

## Monte Carlo, Entegraller, MCMC

Fizik, biyoloji ve özellikle makina öğrenimi problemlerinde bazen çok boyutlu bir fonksiyon üzerinden integral almak gerekebiliyor. En basit örnek, mesela bir dağılımın başka bir fonksiyon ile çarpımının beklentisini (expectation) hesaplamak gerektiğinde, ki bu

$$E(f) = \int p(x)f(x)dx$$

entegralidir,  $x \in \mathbb{R}^n$ ,  $p(x)$  dağılım fonksiyonu,  $f(x)$  herhangi bir başka fonksiyon olmak üzere, o zaman tüm  $x$  değerlerini göz önüne alarak (ayrışal bağlamda ya teker teker geçerek, ya da analitik olarak) integral hesabını yapmak gerekecekti.

Fakat  $p(x)$  bir dağılım olduğuna göre, ve bizim gördüğümüz her  $x$  için bir olasılık değeri varsa, bu işi tersine çevirerek,  $p(x)$ 'teki olasılıklara göre belli (az) sayıda  $x$  üretirsek, ve sadece bu  $x$ 'leri integral hesabında kullanırsak yaklaşık olarak gerçek integral hesabına yaklaşmış oluruz.

Bu mantıklı değil mi? Düşünürsek, mesela 10 değeri 0.4 olasılığında ise, 5 değeri 0.1 olasılığında ise, hem sayı, hem olasılığı ile çarpıp yerine “daha fazla 10 değeri üretmek” ve bu değerleri  $f$ 'e geçmek, toplamak, sonra bölmek, vs. yaklaşık olarak aynı kapağı çıkar. Yani

$$E_N = \frac{1}{N} \sum_{i=1}^N f(x^{(i)})$$

üstteki integralin yaklaşık temsilidir,  $x^{(i)}$   $p(x)$  olasılığına göre üretilen sayıları temsil ediyor. Üstteki bağlantının teorik olarak ispatı da var, bu ispatı burada vermeyeceğiz.

İşte Monte Carlo integral hesabının artasında yatan numara budur.

## MCMC

Demek ki Monte Carlo integralinin işlemesi için  $p(x)$ 'den örnekleme yapmak gerekiyor. Şimdi ikinci numaraya gelelim.

Bazen ne yazık ki  $p(x)$ 'den örnekleme yapmak kolay olmuyor. Bu durumlar için  $p(x)$  yerine onu yaklaşık olarak temsil eden bir  $\pi(x)$ 'i elde etmekle uğraşıyoruz. Bu  $\pi(x)$  ise bir Markov Zincirinin (Markov Chain -yine MC harfleri!-) duran dağılımı olarak hayal ediliyor.

Markov Zinciri teorisinde bir geçiş matrisi, yani Markov Zincirinin kendisi verilir, ve duran dağılımın hesaplanması istenir. MCMC problemlerinde ise, yani Monte Carlo integrali için Markov Zinciri kullanıldığı durumlarda elimizde bir  $\pi(x)$  dağılımı vardır ve bir Markov Zinciri oluşturmamız gerekir. Nihai dağılımı biliriz, ve bu dağılıma “giden” geçişleri üretiriz. Bu geçişleri öyle ayarlayabiliriz ki üretilen rasgele sayılar hedef dağılımından geliyormuş gibi olur.

Geçişleri üretmek için literatürde bir çok teknik vardır. Önemli Örnekleme (Importance Sampling)

tance Sampling), Ornekleme ve Oneme Gore Tekrar Ornekleme (Sampling Importance Resampling), Metropolis-Hastings, Gibbs Orneklemesi gibi teknikleri vardır, ve detaylari degisik olsa da hepsi de MCMC kategorisine girer, ve yapmaya calistiklari  $\pi(x)$ 'e giderken bir sekilde bir gecisleri, zinciri ortaya cikartmak ve bu gecisleri integral hesabinda kullanmaktır.

Ustteki tekniklerden en yaygin kullanilani Metropolis-Hastings algoritmasidir.

Not: Bu alandaki makalelerde bir dagilimin “belli bir carpimsal sabite kadar” bilindigi (known up to a multiplicative constant) soylenir. Bu soz aslinda su anlama gelir. Mesela ayriksal bir dagilimimiz var, ama bu dagilimin kendisini, su halini biliyoruz

[ 4.3 2. 8.4 8.7 1.8]

Bu bir dagilim degil, cunku ogelerin toplami 1 degil. Onu bir dagilim haline cevirmek icin, tum ogeleri toplamak ve bu vektördeki tum sayilari bu toplam ile bolmek gerekir. Toplam 25.2, bolersek

[ 0.17063492 0.07936508 0.33333333 0.3452381 0.07142857]

Ilk vektor “belli bir carpimsal sabite kadar” bilinen dagilim, carpimsal sabit 25.2. Esas dagilim ikinci vektor.

Peki niye bu sozu soyleyenler toplami hesaplayip gercek dagilimi hesaplamiyorlar? Sebep performans. Bazen ayriksal dagilim o kadar yuksek boyutlu, fazla oge iceren bir halde oluyor ki, performans acisindan bu basit toplam hesabini yapmak bile cok pahali oluyor. Iste MCMC metotlarinin bir guzel tarafi daha burada, dagilimin kendisi olmasa bile belli bir carpimsal sabite kadar bilinen versiyonlari ile gayet rahat bir sekilde isliyorlar.

Kaynaklar

Algorithmic Machine Learning, Stephen Marsland