

# Om Markov Chain Monte Carlo

Gunnar Englund  
Matematisk statistik  
KTH

Ht 2001

## 1 Inledning

Markov Chain Monte Carlo (MCMC) är en modern teknik att simulera komplicerade fördelningar som har fått stora tillämpningar i Matematisk statistik, framför allt vad gäller bildbehandling, statistisk mekanik, Bayesiansk statistik etc. Till skillnad från vanlig simulering genererar man en sekvens utfall enligt en Markovkedja med målfördelningen som stationär (och förhoppningsvis asymptotisk) fördelning.

## 2 Ergodisatser och stora talens lag

Om en Markovkedja är ergodisk på tillståndsrummet  $E$  med den asymptotiska fördelningen  $\pi(x)$ ,  $x \in E$  kan man (med godtycklig noggrannhet) beräkna storheter av typen

$$E_\pi(f(X)) = \sum_{x \in E} \pi(x) f(x)$$

för godtyckliga funktioner  $f$  genom att för en enskild realisering (tidsutveckling)  $X_1, X_2, \dots$  beräkna tidsmedelvärdet

$$\hat{f}_n = \frac{1}{n} \sum_{i=1}^n f(X_i).$$

Man kan visa att  $\hat{f}_n \rightarrow E_\pi(f(X))$  då  $n \rightarrow \infty$  om kedjan är ergodisk.

**Exempel 2.1** Om vi tar

$$f(x) = \begin{cases} 1 & \text{om } x = j \\ 0 & \text{annars} \end{cases}$$

ser vi att  $\hat{f}_n$  =relativa frekvensen av besök i tillståndet  $j$  och att  $E_\pi(f(X)) = \pi(j)$ . Detta betyder att vi kan bestämma  $\pi$ -vektorn ur en realisering.  $\square$

**Exempel 2.2** Om vi tar  $f(x) = x$ ,  $x \in E$  är

$$\hat{f}_n = \sum_{x \in E} x \hat{p}_x$$

där  $\hat{p}_x$ =relativa frekvensen av besök i tillstånd  $x$ . Å andra sidan betyder  $E_\pi(f(X))$  väntevärdet i stationära (asymptotiska) fördelningen.  $\square$

**Exempel 2.3** Om vi tar  $g(x) = x^2$ ,  $x \in E$  får vi

$$\hat{g}_n = \sum_{x \in E} x^2 \hat{p}_x$$

som alltså kan användas för att skatta variansen i  $\pi$ -fördelningen ur en realisering genom  $\hat{g}_n - (\hat{f}_n)^2$  där  $\hat{f}_n$  erhålls ur föregående exempel. (Jämför  $V(X) = E(X^2) - (E(X))^2$ .)  $\square$

Denna typ av ergodisatser svarar precis mot stora talens lag dvs att (med sannolikhet 1)

$$\frac{1}{n}(X_1 + X_2 + \dots + X_n) \rightarrow E(X)$$

där  $X_1, X_2, \dots$  är oberoende likafördelade stokastiska variabler med samma fördelning som  $X$ . På precis samma sätt gäller för godtyckligt  $f$  att

$$\frac{1}{n}(f(X_1) + f(X_2) + \dots + f(X_n)) \rightarrow E(f(X)).$$

Det fiffiga med ergodiska Markovkedjor är alltså att de vad gäller tidsmedelvärden fungerar på samma sätt som om det rörde sig om oberoende variabler. Dock kan konvergensthastigheten bero av Markovkedjans utseende.

Vad som beskrivits ovan gäller precis oförändrat om de ingående variablerna är flerdimensionella, t ex  $d$ -dimensionella. Vi har alltså i stället för  $X_n$  en  $d$ -dimensionell variabel  $\mathbf{X}_n = (X_{n1}, X_{n2}, \dots, X_{nd})$ . Den stokastiska processen  $\mathbf{X}_1, \mathbf{X}_2, \dots$  utgör alltså en Markovkedja med  $d$ -dimensionellt tillståndsrum. Om tillståndsrummet är ändligt eller uppräkneligt oändligt är det egentligen ingen skillnad mot det en-dimensionella fallet.

Med hjälp av en lång realisering av Markovkedjan kan man alltså approximativt (med godtycklig noggrannhet) beräkna en massa intressanta storheter rörande  $\pi$ -fördelningen, t ex sannolikheter, väntevärden, medianer, kovarianser mellan komponenter etc.

### 3 Metropolis-Hastings algoritmen

Vad man alltså vill göra är att simulera en realisering av Markovkedjan och därigenom kunna approximativt beräkna  $E_\pi(f(X))$  genom att beräkna  $\hat{f}_n$  för

något stort  $n$  (kanske  $n=10000$  eller så) och utnyttja det faktum att kedjan är ergodisk och att alltså  $\hat{f}_n \rightarrow E_\pi(f(X))$ .

I Markov Chain Monte Carlo (MCMC) använder man detta för att kunna simulera en given  $d$ -dimensionell fördelning där  $d$  är ganska stor (kanske  $d = 100$  eller liknande). Uppgiften är alltså på sätt och vis omvänd mot den vanliga där man ju typiskt vill undersöka om en given Markovkedja är ergodisk och vad den i så fall har för asymptotisk fördelning. Här vill man i stället konstruera en Markovkedja som har en viss given fördelning till asymptotisk fördelning. Detta visar sig mycket lätt genom den så kallade Metropolis-Hastings algoritmen. Oftast är det dessutom mycket lätt att se att kedjan blir ergodisk genom att t ex visa att den blir irreducibel och aperiodisk.

Idén i algoritmen är att man utifrån aktuellt värde  $\mathbf{x}$  ger ett förslag  $\mathbf{y}$  på nästa värde och sen väljer att acceptera eller förkasta förslaget med en listigt avpassad sannolikhet. Ett förkastat värde innebär att man ligger kvar. Vi har alltså en förslagsfördelning

$$q(\mathbf{x}, \mathbf{y}) = P(\text{föreslå } \mathbf{y} \mid \text{aktuellt värde är } \mathbf{x}).$$

Denna förslagsfördelning kan förvånande nog se ut nästan hur som helst. Förslaget accepteras med sannolikheten  $\alpha(\mathbf{x}, \mathbf{y}) = P(\text{acceptera förslaget } \mathbf{y} \mid \text{givet från läget } \mathbf{x})$ . Denna sannolikhet väljes listigt till

$$\alpha(\mathbf{x}, \mathbf{y}) = \min \left( 1, \frac{\pi(\mathbf{y})q(\mathbf{y}, \mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x}, \mathbf{y})} \right).$$

En liten, men viktig observation är att målfördelningen ( $\pi(\mathbf{x}), \mathbf{x} \in E$ ) förekommer i både täljare och nämnare och detta betyder att den inte behöver vara normerad, dvs uppfylla

$$\sum_{\mathbf{x} \in E} \pi(\mathbf{x}) = 1.$$

Detta till synes ovidkommande faktum är av ytterst stort intresse eftersom just normeringen i många tillämpningar är mycket komplicerad då  $E$  är stort.

Algoritmen ger uppenbarligen upphov till en tidshomogen Markovkedja, eftersom nästa tillstånd bara beror av närmast föregående tillstånd och dynamiken är oberoende av tiden.

Sammanfattningsvis har vi alltså algoritmen

- 1) Om  $\mathbf{X}_n = \mathbf{x}$  : Förslag  $\mathbf{y}$  till  $\mathbf{X}_{n+1}$  enligt förslagsfördelning  $q(\mathbf{x}, \mathbf{y})$ .
- 2) Acceptera förslaget  $\mathbf{X}_{n+1} = \mathbf{y}$  givet från  $\mathbf{x}$  med sannolikhet

$$\alpha(\mathbf{x}, \mathbf{y}) = \min \left( 1, \frac{\pi(\mathbf{y})q(\mathbf{y}, \mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x}, \mathbf{y})} \right)$$

annars  $\mathbf{X}_{n+1} = \mathbf{x}$

Algoritmen innebär att vi för den resulterande Markovkedjan får övergångsmatrisen  $P$  där för  $\mathbf{y} \neq \mathbf{x}$  vi har

$$P(\mathbf{x}, \mathbf{y}) = P(\mathbf{X}_{n+1} = \mathbf{y} \mid \mathbf{X}_n = \mathbf{x}) = q(\mathbf{x}, \mathbf{y})\alpha(\mathbf{x}, \mathbf{y})$$

eller i ord: För att gå från  $\mathbf{x}$  till  $\mathbf{y}$  måste vi först föreslå  $\mathbf{y}$  från läget  $\mathbf{x}$  och sen acceptera förslaget. På diagonalen i  $P$ -matrisen har vi

$$P(\mathbf{x}, \mathbf{x}) = 1 - \sum_{\mathbf{y} \neq \mathbf{x}} P(\mathbf{x}, \mathbf{y}).$$

Detta att ligga kvar i  $\mathbf{x}$  kan uppstå på två sätt. För det första kan vi tvingas att inte acceptera ett givet förslag eller för det andra, råka föreslå  $\mathbf{x}$  - detta förslag accepteras ju eftersom  $\alpha(\mathbf{x}, \mathbf{x}) = 1$ .

## 4 Tidsreversibilitet

Att den resulterande Markovkedjan får  $\pi(\mathbf{x})$ ,  $\mathbf{x} \in E$  till stationär fördelning beror på att kedjan blir reversibel med avseende på fördelningen  $\pi(\mathbf{x})$ ,  $\mathbf{x} \in E$ , dvs uppfyller

$$\pi(\mathbf{x})P(\mathbf{x}, \mathbf{y}) = \pi(\mathbf{y})P(\mathbf{y}, \mathbf{x})$$

för alla  $\mathbf{x}$  och  $\mathbf{y}$ . Detta inses lätt eftersom för  $\mathbf{x} \neq \mathbf{y}$  har vi

$$\begin{aligned} \pi(\mathbf{x})P(\mathbf{x}, \mathbf{y}) &= \pi(\mathbf{x})q(\mathbf{x}, \mathbf{y})\alpha(\mathbf{x}, \mathbf{y}) = \pi(\mathbf{x})q(\mathbf{x}, \mathbf{y}) \min\left(1, \frac{\pi(\mathbf{y})q(\mathbf{y}, \mathbf{x})}{\pi(\mathbf{x})q(\mathbf{x}, \mathbf{y})}\right) = \\ &= \min(\pi(\mathbf{x})q(\mathbf{x}, \mathbf{y}), \pi(\mathbf{y})q(\mathbf{y}, \mathbf{x})) = \pi(\mathbf{y})P(\mathbf{y}, \mathbf{x}). \end{aligned}$$

För  $\mathbf{x} = \mathbf{y}$  är villkoret trivialt uppfyllt.

Man kan lätt visa att en Markovkedja som är reversibel med avseende på en fördelning  $\pi$  har  $\pi$  till stationär fördelning, ty om vi summerar reversibilitetsvillkoret över  $\mathbf{x}$  erhåller vi

$$\sum_{\mathbf{x} \in E} \pi(\mathbf{x})P(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{x} \in E} \pi(\mathbf{y})P(\mathbf{y}, \mathbf{x}) = \pi(\mathbf{y}) \sum_{\mathbf{x} \in E} P(\mathbf{y}, \mathbf{x}) = \pi(\mathbf{y})$$

eftersom vi har radsumman 1 i  $P$ -matrisen. Alltså gäller

$$\pi(\mathbf{y}) = \sum_{\mathbf{x} \in E} \pi(\mathbf{x})P(\mathbf{x}, \mathbf{y})$$

vilket (jämför  $\mathbf{p}^{(1)} = \mathbf{p}^{(0)}P$ ) säger att  $\pi(\mathbf{x})$ ,  $\mathbf{x} \in E$  är en stationär fördelning. Är kedjan ergodisk är denna stationära fördelning också den asymptotiska oavsett startvärde.

Man kan tolka

$$\pi(\mathbf{x})P(\mathbf{x}, \mathbf{y}) = P(\mathbf{X}_n = \mathbf{x})P(\mathbf{X}_{n+1} = \mathbf{y} \mid \mathbf{X}_n = \mathbf{x}) = P(\mathbf{X}_n = \mathbf{x} \& \mathbf{X}_{n+1} = \mathbf{y})$$

som ett "flöde" från  $\mathbf{x}$  till  $\mathbf{y}$  och på samma sätt är högerledet

$$\pi(\mathbf{y})P(\mathbf{y}, \mathbf{x}) = P(\mathbf{X}_n = \mathbf{y})P(\mathbf{X}_{n+1} = \mathbf{x} | \mathbf{X}_n = \mathbf{y}) = P(\mathbf{X}_n = \mathbf{y} \& \mathbf{X}_{n+1} = \mathbf{x})$$

"flödet" från  $\mathbf{y}$  till  $\mathbf{x}$ . Tidsreversibiliteten innebär alltså att dessa flöden balanserar varandra och det är därför naturligt att kedjan har  $\pi$  som stationär fördelning.

Man kan säga att acceptans-sannolikheten  $\alpha(\mathbf{x}, \mathbf{y})$  avpassats så att dessa flöden precis balanserar varandra.

**Exempel 4.1** Låt  $\chi = I \times J$ -gitter med 0:or och 1:or. och låt tillståndsrummet vara  $E = \{x \in \chi \text{ två grannar inte båda } 1\}$ . De tillåtna konfigurationerna är alltså  $I \times J$ -matriser av 0:or och 1:or men där det inte kan stå två 1:or intill varandra i sidled eller höjddled. Om vi tar  $J = I = 10$  har vi ett gitter med 100 punkter och man kan inse att det inte ens är en lätt uppgift att beräkna storleken av  $E$ , dvs att bestämma hur många "tillåtna" konfigurationer det finns!

Låt  $\pi$  vara en fördelning på  $\chi$  som är 0 utanför  $E$ , t ex  $\pi(x) = \text{konstant}$  för  $x \in E$  och 0 för övrigt (likformig fördelning på  $E$ ). Notera att vi faktiskt inte vet hur stort  $E$  är!

Som förslagsfördelning (dvs  $q(x, y)$ ) kan vi ta:

Välj nod  $(i, j)$  på måfå med sannolikhet  $1/IJ$ . Välj värde i noden som 0 eller 1 med sannolikhet  $1/2$  vardera.

Eventuellt inträffar att  $y \notin E$  dvs att förslaget innebär en "otillåten" konfiguration, men då är  $\pi(y) = 0$  och alltså  $\alpha(x, y) = 0$ , dvs sådana förslag ignoreras och kedjan ligger kvar där den var.

Om  $y \in E$  (dvs en tillåten konfiguration) accepteras förslaget ty

$$\alpha(x, y) = \min \left( 1, \frac{\pi(y)q(y, x)}{\pi(x)q(x, y)} \right) = 1$$

eftersom  $q(x, y) = q(y, x)$  och  $\pi(x) = \pi(y)$ .

Kedjan är uppenbarligen irreducibel och aperiodisk, dvs ergodisk! Man kan inse detta genom att givet två konfigurationer  $x$  och  $y$  så kan vi börja med att plocka bort alla 1:or i  $x$  och sen sätta dit 1:orna som ingår i  $y$  - detta visar irreducibiliteten. Uppenbarligen har vi positiv sannolikhet att ligga kvar i t o m varje tillstånd vilket visar aperiodiciteten. Eftersom kedjan är ändlig är den ergodisk. .

Med  $f(x) = \text{antal } 1\text{:or i } x$  får vi medelantal 1:or ur

$$\hat{f}_n = \frac{1}{n} \sum_{i=1}^n f(x_i).$$

Med  $I = J = 10$  erhålls ( $n = 10000$ )

- 1)  $E(\text{antal 1:or}) \approx 23.40$ ,
- 2)  $E(\text{max antal 1:or i raderna}) \approx 3.88$ .
- 3) Dessa bådas korrelation  $\approx 0.4240$ .

Det blir nästan lika lätt med ett allmänt  $\pi$  men då blir acceptanssannolikheten

$$\alpha(x, y) = \min \left( 1, \frac{\pi(y)}{\pi(x)} \right)$$

□

## 5 Specialfall av Metropolis-Hastings algoritmen

### 5.1 Metropolis algoritmen

Ett intressant specialfall är om vi väljer förslagsfördelningen så att  $q(\mathbf{x}, \mathbf{y}) = q(\mathbf{y}, \mathbf{x})$  dvs att sannolikheten att föreslå  $\mathbf{y}$  från  $\mathbf{x}$  är lika stor som sannolikheten att föreslå  $\mathbf{x}$  från  $\mathbf{y}$ . Vi får då acceptanssannolikheten  $\alpha(\mathbf{x}, \mathbf{y}) = \min(1, \pi(\mathbf{y})/\pi(\mathbf{x}))$ . Detta kallas Metropolis algoritmen. I exemplet ovan användes denna typ av förslagsfördelning.

### 5.2 Oberoende simulering

Om man tar  $q(\mathbf{x}, \mathbf{y}) = \pi(\mathbf{y})$  dvs att förslagsfördelningen är densamma som målfördelningen ser man att  $\alpha(\mathbf{x}, \mathbf{y}) = 1$  och alla förslag accepteras alltså och man får oberoende slumpstal med fördelningen  $\pi(\mathbf{x})$ ,  $\mathbf{x} \in E$ . Oberoende simulering (sedvanlig Monte Carlo-metod) är alltså ett specialfall av Metropolis-Hastings algoritmen.

### 5.3 Koordinatvis uppdatering

En mycket intressant variant då man vill simulera en flerdimensionell fördelning är att uppdatera  $\mathbf{X}_n = (X_{n1}, X_{n2}, \dots, X_{nd})$  en koordinat i taget.

Vår målfördelning

$$\pi_{X_1, X_2, \dots, X_d}(x_1, x_2, \dots, x_d) = \pi_{\mathbf{X}}(\mathbf{x})$$

är  $d$ -dimensionell och vi vill uppdatera en koordinat i taget. Vid uppdatering av koordinat  $i$  är alltså alla de övriga konstanta och den enda förändringen som sker är ett byte av värde i koordinat  $i$ . Man har alltså vid uppdateringen av koordinat  $i$  en förslagsfördelning  $q_i(\mathbf{x}, \mathbf{y})$  som är 0 om inte  $\mathbf{y}$  överensstämmer med  $\mathbf{x}$  för alla koordinater utom nr  $i$ . Detta betyder att förslagen från  $q_i$  lämnar alla koordinater utom nr  $i$  oförändrade. Till denna förslagsfördelning  $q_i$  hör en acceptanssannolikhet  $\alpha_i(\mathbf{x}, \mathbf{y})$  definierad som ovan fast med utgångspunkt i  $q_i$ . Detta ger upphov till en övergångsmekanism beskriven av en matris  $\mathbf{P}_i$ .

I allmänhet löper man systematiskt igenom alla koordinaterna innan man anser att man fått en ny observation i Markovkedjan. Ett steg i Markovkedjan svarar alltså mot en uppdatering av samtliga koordinater. Om vi låter  $\mathbf{P}_i$  vara övergångsmatrisen vid uppdatering av koordinat  $i$  består ett steg i Markovkedjan av att vi först applicerar  $\mathbf{P}_1$  och sedan  $\mathbf{P}_2$  osv t o m  $\mathbf{P}_d$ . Markov-kedjans övergångsmatris är alltså

$$\mathbf{P} = \mathbf{P}_1 \mathbf{P}_2 \cdots \mathbf{P}_d.$$

Orsaken till att vi vill uppdatera alla koordinater innan vi anser oss ha fått ett nytt tillstånd är att kedjan annars inte skulle vara tidshomogen. En alternativ metod som ger en tidshomogen Markovkedja skulle vara att lotta fram en koordinat  $I$  (lika sannolikheter för alla  $d$  koordinater) och sen uppdatera denna med hjälp av motsvarande matris  $\mathbf{P}_i$ .

## 5.4 Gibbs-sampling

Ett mycket viktigt, intressant och praktiskt användbart specialfall av koordinatvis uppdatering enligt Metropolis-Hastings algoritmen är Gibbs-sampling som innebär ett mycket speciellt val av förslagsfördelning  $q_i$  vid uppdateringen av koordinat  $i$ . Man låter helt enkelt förslagsfördelningen  $q_i$  vara den betingade fördelningen för  $X_i$  givet alla de övriga koordinaterna.

Vi försöker alltså konstruera en  $d$ -dimensionell Markovkedja med målfördelningen

$$\pi_{X_1, X_2, \dots, X_d}(x_1, x_2, \dots, x_d) = \pi_{\mathbf{X}}(\mathbf{x})$$

som täthet (sannolikhetsfunktion). Vid Gibbs-sampling gör man alltså detta genom att uppdatera en koordinat i taget med den betingade fördelningen för denna komponent givet alla de övriga, dvs för koordinat nr  $i$  låter vi förslagsfördelningen vara

$$P(X_i = y \mid X_1 = x_1; X_2 = x_2; \cdots; X_{i-1} = x_{i-1}; X_{i+1} = x_{i+1}; \cdots; X_d = x_d)$$

där vi alltså bara ändrar koordinat  $i$  och lämnar de övriga oförändrade. Man kan ganska lätt visa att acceptanssannolikheten för detta förslag är 1.

## 5.5 Exempel på Gibbs-sampling: Ising-modell för spinn

Som ett icke-trivialt exempel på att det kan vara lätt att simulera en mycket komplicerad fördelning tar vi följande exempel med anknytning till Statistisk mekanik.

En enkel två-dimensionell modell för magnetiskt spinn är att vi har ett  $n \times m$  gitter

$$S = \{(i, j) : i = 1, 2, \dots, n, j = 1, 2, \dots, m\}.$$

I dessa  $nm$  punkter finns slumpmässiga spinn  $X_{i,j}$  som kan anta värdena  $+1$  och  $-1$  som svarar mot spinn upp respektive spinn ner. Vi låter  $\mathbf{X} = (X_{i,j}, (i, j) \in S)$ . Vi har då ett utfallsrum  $\Omega$  för  $\mathbf{X}$  med  $2^{nm}$  tänkbara utfall.

Notera att även om gittret är så litet som  $100 \times 100$  finns  $2^{10000}$  tänkbara konfigurationer. Vi vill nu simulera ett sådant utfall. Problemet är att vi antar att det finns en (temperaturberoende) växelverkan mellan de olika gitterpunkterna och då framför allt mellan gitterpunkter som ligger nära varandra. Vi definierar en "omgivning"  $N(i, j)$  till gitterpunkten  $(i, j)$  t ex

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

där  $(i, j)$  ligger i mitten och 1:or står för "omgivningspunkter". Ett annat tänkbart utseende på "omgivningen" skulle vara

$$\begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

Växelverkan sker primärt endast mellan mittpunkten och dess grannar, men detta beroende "fortplantas" genom gittret till att bli ett beroende som kanske sträcker sig över stora områden. Ofta knyter man ihop vänster- och högerkanten, respektive över- och underkanten av  $S$  för att slippa randeffekter.

Vi definierar en energi

$$E(\mathbf{x}) = -J \sum_{(i,j) \in S} \sum_{(i',j') \in N(i,j)} x_{i,j} x_{i',j'}$$

där  $J > 0$  kvantifierar växelverkans storlek. Detta innebär att energin är låg (stabil tillstånd) om näraliggande gitterpunkter har samma spinn-riktning.

Vi antar nu för ett fixt värde på parametern  $\beta > 0$  att

$$p_\beta(\mathbf{x}) = P(\mathbf{X} = \mathbf{x}) = \frac{e^{-\beta E(\mathbf{x})}}{Z}$$

där  $Z$  är en normeringskonstant dvs

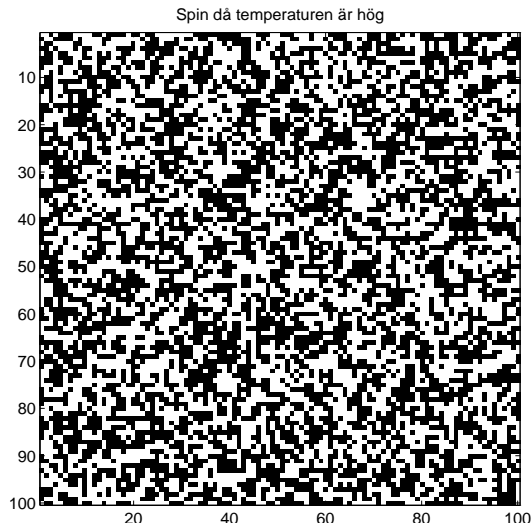
$$Z = \sum_{\mathbf{y} \in \Omega} e^{-\beta E(\mathbf{y})}$$

Den som kan lite statistisk mekanik ser att  $Z$  är "tillståndssumman" och att  $p_\beta(\mathbf{x})$  utgör den s k kanoniska fördelningen eller Gibbsfördelningen. Parametern  $\beta$  svarar fysikaliskt mot  $1/\text{temperaturen}$ . Systemet antas alltså vara placerat i ett stort omgivande system ("värmebad") med temperatur  $1/\beta$  som det utbyter energi med.

Det är uppenbart att tillståndssumman  $Z$  är extremt besvärlig att beräkna, medan  $E(\mathbf{x})$  är förhållandevis enkel.  $Z$  kan ju som i exemplet ovan kunna bestå av  $2^{10000}$  termer!

Man vill gärna simulera  $p_\beta$ -fördelningen och helst för ett tätt gitter av  $\beta$ -värden eftersom en hel del termodynamiskt intressanta storheter som t ex specifikt





Figur 1: Spin i Ising-modell då temperaturen är hög.

värme kan beräknas ur  $p_\beta$ -fördelningen. Ett typiskt utseende av spinnen på gittret är att för låga  $\beta$ -värden (svarande mot höga temperaturer) är de olika gitterpunkternas spinn rätt oberoende av varandra (se figur 1), medan för höga  $\beta$ -värden (dvs låga temperaturer) har stora regioner av  $S$  samma spinnriktning (se figur 2). Vid en viss kritisk temperatur sker ett dramatiskt omslag av uppträdandet – över den temperaturen är spinnen kaotiska, men under den temperaturen får stora regioner av  $S$  samma magnetisering, s k fasövergångar. Det finns alltså stort intresse av att kunna simulera uppträdandet av dessa spinn-system.

Hur kan man nu simulera sådana spinn, dvs hur får man ett utfall från  $p_\beta$ -fördelningen? Idén är att för fixt  $\beta$  konstruera en Markovkedja

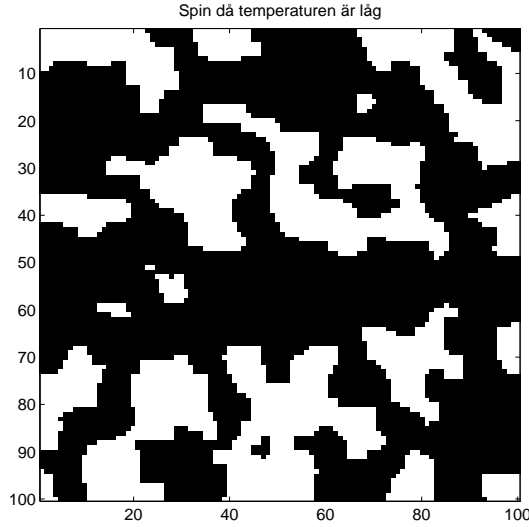
$$\mathbf{X} = (X_{i,j}; (i,j) \in S)$$

som har  $p_\beta$  till asymptotisk fördelning.

Vad man gör är att utifrån en godtycklig spinn-konfiguration successivt uppdatera de  $nm$  gitterpunkternas spinn med fördelningen för just det spinnets betingat av alla de övrigas aktuella värde!

Om vi låter  $\mathbf{x}_{-(i,j)}$  beteckna spinnen i alla punkter i gittret  $S$  utom i punkten  $(i,j)$  så inser man att fördelningen för spinnets i  $(i,j)$  givet alla de övriga blir

$$\begin{aligned} P(X_{i,j} = 1 | \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) &= (\text{Jfr } P(A|B) = P(A \cap B)/P(B)) = \\ &= \frac{P(X_{i,j} = 1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)})}{P(\mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)})} = (\text{Jfr } P(A) = P(A \cap B) + P(A \cap B^*)) = \\ &= \frac{P(X_{i,j} = 1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)})}{P(X_{i,j} = 1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) + P(X_{i,j} = -1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)})} = \end{aligned}$$



Figur 2: Spin i Ising-modell då temperaturen är låg.

$$\begin{aligned}
&= \frac{\exp(-\beta E(X_{i,j} = 1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}))}{\exp(-\beta E(X_{i,j} = 1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)})) + \exp(-\beta E(X_{i,j} = -1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}))} = \\
&\text{(förkorta med sista termen i nämnaren)} \\
&= \frac{\exp(-\beta (E(X_{i,j} = 1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) - E(X_{i,j} = -1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)})))}{\exp(-\beta (E(X_{i,j} = 1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) - E(X_{i,j} = -1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}))) + 1}.
\end{aligned}$$

Men uttrycket i exp-funktionerna blir

$$\begin{aligned}
&E(X_{i,j} = +1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) - E(X_{i,j} = -1; \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) = \\
&= -J \sum_{(i',j') \in N(i,j)} (+1)x_{i',j'} + J \sum_{(i',j') \in N(i,j)} (-1)x_{i',j'} = -2J \sum_{(i',j') \in N(i,j)} x_{i',j'}
\end{aligned}$$

och detta sista blir  $-2J(N_+(i,j) - N_-(i,j))$  där  $N_+(i,j)$  = antalet spinn upp i  $N(i,j)$  och  $N_-(i,j)$  = antalet spinn ner i  $N(i,j)$ , dvs i omgivningen till  $(i,j)$ .

Alltså får vi

$$P(X_{i,j} = +1 | \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) = \frac{\exp(2\beta J(N_+(i,j) - N_-(i,j)))}{\exp(2\beta J(N_+(i,j) - N_-(i,j))) + 1}$$

och

$$\begin{aligned}
P(X_{i,j} = -1 | \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) &= 1 - P(X_{i,j} = +1 | \mathbf{X}_{-(i,j)} = \mathbf{x}_{-(i,j)}) = \\
&= \frac{1}{\exp(2\beta J(N_+(i,j) - N_-(i,j))) + 1}.
\end{aligned}$$

Betingade fördelningen för  $X_{i,j}$  givet alla övriga spinn beror alltså enkelt av bara antalet spinn upp respektive ner bland närmaste grannarna, dvs de som de primärt växelverkar med.

Vi kan då “uppdatera” en spinn-punkt i taget där sannolikheten för spinn= $+1$  resp spinn= $-1$  bara beror på ovanstående sätt av spinnen bland de närmsta grannarna. Vi stegar alltså runt gittret en gitterpunkt i taget och uppdaterar den med alla andra fixa. När vi så vandrat runt gittret ett helt varv har vi fått en ny observation av  $\mathbf{X}$ . Detta utgör då en irreducibel aperiodisk Markovkedja som har  $p_\beta$  till stationär fördelning och vi kan således på detta sätt simulera  $p_\beta$ .

Detta att uppdatera en komponent i taget i  $\mathbf{X}$  med den betingade fördelningen givet de övriga komponenterna är essensen i den s k Gibbs-algoritmen eller Gibbs-samplern.