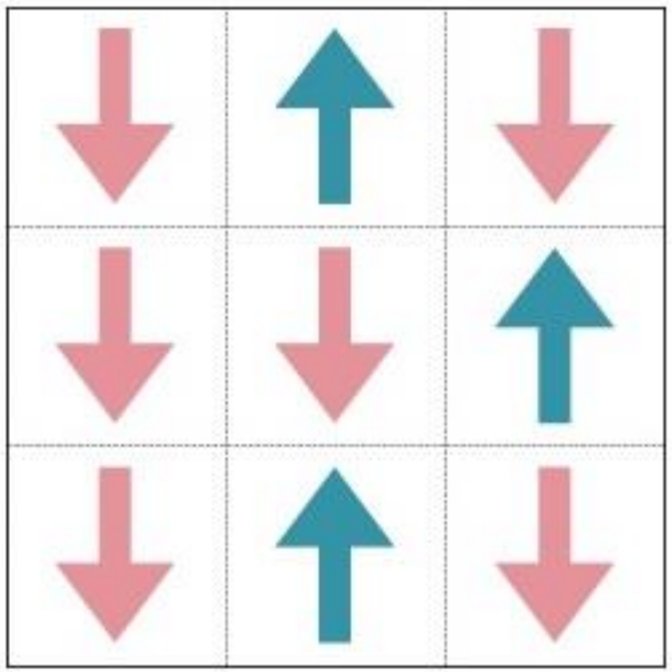
Monte Carlo Yöntemleri
ve
Ising Modeli

Arif Bayırlı

Boğaziçi Ünv. Fizik Bölümü

16. (mini) Fizik Haftası (22-24 Ocak 2018)

Ising Modeli



- 2 boyutlu spin dizisi – durum uzayı {-1 (aşağı),1 (yukarı) }
- Toplam Enerji:

$$E = -\left(\sum_{<i,j>} J\, \sigma_i \sigma_j + \sum_j B \sigma_j \right)$$

  - $< i, j >$: komşuluk ilişkisi
  - J = +/- 1 Ferromanyetik/Anti-ferromanyetik
  - B: Manyetik alan
- Sadece spin etkileşimlerini göz önüne alacağız (B = 0)
- Her bir durumun olasılığı:

$$\pi(\sigma) = \frac{e^{-\beta E(\sigma)}}{Z}$$

  - Z: Bölüşüm fonksiyonu (Partition Function)
- İlgilendiğimiz problemler:
  - Ortalama mıknatıslanma: $\frac{1}{N} \sum_j \sigma_j$
  - Ortalama enerji: $\bar{E}$
- Z'i hesaplamak için durum uzayındaki her bir noktayı gezmeliyiz
  - Bizim örneğimizde N = 9 >>> $2^9 = 512$ olasılık
  - Büyük N için, örneğin N = 1000 için 2^{1000} (astronomik astronomik)

Oyuncak Model: Yazı gelene kadar at!

Problem:

P(n): Elimdeki parayı artarda attığımda üzerinde yazı yüzünün ilk defa (n+1) denemede gelme olasılığı (n= 0, 1, 2, 3, ...)

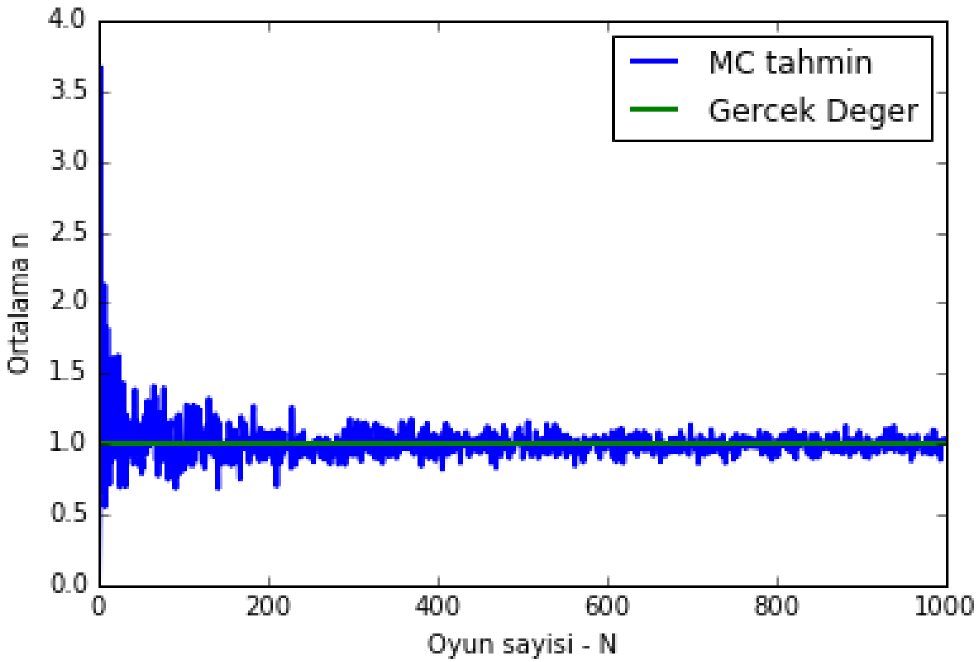
$$P(Y) = \mathrm{p} \; ; \; P(T) = 1 - \mathrm{p} = \mathrm{q}$$

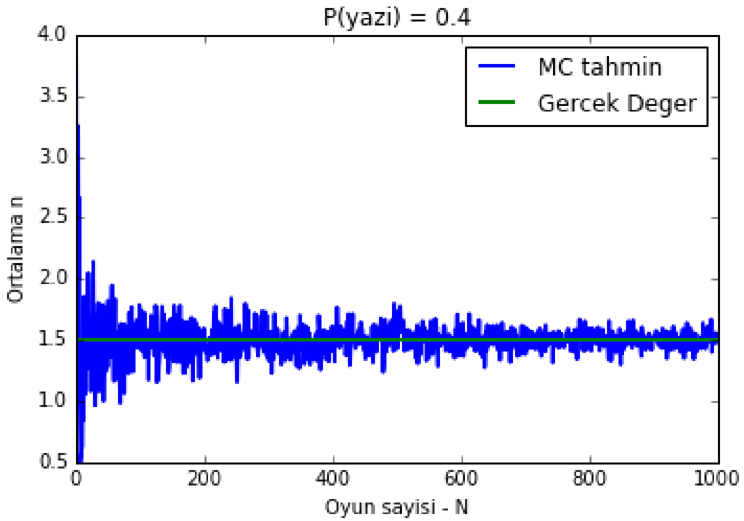
```
N = 5
para = ['H', 'T']
prob = [0.5, 0.5]
denemeler = []
for i in range(N):
        paralar = []
        deneme = 0
        while(True):
            u = np.random.choice(para, p = prob)
            paralar.append(u)
            if(u == 'H'):
                print("Paralar: ", paralar)
                print('# Kacinci denemede yazi geldi: ', deneme+1)
                denemeler.append(deneme)
                break
            else:
                deneme += 1
```

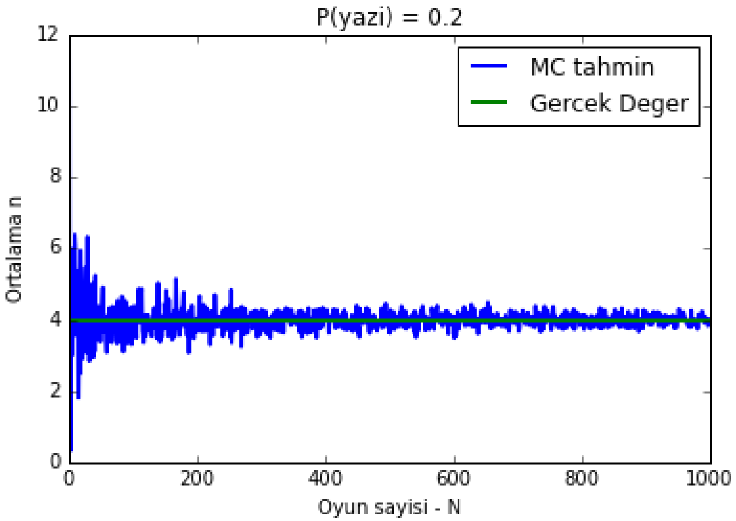
```
('Paralar: ', ['T', 'H'])
('# Kacinci denemede yazi geldi: ', 2)
('Paralar: ', ['T', 'T', 'T', 'T', 'T', 'H'])
('# Kacinci denemede yazi geldi: ', 6)
('Paralar: ', ['H'])
('# Kacinci denemede yazi geldi: ', 1)
('Paralar: ', ['T', 'T', 'T', 'H'])
('# Kacinci denemede yazi geldi: ', 4)
('Paralar: ', ['H'])
('# Kacinci denemede yazi geldi: ', 1)
```

N = 1000 kez oynarsak ortalama kaç olur?

```python
N = 1000
mean = []
para = ['H', 'T']
prob = [0.5, 0.5]
for i in range(N):
    denemeler = []
    for j in range(i+1):
        paralar = []
        deneme = 0
        while(True):
            u = np.random.choice(para, p = prob)
            paralar.append(u)
            if(u == 'H'):
                denemeler.append(deneme)
                break
            else:
                deneme += 1
    mean.append(np.mean(denemeler))
```







Oyuncak Model: Yazı gelene kadar at!

- Problem:

  P(n): Elimdeki parayı artarda attığımda üzerinde yazı yüzünün ilk defa (n+1) denemede gelme olasılığı (n= 0, 1, 2, 3, ...)

$$P(Y) = \mathrm{p}\ ;\ \ P(T) = 1 - \mathrm{p} = \mathrm{q}$$

$$P(n) = q^n p \text{ (Geometrik dağılım)}$$

- $P(n)$ bir olasılık dağılımı mı?

$$\sum_n P(n) = \sum_n q^n p = p \sum_n q^n = p\,\frac{1}{1-q} = 1$$

- Ortalama n değeri ve varyans:
  - Generating Function:

$$f(z) = \sum_n^\infty P(n) z^n = \sum_n^\infty = q^n p z^n = p \sum_n^\infty (qz)^n = \frac{p}{1-qz}$$

$$\bar{n} = \frac{df}{dz}\Big|_{z=1} = \textstyle\sum_n^\infty nP(n)z^{n-1}\Big|_{z=1} = \sum_n^\infty nP(n) = \frac{pq}{(1-qz)^2} = \frac{q}{p}$$

$$\text{Varyans } \sigma^2 = \frac{q}{p^2}$$

Monte Carlo

- İki problemi çözmek için:
  - Problem 1: Verilen bir P(x) olasılık dağılımından örnekler $\{x^{(r)}\}_{r=1}^{R}$ çekmek
  - Problem 2: Fonksiyonların bu dağılım altındaki beklenen değerlerini hesaplamak

$$\Phi = <\varphi(\boldsymbol{x}) > \equiv \int d^N\boldsymbol{x}\, P(\boldsymbol{x})\varphi(\boldsymbol{x})$$

  - $P(\boldsymbol{x})$: Hedef dağılım (target density)
  - $\boldsymbol{x}$: N boyutlu vektör
- İlgilenebileceğimiz örnek $\varphi(\boldsymbol{x})$fonksiyonları:

$$\varphi_1(\boldsymbol{x}) = x, \qquad \varphi_2(\boldsymbol{x}) = x^2$$

$$\overline{\boldsymbol{x}} = \Phi_1 \quad , \qquad \overline{\boldsymbol{x}^2} = \Phi_2; \quad \overline{(\boldsymbol{x} - \overline{\boldsymbol{x}})^2} = \Phi_2 - \Phi_1$$

- Eğer $P(\boldsymbol{x})$'den örnekler üretebilirsek, ikinci problemi çözebiliriz:

$$\hat{\Phi} = \frac{1}{R}\sum_{r} \varphi(\boldsymbol{x}^{(r)})$$

Monte Carlo

- Neden P(x)’den örnekleme yapmak zor?
  - P(x) fonksiyonunu belirli bir katsayıya kadar hesaplayabiliyoruz.

$$P(\boldsymbol{x}) = P^*(\boldsymbol{x})/Z$$

  - Eğer $P^*(x)'$ı hesaplayabiliyorsak, neden bağımsız örnekler çekmek zor?

    Genellikle Z fonksiyonunu (normalizasyon sabiti) bilmiyoruz.

$$Z = \int d^N\boldsymbol{x}\, P^*(\boldsymbol{x})$$

    Z’yi bilsek dahi yüksek boyutlarda her noktada P’yi hesaplamadan örnek çekmek güç (nokta sayısı çok fazla)

- Monte Carlo yöntemleri rastgele sayılar ve çeşitli teknikler kullanarak belirli bir dağılımdan bağımsız örnekler üretmeyi ve sonrasında bu örnekleri kullanarak örneklerin fonksiyonlarının beklenen değerlerini hesaplamayı sağlıyor.

  Eğer $P(\boldsymbol{x})$’den örnekler üretebilirsek, ikinci problemi çözebiliriz:

$$\hat{\Phi} = \frac{1}{R}\sum_r \varphi(\boldsymbol{x}^{(r)})$$

Monte Carlo yöntemleri neden çalışıyor?

- Yansız kestirici (unbiased estimator):

$$\mathsf{E}_\pi \varphi(x) \approx \frac{\varphi(x^{(1)}) + \cdots + \varphi(x^{(N)})}{N} \equiv \bar{E}_{\varphi,N} \, .$$

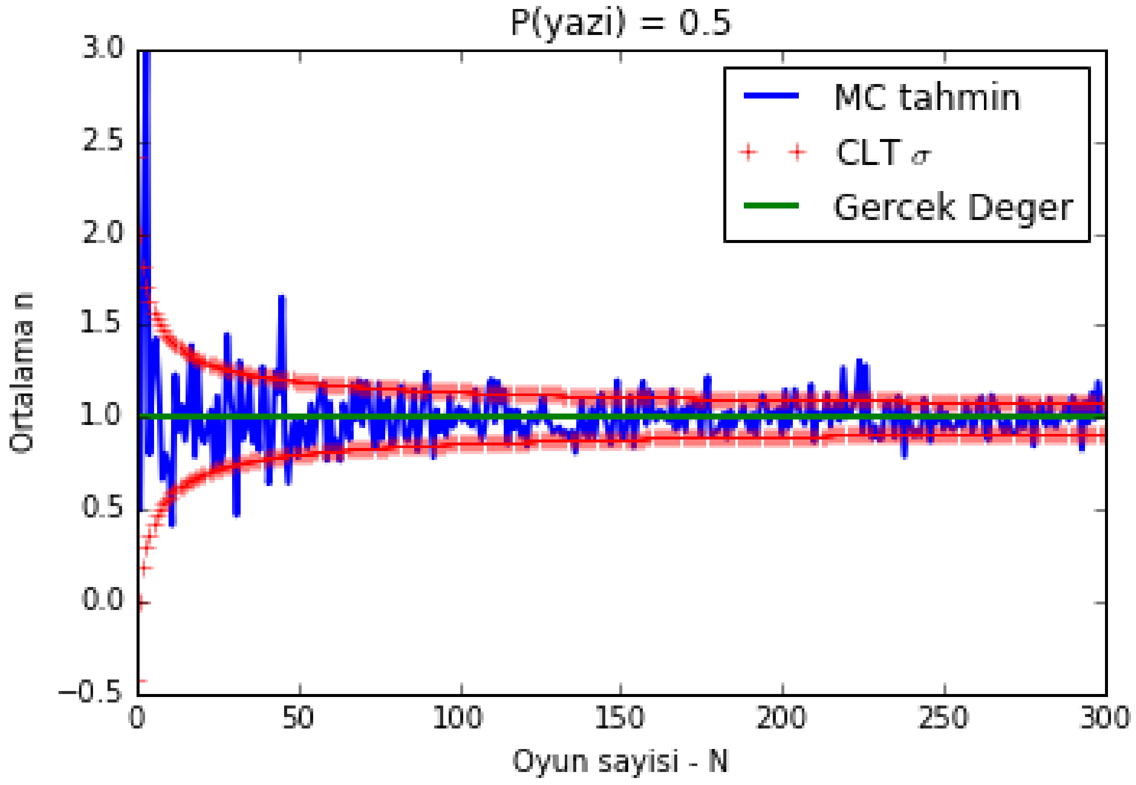
$$\bar{E}_{\varphi,N} \to \mu$$

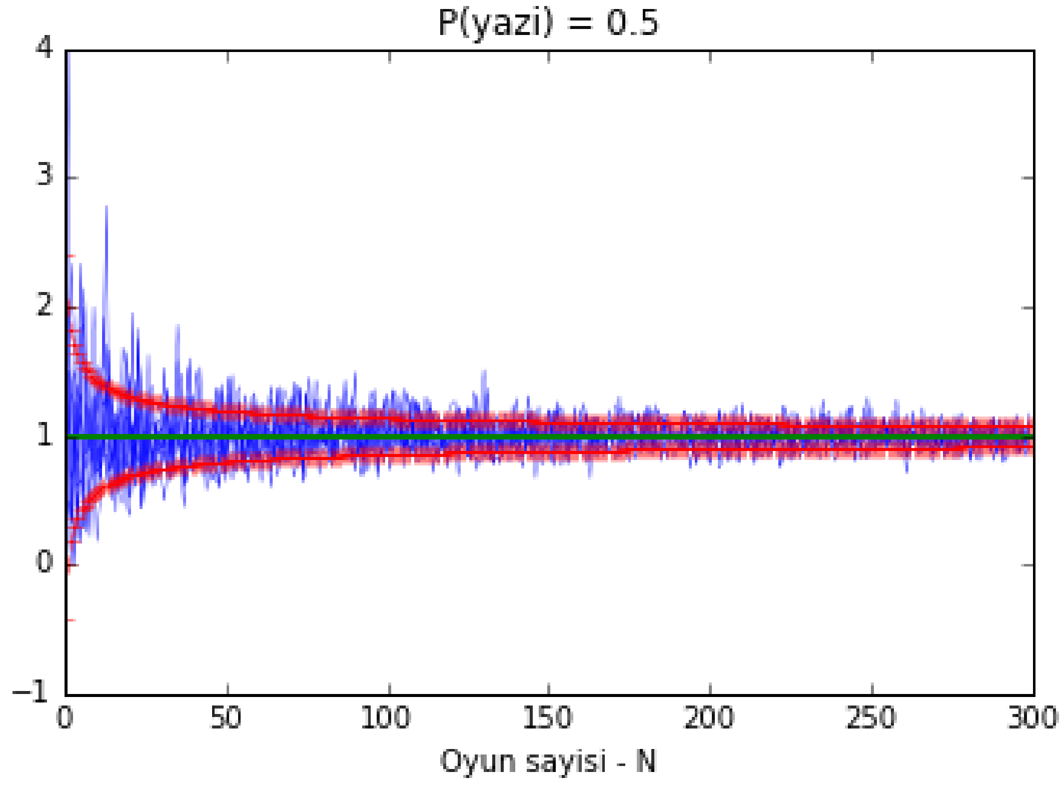
- Sonlu örnek ile kestirim yaptığımızdan dalgalanmalar (fluctuations) kaçınılmaz!
- Merkezi Limit Teoremi (Central Limit Theorem – CLT): Büyük N'ler için (N→ ∞) dalgalanmalar normal dağılıma sahip

$$\bar{E}_{\varphi,N} \sim \mathcal{N}\left(\bar{E}_{\varphi,N} \middle| \mu, \sigma^2/N\right)$$

- Eğer $\pi(x)$ gibi bir dağılım fonksiyonundan birbirinin aynısı ve bağımsız (iid) örnekler çekebilirsek Monte Carlo kestirimcisi ile:
  - Monte Carlo kestirimcisi dalgalanmalı da olsa gerçek değerin yansız kestirimini veriyor. $\mu = \mathsf{E}_\pi \varphi(x)$
  - Dalgalanmalar $\sqrt{N}$ ileölçekleniyor.
  - Yöntemin yakınsaması x'in boyutundan tamamen bağımsız!

Yazı-Tura oyununa geri dönelim





MC yöntemi: Reddetme Örneklemesi

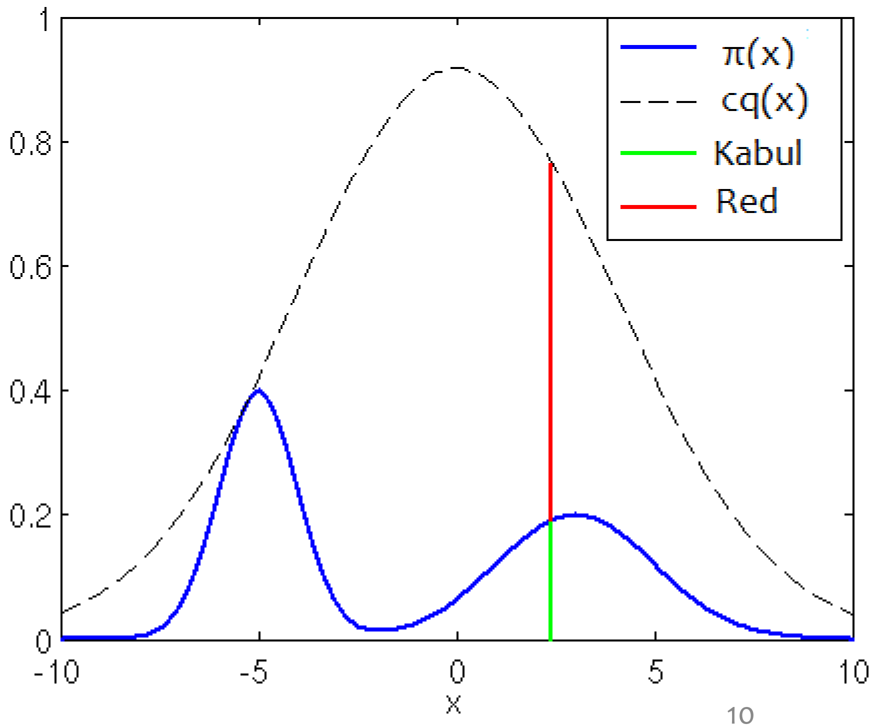
- Elimizde örneklememiz zor olan $\pi(x)$ ve örnekleyebildiğimiz bir $q(x)$ dağılım fonksiyonları olsun.
- Temel fikir: $q(x)$'den örnekler çekerek belirli bir reddetme/kabul etme kriteri uygulayarak kabul edilen örneklerin sonuçta $\pi(x)$'e göre dağılmış olmalarını sağlamak
- Gerekli şart: $q(x)$'in $\pi(x)$'i örtmesi (envelope) gerekiyor:

$$c\,q(x) > \pi(x)\,;\;\; c > 1$$

- Örnekleri kabul şartı:

$$u \sim Uniform[0,1]$$

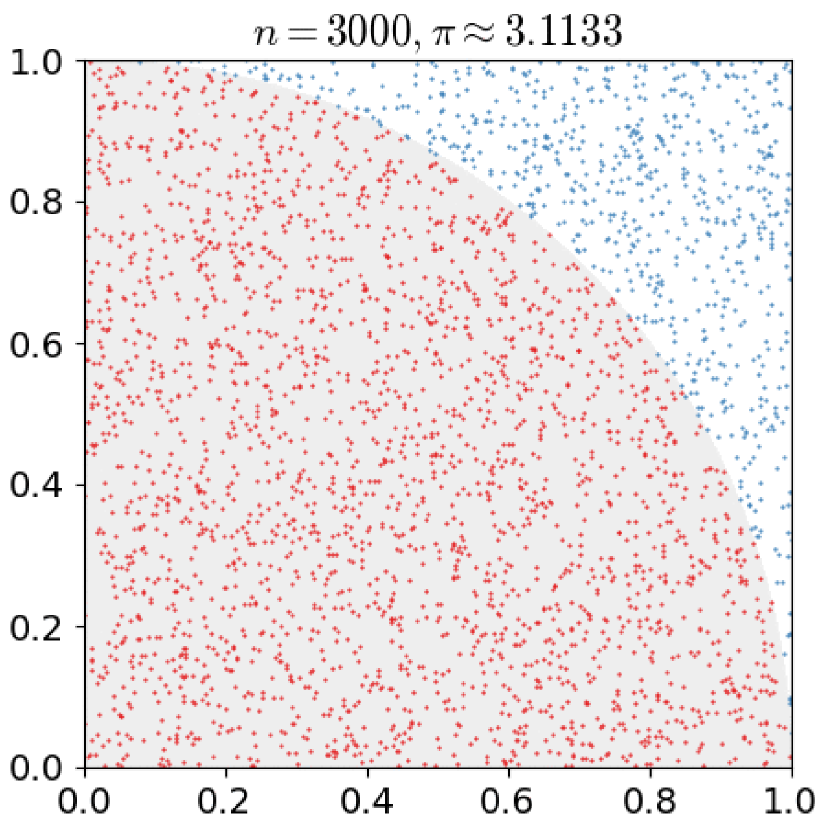
$$\frac{\pi(x)}{c\,q(x)} > u$$



MC yöntemi: Reddetme Örneklemesi

- Farklı bir reddetme kriteri: Geometrik
  - Daire içinden düzgün dağılmış noktalar üretmek
    - $x, y \sim Uniform[0,1]$
    - Eğer $\sqrt{x^2 + y^2} < R \rightarrow$ Kabul Et; değilse Reddet
    - π'yi hesaplama yöntemi!

$$\frac{n_{hit}}{n_{toplam}} = \frac{\pi 1^2/4}{1^2} = \frac{\pi}{4}$$



MC yöntemi: Metropolis-Hastings

<<Metropolis-Hastings yöntemi Markov Zincirleri'nin durağan dağılımları kullanılarak örnekler üretmeye dayanıyor.>>

Peki Markov Zinciri nedir?

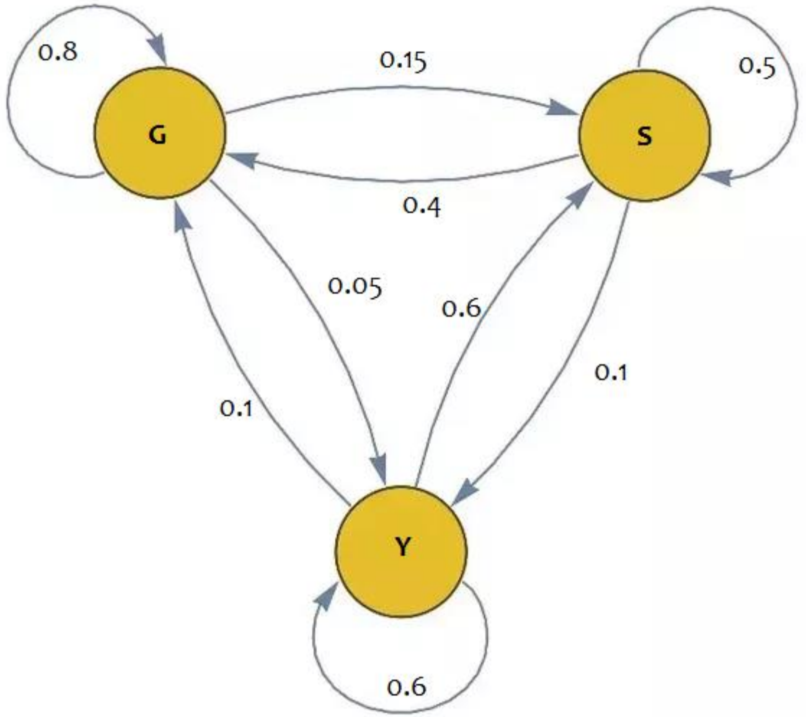
Markov Zinciri

- Belirli bir durum uzayında, ardışık olarak durumdan duruma rastgele geçişlerin olduğu süreçler
  $$x^{(0)} \to x^{(1)} \to x^{(2)} \to \cdots x^{(t)} \to \cdots$$
- Markov Zincir'lerini tanımlayan 3 öğe:
  - Durum uzayı x; zincirin alabileceği tüm değerleri içeren küme
  - Geçiş operatörü (transition operatör) $p(x^{t+1}|x^t), x^t$ durumundan x^{t+1} durumuna geçiş olasılığını belirten fonksiyon
  - ilk başta $(t = 0)$ sistemin hangi durumlarda hangi olasılıkla olduğunu belirten başlangıç dağılımı $\pi(0)$
- Markov zinciri $\pi(0)'$daki dağılımdan bir durumdan başlayıp zamanla $p\left(x^{(t+1)}\middle|x^{(t)}\right)$'ya göre evrilir
- Belirli bir zamandaki dağılım yalnızca bir önceki durum dağılımına bağlıdır:
  $$p\left(x^{(t)}\middle|x^{(t-1)}, x^{(t-2)}, \ldots, x^{0}\right) = p\left(x^{(t)}\middle|x^{(t-1)}\right)$$
- Belirli koşullar altında Markov zinciri $t \to \infty$ için durağan (stationary) dağılıma sahip!
  $$p\left(x^{(t+1)}\middle|x^{(t)}\right) = p\left(x^{(t)}\middle|x^{(t-1)}\right)$$

Markov Zinciri – Bir Örnek

- Üç farklı hava koşulu – {güneşli, sisli, yağmurlu}
- Eğer hava bugün güneşliyse:
  - $P\left(X^{haftaya} = güneşli \middle| X^{bugün} = güneşli\right) = 0.8$
  - $P\left(X^{haftaya} = sisli \middle| X^{bugün} = güneşli\right) = 0.15$
  - $P\left(X^{haftaya} = yağmurlu \middle| X^{bugün} = güneşli\right) = 0.05$
- Eğer hava bugün sisliyse:
  - $P\left(X^{haftaya} = güneşli \middle| X^{bugün} = sisli\right) = 0.4$
  - $P\left(X^{haftaya} = sisli \middle| X^{bugün} = sisli\right) = 0.5$
  - $P\left(X^{haftaya} = yağmurlu \middle| X^{bugün} = sisli\right) = 0.1$
- Eğer hava bugün yağmurluysa:
  - $P\left(X^{haftaya} = güneşli \middle| X^{bugün} = yağmurlu\right) = 0.1$
  - $P\left(X^{haftaya} = sisli \middle| X^{bugün} = yağmurlu\right) = 0.3$
  - $P\left(X^{haftaya} = yağmurlu \middle| X^{bugün} = yağmurlu\right) = 0.6$
- Tüm bu olasılıklar 3x3 geçiş matrisi ile modellenebilir:

$$\begin{bmatrix} 0.8 & 0.15 & 0.05 \\ 0.4 & 0.5 & 0.1 \\ 0.1 & 0.3 & 0.6 \end{bmatrix}$$



Markov Zinciri – Bir Örnek

- Geçiş kuralı: P = $\begin{bmatrix} 0.8 & 0.15 & 0.05 \\ 0.4 & 0.5 & 0.1 \\ 0.1 & 0.3 & 0.6 \end{bmatrix}$

- Eğer Markov zincirini $x = [0 \quad 0 \quad 1]$ durumundan yani 1 olasılıkla yağmurlu durumdan başlatıp evriltirsek

$$\pi(0) = [0 \quad 0 \quad 1]$$

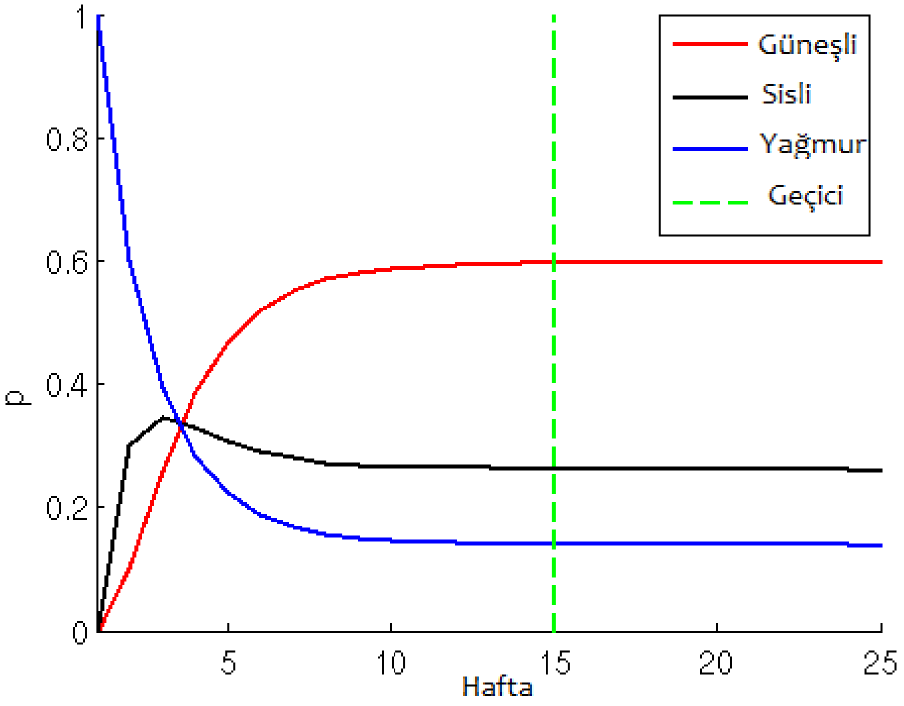
$P\left(x^{1.hafta}\right) = \pi(0)$*P = $[0.1 \quad 0.3 \quad 0.6]$

$P\left(x^{2.hafta}\right) = P\left(x^{1.hafta}\right)$*P = $\pi(0)*P^2$= [0.26, 0.35, 0.39]

…

$P\left(x^{24.hafta}\right)$ = $\pi(0)*P^{24}$ = [0.59, 0.26, 0.15]

Durağan Dağılım



Markov Zinciri Monte Carlo (MCMC)

Temel Fikir: Durağan dağılımı örnek çekmek istediğimiz hedef dağılım olan öyle bir Markov Zinciri tasarlayalım ki, bu Markov Zinciri simule ederek durağan dağılıma ulaştıktan sonra çektiğimiz bir örnek, peşinde olduğumuz hedef dağılımdan olsun!

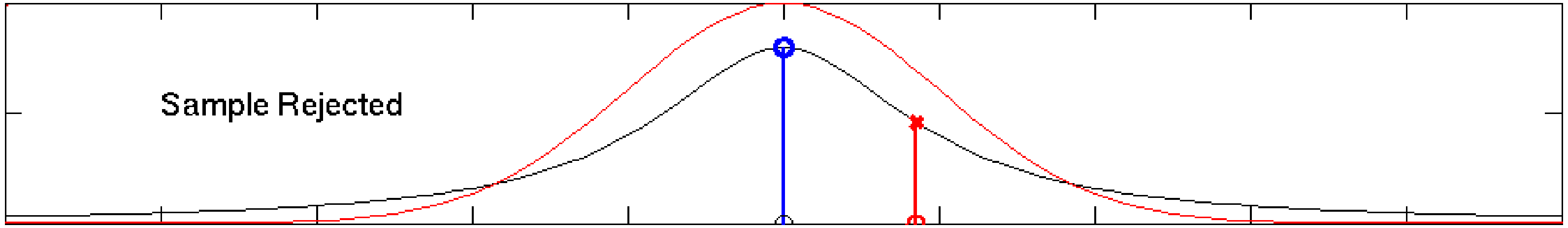
İşimizi görecek Markov Zincirini (geçiş operatörünü) tasarlamanın yöntemi: Metropolis-Hastings algoritması

Metropolis Örnekleyicisi

- Başlangıç olarak $\mathrm{x}(0) \sim \pi(0)$ ile başlayıp bir önerme (proposal) fonksiyonu $q(x^t|x^{t-1})$'dan x^* örneği çekiyoruz. $q(x^t|x^{t-1})$ Markov zinciri geçiş operatörü gibi davranıyor.
- Üretilen x^* örneğinin kabul edilip edilmeyeceği kontrol ediliyor:

$$\alpha_{kabul} = \min(1, \frac{p(x^*)}{p(x^{t-1})})$$

- Metropolis Örnekleyicisinin geçiş operatörü şu şekilde çalışıyor:
  - $u \sim Uniform[0,1]$ eğer α_{kabul}'den küçükse önerilen x^*'i yeni durum olarak kabul et → $x^t = x^*$
  - Değilse bir önceki durumu koru → $x^t = x^{t-1}$
  - Tekrar et
- Örnek:
  - $p(x) = \frac{1}{1+x^2}$
  - $\pi(0) \sim \mathrm{N}(0,1)$
  - $q(x^t|x^{t-1}) \sim N(x^{t-1},1)$



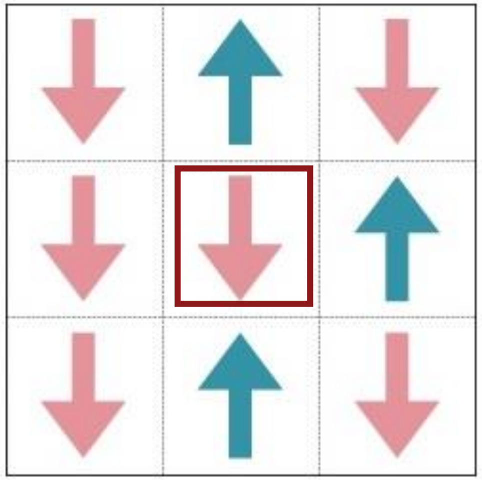
Metropolis-Hastings Örnekleyicisi – simetrik olmayan önerme fonksiyonları

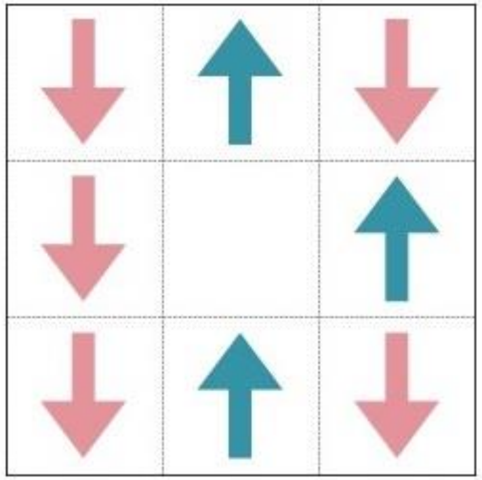
Ising Modeli için Metropolis Örnekleyicisi

- Rastgele bir spin seç ve spin'in yönünü değiştirip değiştirmeyeceğine karar ver.
- Eğer yeni durum düşük enerjiliyse spin'i değiştir, aksi takdirde değiştirme
- Değiştirmeyi kabul etme olasılığı = min(1, $\frac{p(x^*)}{p(x^{t-1})} = \exp(-\beta\Delta E)$)

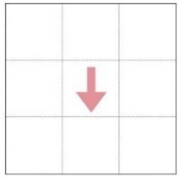
ΔE = 2*$\sigma_i b_i$ (b_i yerel alan)

$$\mathrm{P(kabul;}\ \Delta E, \beta) = \begin{cases} 1 & \Delta E < 0 \\ \exp(-\beta\Delta E) & \Delta E > 0 \end{cases}$$

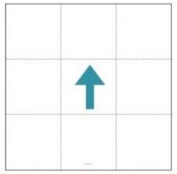




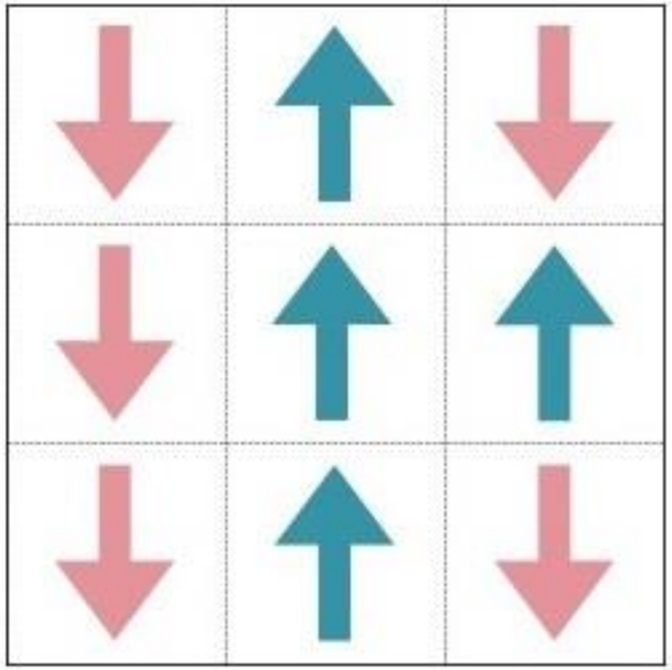
b_i = +3 -1 = 2



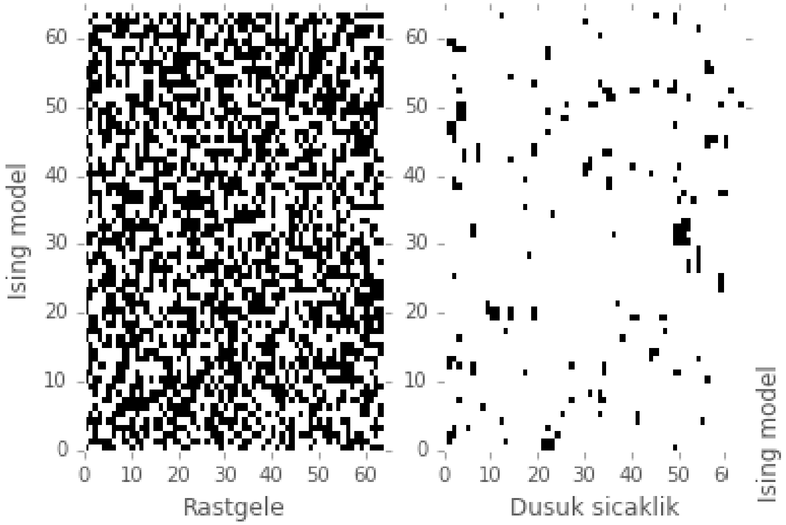
ΔE = -2*2 = -4
Kabul etme!

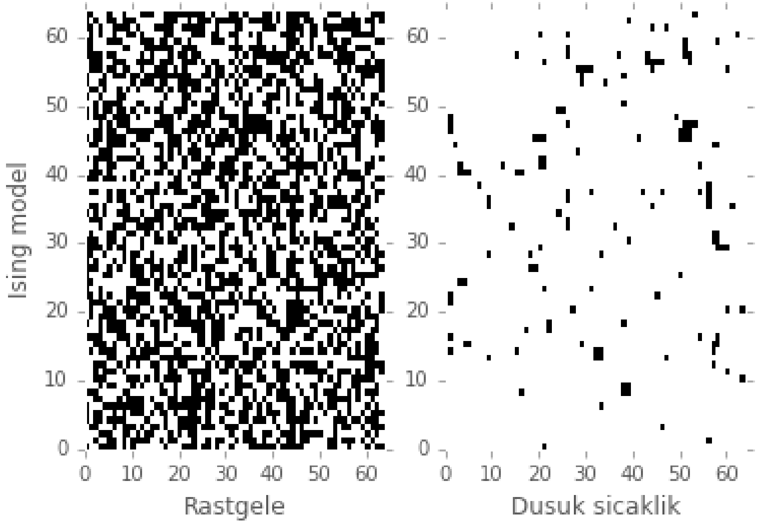


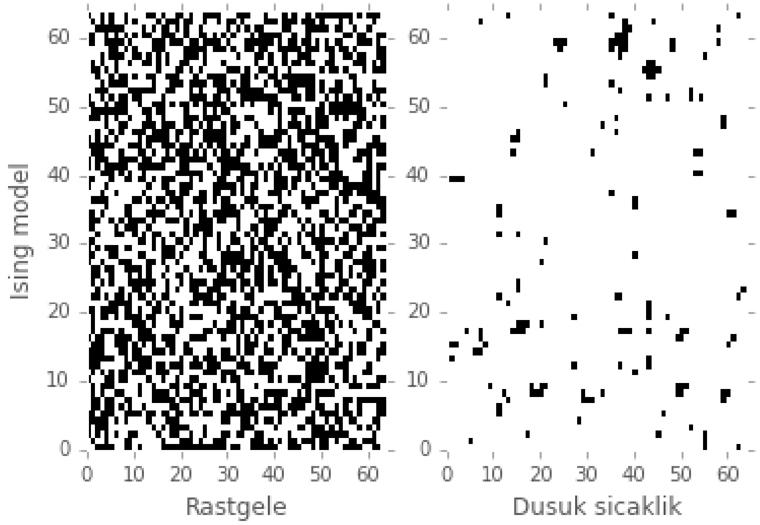
ΔE = 2*2 = 4
Kabul et



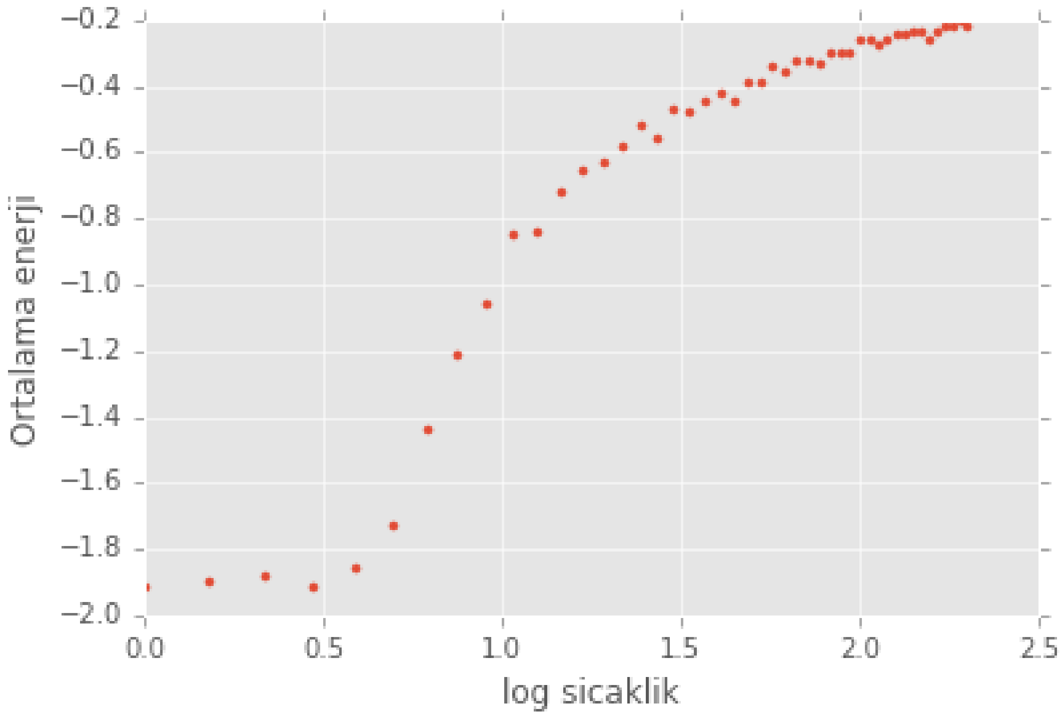
Ising Modelinden Örneklemeler

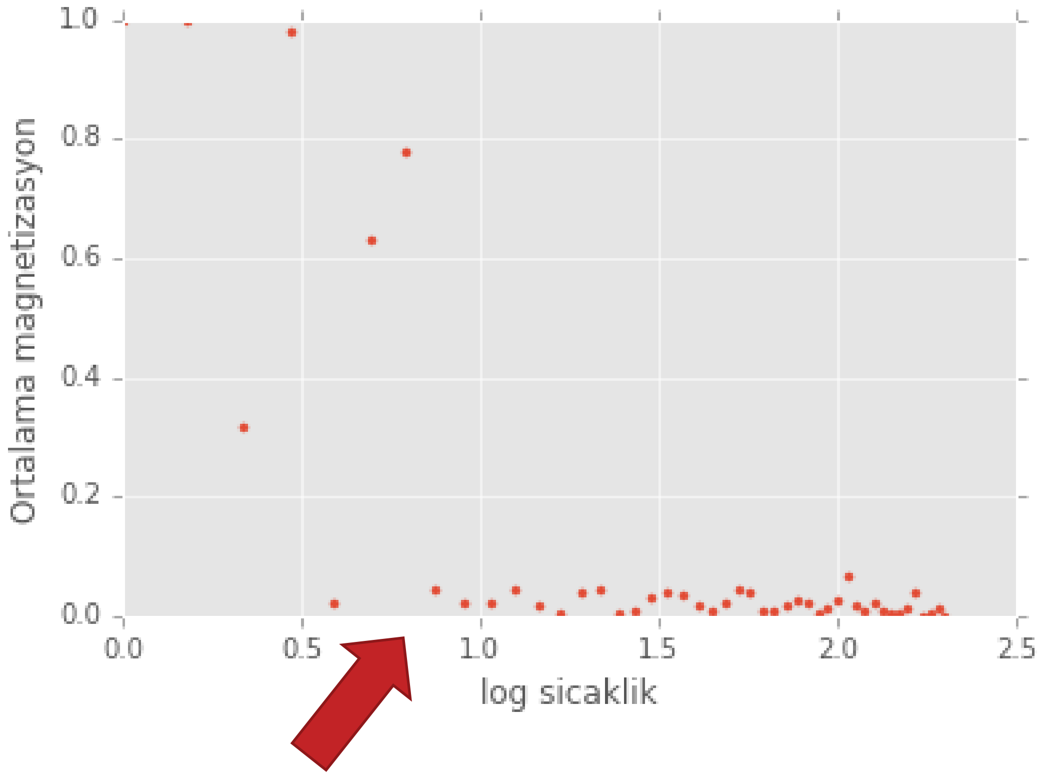






Ortalama Enerji / Magnetizasyon

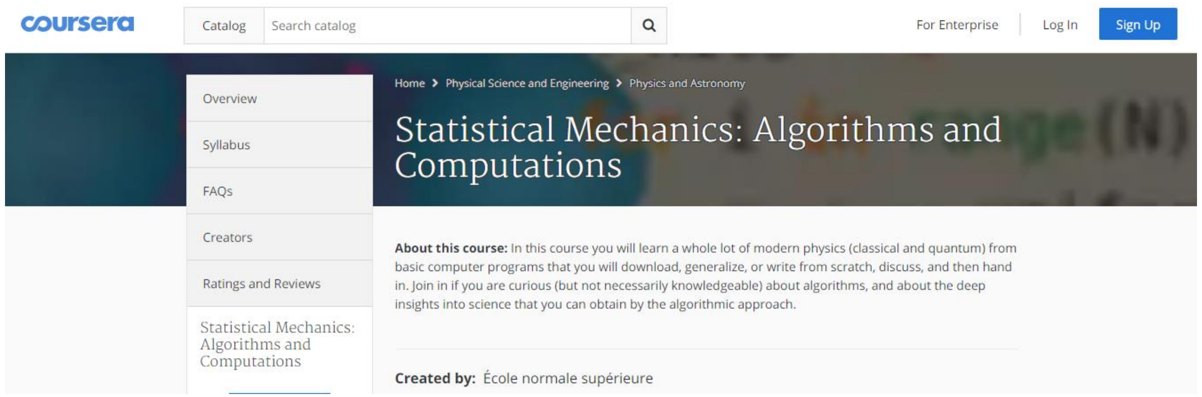


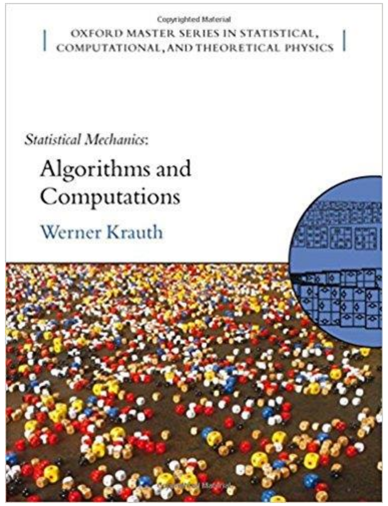


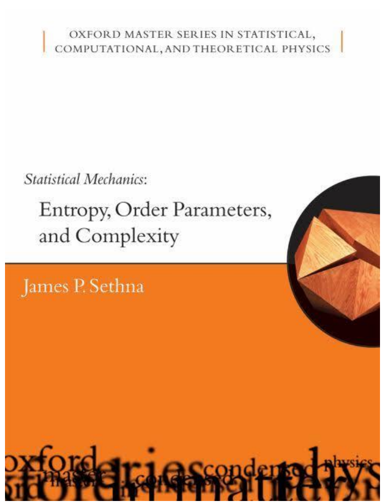
Faz geçişi T ~ 2.2

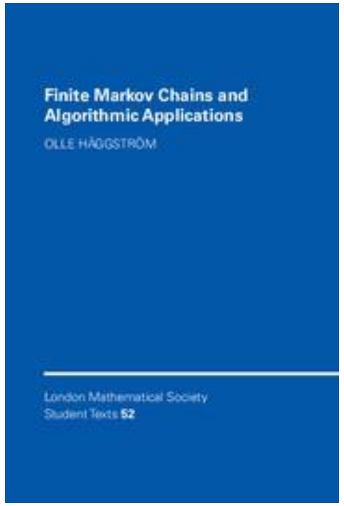
MC diyarından harikalar

- Önem örneklemesi (importance sampling)
- Gibbs örneklemesi
- Simulated Annealing
- Particle Filtering
- Hamilton Monte Carlo









Ders notlarına (yakında) www.aslindafizik.blogspot.com ve www.github.com/abayirli adreslerinden ulaşabilirsiniz.

Teşekkürler!