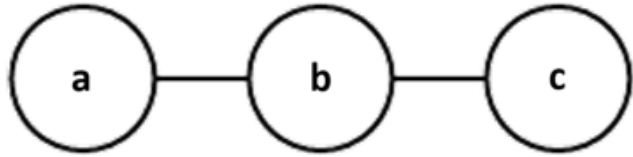
Undirected Models



$p(a, b, c)$

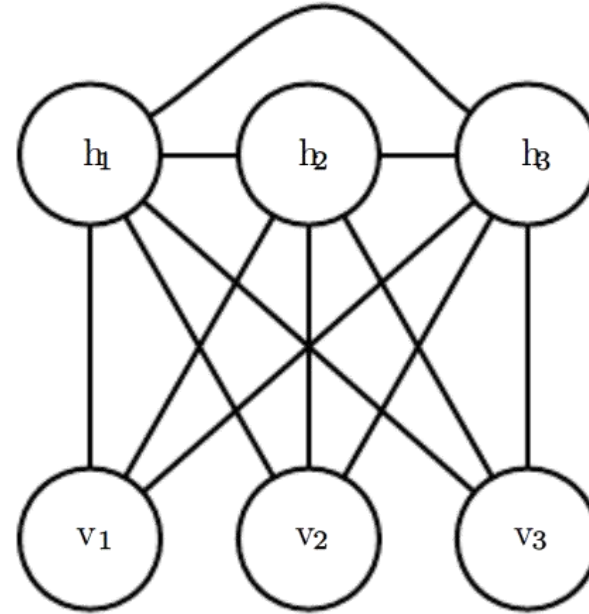
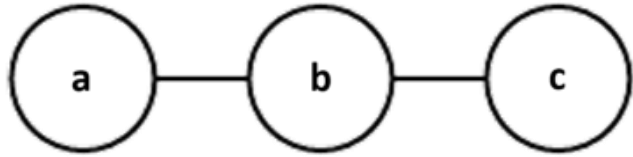


Undirected Models



Klika (*ang. clique*): podzbiór bezpośrednio połączonych ze sobą węzłów

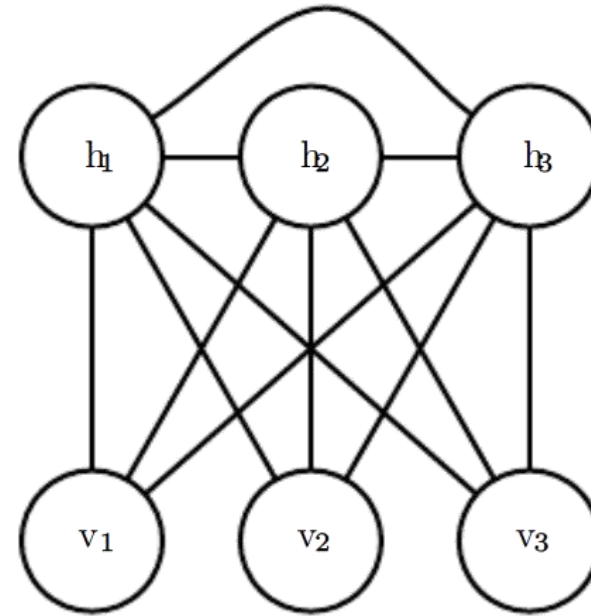
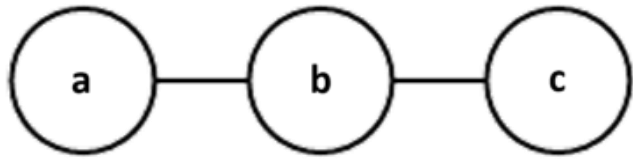
Undirected Models



Maksymalan klika (*ang. maximal clique*):

Klika do której nie można dołączyć żadnego wężła z grafu

Undirected Models



Potencjał kliki: Funkcja przypisana każdej kluce grafu, która modeluje lokalne zależności między zmiennymi w tej kluce.

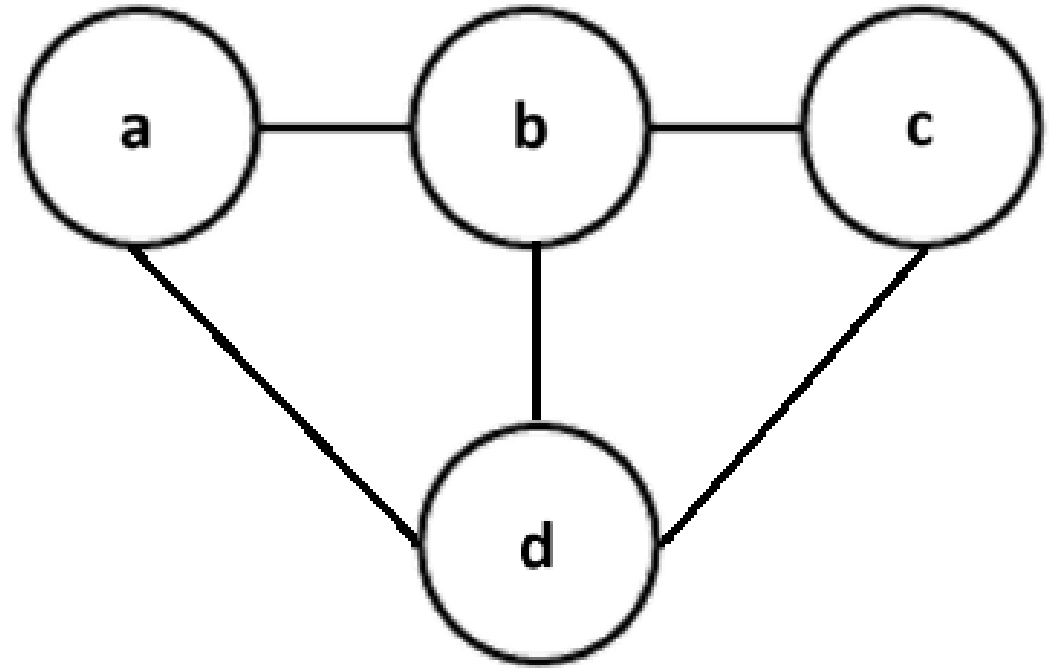
Potencjał kliki jest często oznaczany jako ψ_C , gdzie:

C – klika,

x_C – wartości zmiennych losowych w tej kluce,

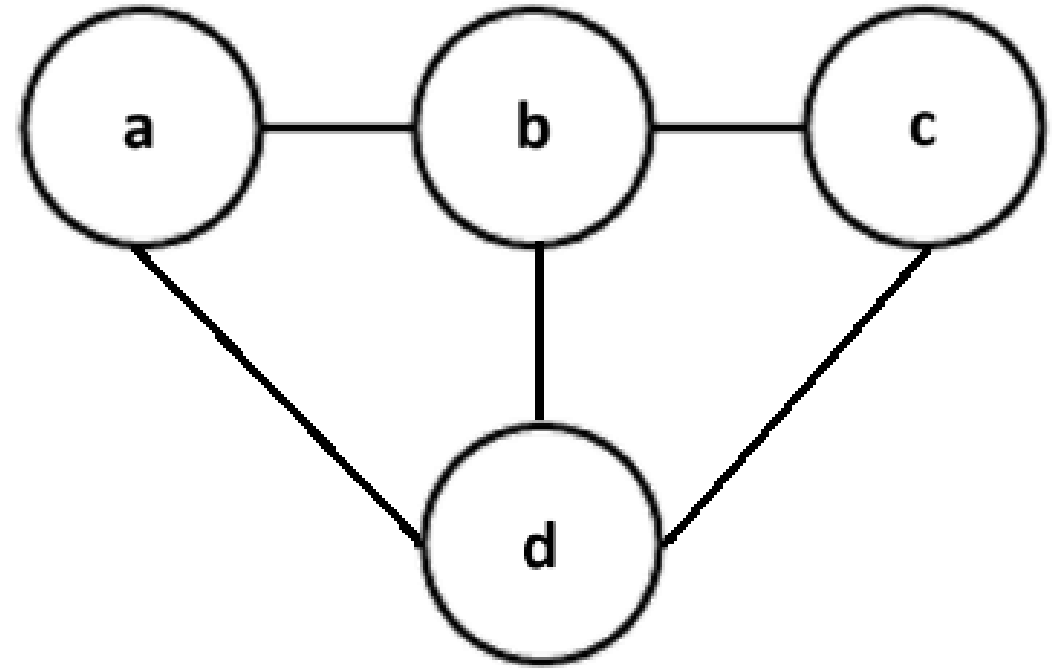
$\psi_C(\cdot)$ – funkcja potencjału, która przypisuje nieujemną wartość liczbową konfiguracji x_C .

Undirected Mode



Maksymalne kliki: {a, b, d}, {b, c, d}

Undirected Mode



Maksymalne kliki: {a, b, d}, {b, c, d}

Funkcja potencjału: $\psi_C(x, y, z) = \begin{cases} 1 & x = y \text{ albo } y = z \\ 2 & x = y = z \end{cases}$

Przykład



Asia



Hania



Paweł



Michał

Przykład



Asia



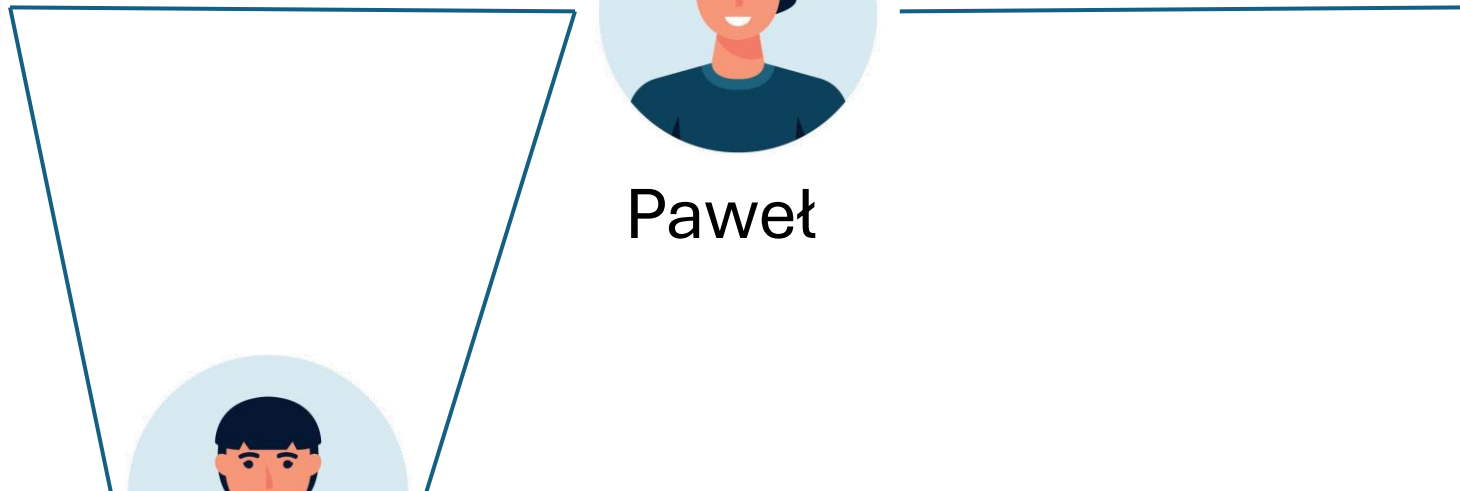
Paweł



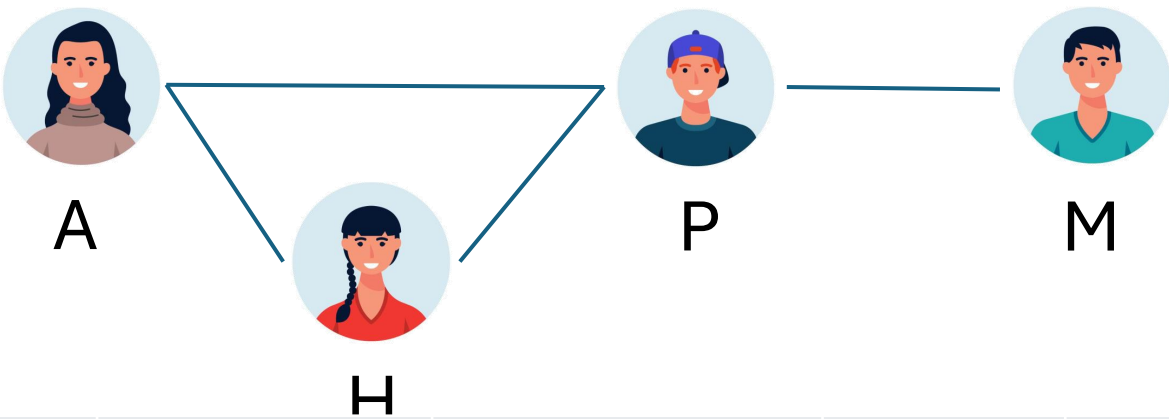
Michał



Hania



Przykład



Krawędzie grafu:

$$E = \{(A, H), (H, P), (A, P), (P, M)\}$$

$$X_A, X_H, X_P, X_M \in \{0, 1\}$$

$$\psi_{i,j}(x_i, x_j) = \begin{cases} e & x_i = x_j \\ 1 & x_i \neq x_j \end{cases}$$

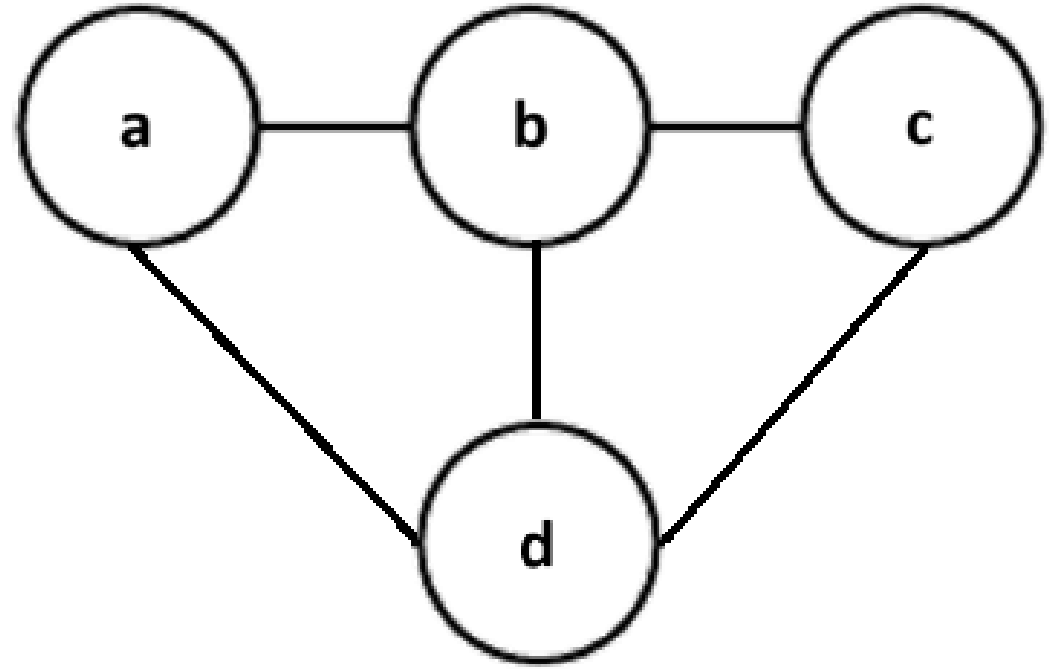
$$P(X_A, X_H, X_P, X_M) = \frac{1}{Z} \prod_{(i,j) \in E} \psi_{i,j}(x_i, x_j)$$

gdzie Z to stała normalizacyjna

$$Z = 205,341$$

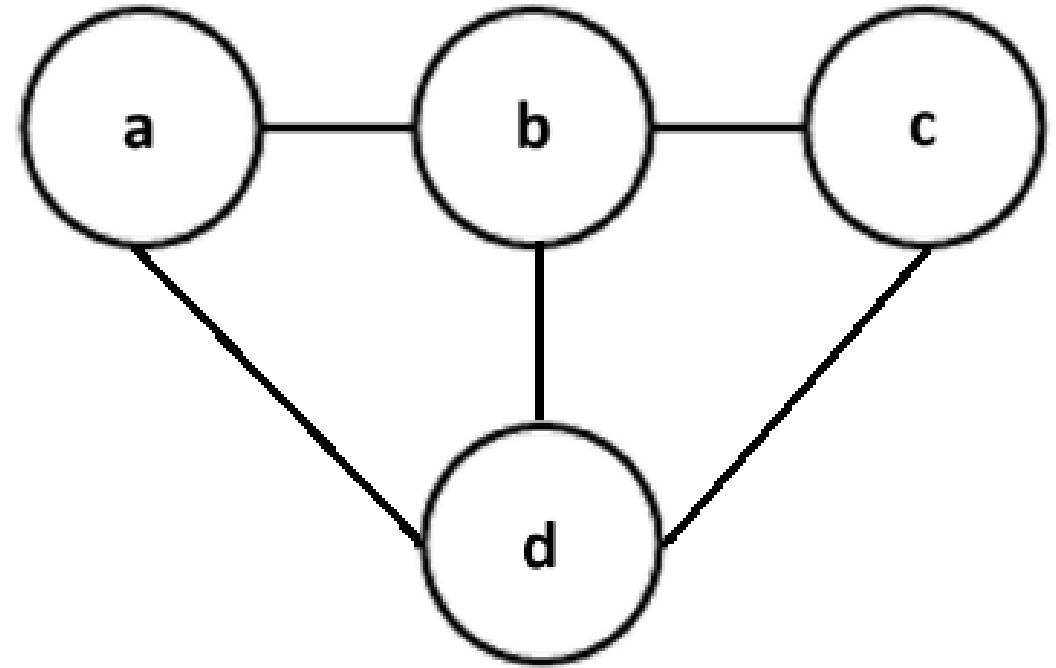
x				E				fun. potencjału				upd	Prob.
A	H	P	M	(A,H)	(H,P)	(A,P)	(P,M)	(A,H)	(H,P)	(A,P)	(P,M)		
0	0	0	0	(0,0)	(0,0)	(0,0)	(0,0)	e	e	e	e	54,5982	0,26589
			1	(0,0)	(0,0)	(0,0)	(0,1)	e	e	e	1	20,0855	0,09782
		1	0	(0,0)	(0,1)	(0,1)	(1,0)	e	1	1	1	2,71828	0,01324
			1	(0,0)	(0,1)	(0,1)	(1,1)	e	1	1	1	2,71828	0,01324
	0	0	0	(0,1)	(1,0)	(0,0)	(0,0)	1	1	e	e	7,38906	0,03598
			1	(0,1)	(1,0)	(0,0)	(0,1)	1	1	e	1	2,71828	0,01324
		1	0	(0,1)	(1,1)	(0,1)	(1,0)	1	e	1	1	2,71828	0,01324
			1	(0,1)	(1,1)	(0,1)	(1,1)	1	e	1	e	7,38906	0,03598
	1	0	0	(1,0)	(0,0)	(1,0)	(0,0)	1	e	1	e	7,38906	0,03598
			1	(1,0)	(0,0)	(1,0)	(0,1)	1	e	1	1	2,71828	0,01324
		1	0	(1,0)	(0,1)	(1,1)	(1,0)	1		e	1	2,71828	0,01324
			1	(1,0)	(0,1)	(1,1)	(1,1)	1	1	e	e	7,38906	0,03598
	1	0	0	(1,1)	(1,0)	(1,0)	(0,0)	e	1	1	e	7,38906	0,03598
			1	(1,1)	(1,0)	(1,0)	(0,1)	e	1	1	1	2,71828	0,01324
		1	0	(1,1)	(1,1)	(1,1)	(1,0)	e	e	e	1	20,0855	0,09782
			1	(1,1)	(1,1)	(1,1)	(1,1)	e	e	e	e	54,5982	0,26589

Undirected Mode



Prawdopodobieństwo: $p(a, b, c, d) = \frac{1}{Z} \prod_{C \in E} \psi_C(C)$

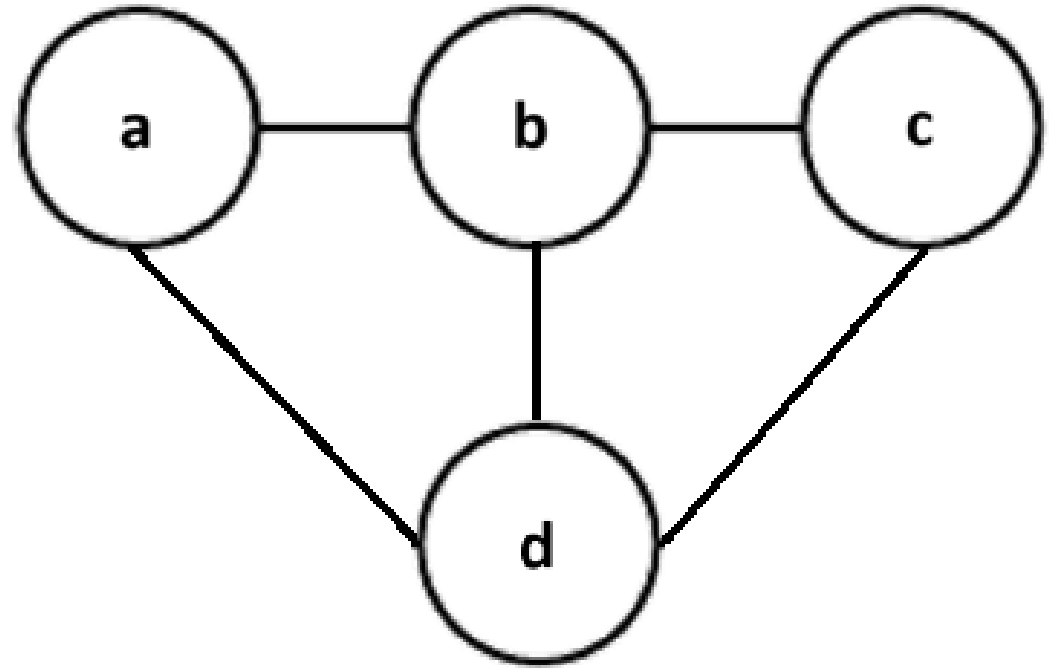
Undirected Mode



Prawdopodobieństwo: $p(\mathbf{x}) = \frac{1}{Z} \prod_{C \in E} \psi_C(C)$

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{C \in G} \psi_C(C) = \frac{1}{Z} e^{\ln(\prod_{C \in E} \psi_C(C))} = \frac{1}{Z} e^{\sum_{C \in E} \ln \psi_C(C)}$$

Undirected Mode



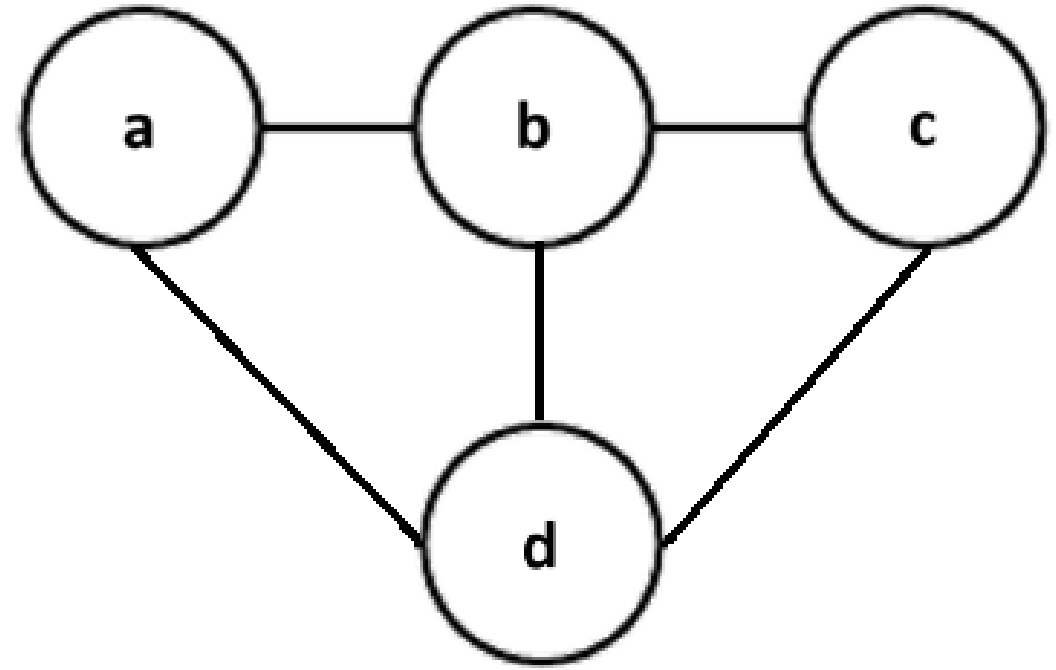
Prawdopodobieństwo: $p(\mathbf{x}) = \frac{1}{Z} \prod_{C \in E} \psi_C(C)$

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{C \in E} \psi_C(C) = \frac{1}{Z} e^{\ln(\prod_{C \in E} \psi_C(C))} = \frac{1}{Z} e^{\sum_{C \in E} \ln \psi_C(C)}$$

Energia:

$$E(\mathbf{x}) = - \sum_{C \in E} \ln \psi_C(C)$$

Undirected Mode



Prawdopodobieństwo: $p(\mathbf{x}) = \frac{1}{Z} \prod_{C \in \mathcal{G}} \psi_C(C)$

$$p(\mathbf{x}) = \frac{1}{Z} \prod_{C \in \mathcal{E}} \psi_C(C) = \frac{1}{Z} e^{\ln(\prod_{C \in \mathcal{E}} \psi_C(C))} = \frac{1}{Z} e^{\sum_{C \in \mathcal{E}} \ln \psi_C(C)} = \frac{e^{-E(\mathbf{x})}}{Z}$$

Energia:

$$E(\mathbf{x}) = - \sum_{C \in \mathcal{E}} \ln \psi_C(C)$$

Organiczna Maszyna Boltzmannna

ang. Restricted Boltzmann Machine (RBM)

Czym są RBM?

Warstwa widoczna: $\mathbf{v} = \{v_1, \dots, v_d\}$

Warstwa ukryta: $\mathbf{h} = \{h_1, \dots, h_H\}$

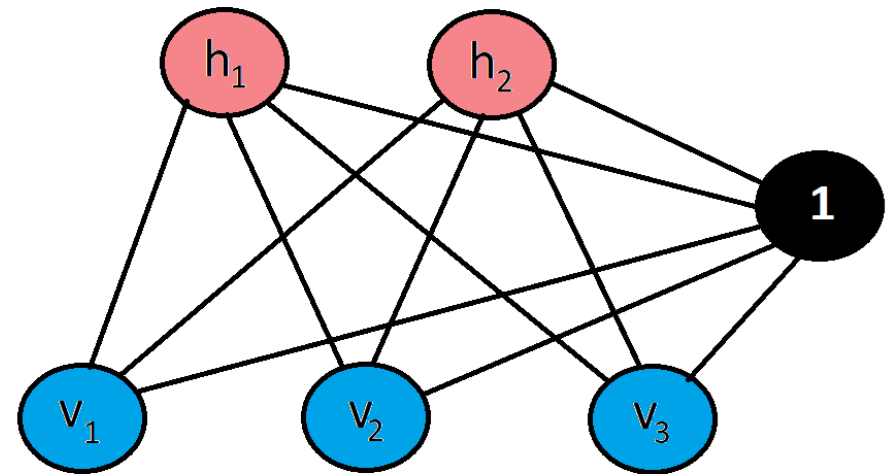
Binary values: $(\mathbf{v}, \mathbf{h}) \in \{0,1\}^{d+H}$

Parametry modelu:

\mathbf{w} – macierz wag o wymiarach d na H

\mathbf{b} – bias warstwy widocznej, wektor długości d

\mathbf{c} – bias warstwy ukrytej, wektor długości H



Czym są RBM?

Warstwa widoczna: $\mathbf{v} = \{v_1, \dots, v_d\}$

Warstwa ukryta: $\mathbf{h} = \{h_1, \dots, h_H\}$

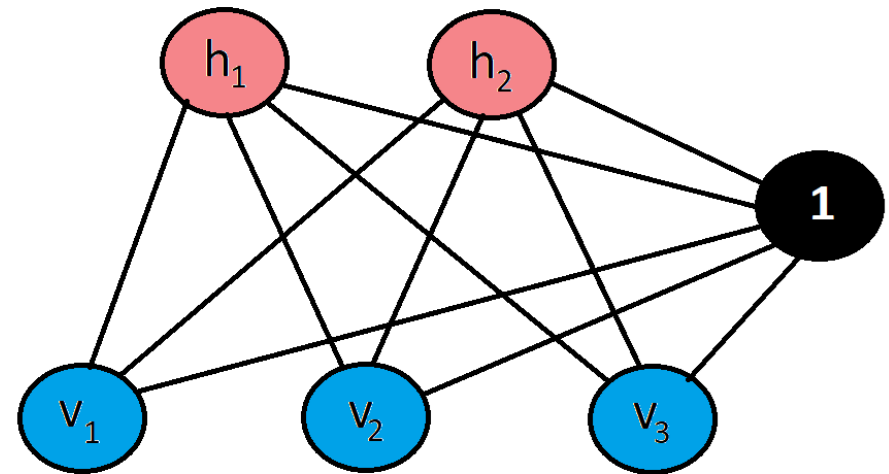
Binary values: $(\mathbf{v}, \mathbf{h}) \in \{0,1\}^{d+H}$

Model wyznacza **prawdopodobieństwa** aktywowania neuronu w danej warstwie na podstawie aktywacji neuronów w przeciwnej warstwie i parametrów modelu.

σ to funkcja sigmoidalna.

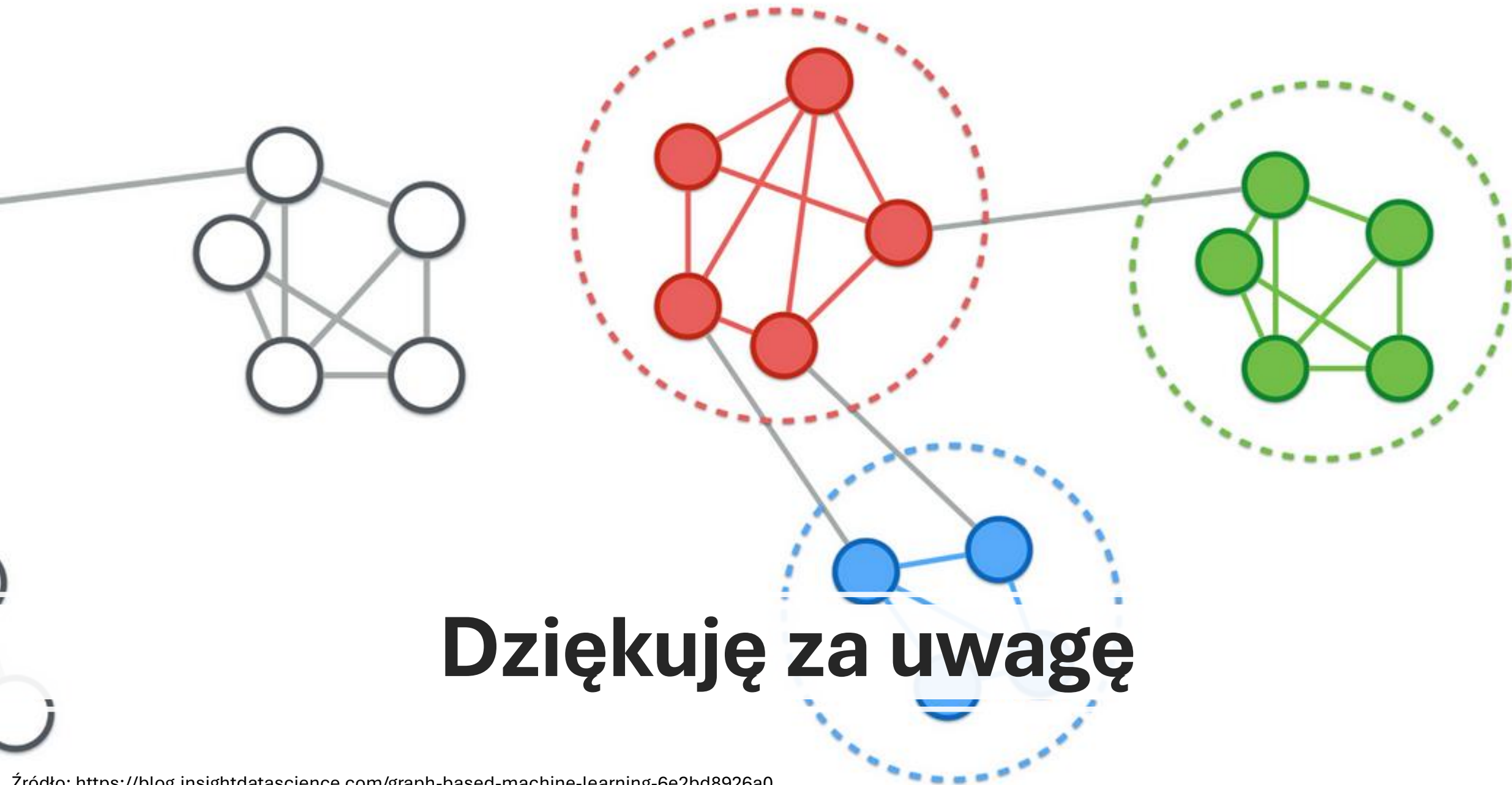
$$p(h_j = 1|\mathbf{v}) = \sigma\left(c_j + \sum_{i=1}^d w_{ij}v_i\right)$$

$$p(v_i = 1|\mathbf{h}) = \sigma\left(b_i + \sum_{j=1}^H w_{ij}h_j\right)$$



Pytania / Komentarze / Uwagi





Dziękuję za uwagę