Optimizasyon

Yapay Zekâ ve Veri Dersleri Rehberi · Kısım V: Matematik Temeli

Amaç ve kısıt | Doğrusal programlama | Konvekslik | Gradyan yöntemleri | YZ'de nerede | Komşu dersler

Sercan KÜLCÜ · Giresun Üniversitesi · Bilgisayar Mühendisliği

© Sercan Külcü. Tüm hakları saklıdır. · Sunumun etkileşimli HTML sürümü: https://sercankulcu.github.io/files/ai_courses/sunumlar/Bolum_33_Optimizasyon.html

GİRİŞ

Ders Hangi Soruyu Sorar?

> **Bir amaç fonksiyonunu en iyi yapan değerler nasıl bulunur?**

Optimizasyon, kısıtlara uyarak bir **amaç fonksiyonunu** en küçük ya da en büyük yapan değerleri bulmayı inceler. Yapay zekâda bir modeli eğitmek, kayıp fonksiyonunu en küçükleyen ağırlıkları aramaktır; bu yüzden her öğrenme algoritmasının içinde bir optimizasyon yöntemi çalışır.

Bir bakışta

YAKLAŞIM	GİRDİ	ÇIKTI
amaç ve kısıtların matematiksel modeli	amaç fonksiyonu ve kısıtlar	en iyi ya da ona yakın çözüm

BAŞARI ÖLÇÜTÜ	MATEMATİK YÜKÜ	VERİ İHTİYACI
amaç değeri, yakınsama hızı	kuramın kendisi	gerekmez

Kısa Tarihçe

1847
Cauchy, denklem sistemlerini çözmek için en dik iniş (gradyan inişi) yöntemini önerir.

1947
Dantzig doğrusal programlama için simpleks yöntemini geliştirir.

1951
Kuhn ve Tucker, kısıtlı problemlerde en iyiliğin koşullarını (KKT) yayımlar.

1957
Bellman'ın Dynamic Programming kitabı yayımlanır.

1984
Karmarkar, doğrusal programlama için polinom zamanlı iç nokta yöntemini tanıtır.

2014
Kingma ve Ba, derin ağların eğitiminde yaygınlaşacak Adam yöntemini önerir.

DERS

Temel Kavramlar

Amaç fonksiyonu
En küçüklenecek maliyet ya da kayıp.

Kısıtlar
Çözümün uyması gereken eşitlik ve eşitsizlikler.

Doğrusal programlama
Doğrusal amaç ve kısıtlar; simpleks yöntemi.

Konvekslik
Her yerel en iyi aynı zamanda global en iyidir.

Gradyan inişi
Eğimin tersi yönünde küçük adımlar atmak.

Stokastik gradyan
Her adımda verinin küçük bir parçasını kullanmak.

Lagrange çarpanları
Kısıtları amaca katarak çözmek; KKT koşulları.

Tam sayılı programlama
Değişkenler tam sayı; çoğu zaman çok zordur.

Tipik İş Akışı



Örnek: Bir sinir ağını eğitmek bir optimizasyon problemidir: değişkenler ağırlıklar, amaç ortalama kayıptır. Problem konveks değildir; bu yüzden küçük gruplarla stokastik gradyan inişi ya da Adam kullanılır ve kayıp eğrisi doğrulama kümesinde izlenir.

DERS

Derste Neler İşlenir?

1. Problem kurma: değişken, amaç, kısıt
2. Doğrusal programlama ve simpleks yöntemi
3. Dualite ve duyarlılık analizi
4. Konveks kümeler ve konveks fonksiyonlar
5. Konveks problemler: QP, SOCP, SDP
6. En iyilik koşulları: Lagrange ve KKT
7. Gradyan inişi ve adım boyu seçimi
8. Newton ve yarı-Newton yöntemleri (BFGS)
9. Kısıtlı optimizasyon ve iç nokta yöntemleri
10. Stokastik gradyan, momentum ve Adam
11. Tam sayılı programlama ve dal-sınır
12. Dinamik programlama

Kazandırdığı beceriler

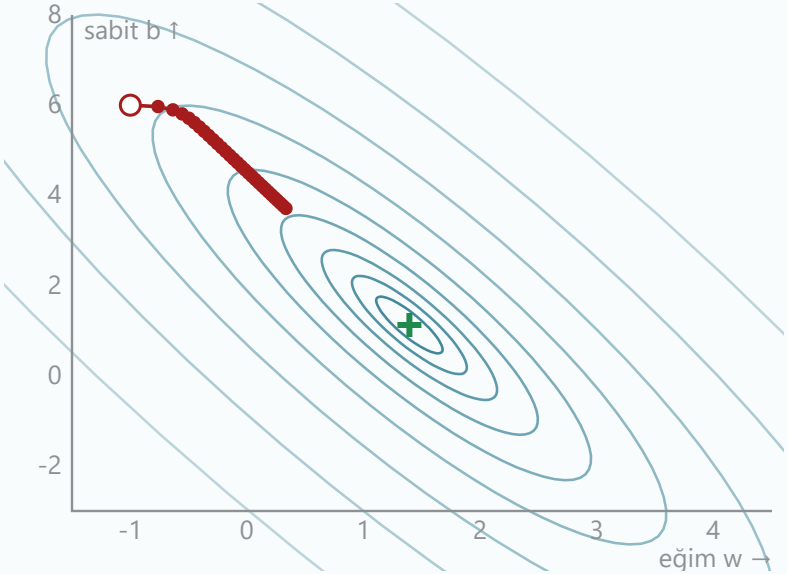
CVXPY ile konveks problem kurmak · PuLP ile doğrusal model yazmak · scipy.optimize kullanmak · konveksliği göstermek · yakınsamayı izlemek ve ayarlamak

Tipik bir dönemin akışıdır; sıra ve kapsam üniversiteden üniversiteye değişir.

DENE

Fiyat Modelini Eğitmek

Ev fiyatını “fiyat = w·alan + b” ile tahmin eden modelin iki ağırlığı, kaybı en aza indiren yere gradyan inişiyle gidiyor. Haritada başlangıca tıklayın, öğrenme hızını değiştirin.



Öğrenme hızı η = **0,05**

Ham alan | Ölçeklenmiş alan

1 adım | Çalıştır | Sıfırla

Adım **30**: kayıp **1,89** (en düşük 0,18)
w = 0,33, b = 3,71

Vadinin basıklığı (koşul sayısı) **26**; kararlı kalmak için η < 0,18. Dar yönde salınıp uzun yönde ağır ilerler.

Öğrenme hızı küçükse model ağır ilerler, büyükse dar vadide sekip ıraksar. Özellikleri ölçeklemek vadiyi yuvarlar; sinir ağlarında normalleştirme bu yüzden eğitimi hızlandırır.

KONUM

Ön Koşullar

Bu dersten önce bilinmesi yararlı olan dersler ve neden gerektikleri.

Ders	Neden gerekir
Doğrusal Cebir	gradyan, Hessian, konvekslik
Analiz (Kalkülüs)	türev, kısmi türev ve Taylor açılımı: gradyanın ve Hessian'ın dili
Algoritmalar	dinamik programlama, açgözlü yöntemler ve karmaşıklık

KONUM

Bu Ders Sana Göre mi?

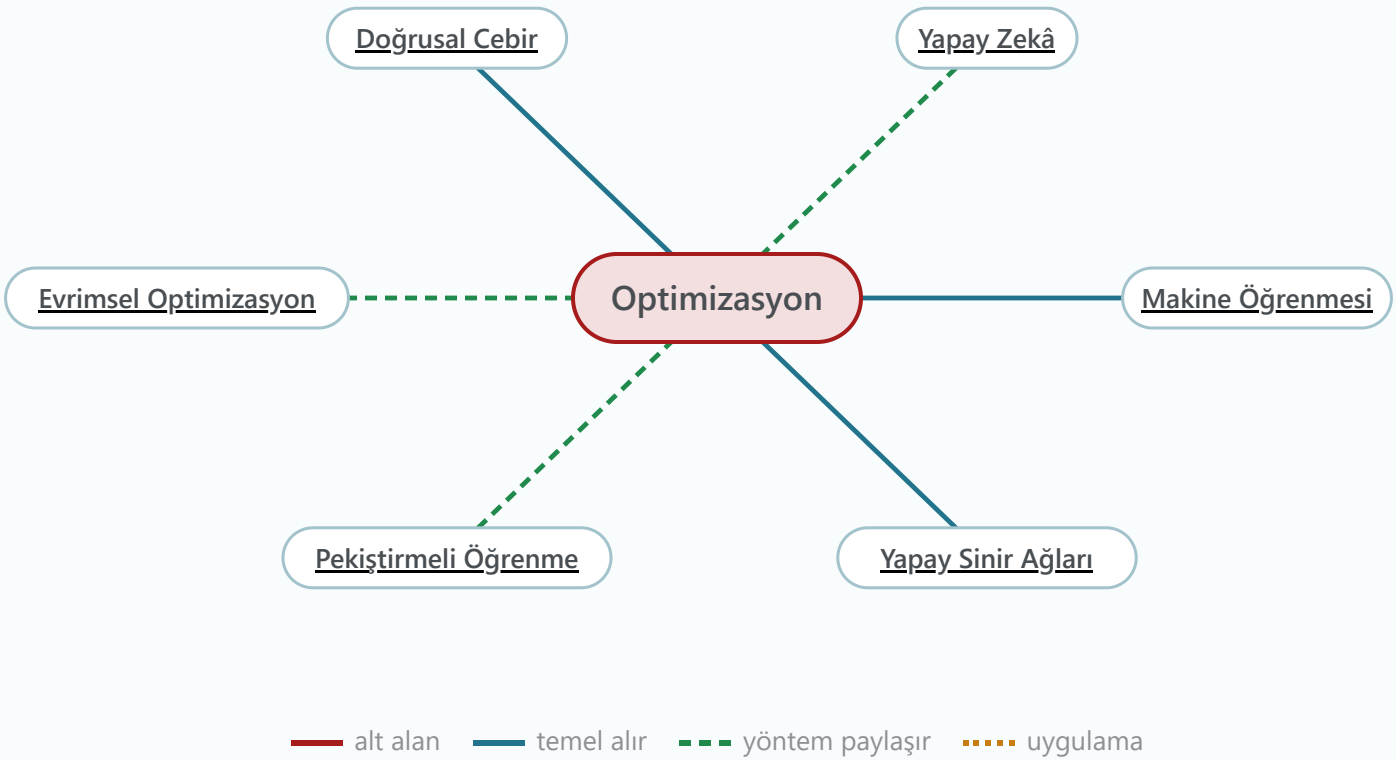
Kendine sor; "hayır" dediklerin için sağdaki sütun nereden başlayacağını gösterir.

Kendine sor	Hayırsa önce
Çok değişkenli bir fonksiyonun gradyanını hesaplayabiliyor musun?	Analiz dersindeki kısmi türev ve zincir kuralı
Bir fonksiyonun Taylor açılımını ikinci terime kadar yazabiliyor musun?	Analiz dersindeki Taylor serileri
Matris-vektör çarpımı yapıp bir doğrusal sistemi çözebiliyor musun?	Doğrusal Cebir
Python'da NumPy dizilerini bir döngüde adım adım güncelleyebiliyor musun?	Programlamaya giriş, ardından NumPy'a kısa bir giriş

Çoğuna "evet" diyebiliyorsan derse hazırsın; bir ya da iki "hayır" dönem içinde kapanabilir.

KONUM

Dersin Komşuları



ÖRTÜŞME

Örtüştüğü Dersler

İlişki	Ders	Ortak konu
temel alır	Makine Öğrenmesi bu dersi temel alır	kayıp en küçükleme, düzenlileştirme
temel alır	Yapay Sinir Ağları bu dersi temel alır	gradyan inişi, öğrenme hızı
temel alır	Bu ders Doğrusal Cebir dersini temel alır	gradyan, Hessian, konvekslik
yöntem paylaşır	Yapay Zekâ ile yöntem paylaşır	durum uzayında arama, kısıt sağlama
yöntem paylaşır	Pekiştirmeli Öğrenme ile yöntem paylaşır	dinamik programlama, politika gradyanı
yöntem paylaşır	Evrimsel ve Metasezgisel Optimizasyon ile yöntem paylaşır	türevsiz ve global optimizasyon

Ayrıştığı Noktalar (1/2)

Ölçüt	Optimizasyon	Evrimsel Optimizasyon	Yapay Zekâ
Temel soru	Bir amaç fonksiyonunu en iyi yapan değerler nasıl bulunur?	Türev yokken ya da arama uzayı çok büyükken iyi bir çözüm nasıl bulunur?	Bir makine arayarak, çıkarım yaparak ve plan kurarak nasıl akıllıca davranır?
Yaklaşım	amaç ve kısıtların matematiksel modeli	popülasyon ve sezgisel arama	arama, mantık, olasılık ve öğrenme bir arada
Girdi	amaç fonksiyonu ve kısıtlar	amaç fonksiyonu ve çözüm gösterimi	problem tanımı: durumlar, eylemler, hedef
Çıktı	en iyi ya da ona yakın çözüm	iyi bir çözüm (en iyisi garanti değil)	eylem dizisi, plan ya da karar
Başarı ölçütü	amaç değeri, yakınsama hızı	çözüm kalitesi, değerlendirme sayısı	hedefe ulaşma, maliyet, rasyonellik
Matematik yükü	kuramın kendisi	düşük-orta	orta: mantık, olasılık, çizge
Veri ihtiyacı	gerekmez	gerekmez; değerlendirme yeter	az; çoğu zaman model ve bilgi yeter

- **Evrimsel Optimizasyon:** Evrimsel yöntemler türev kullanmadan arar; bu ders problemin yapısını (türev, konvekslik) kullanan yöntemleri öğretir.
- **Yapay Zekâ:** YZ işe problemi etmen ve çevre olarak modellemekle başlar; optimizasyon matematiksel olarak kurulmuş problemi çözer.

Ayrıştığı Noktalar (2/2)

Ölçüt	Optimizasyon	Makine Öğrenmesi	Pekiştirmeli Öğrenme
Temel soru	Bir amaç fonksiyonunu en iyi yapan değerler nasıl bulunur?	Bir program örneklerden nasıl genelleme yapar?	Yalnız ödül sinyaliyle, deneme yanılma yoluyla iyi kararlar nasıl öğrenilir?
Yaklaşım	amaç ve kısıtların matematiksel modeli	istatistiksel öğrenme	ödülle deneme yanılma
Girdi	amaç fonksiyonu ve kısıtlar	etiketli ya da etiketsiz örnekler	ortamla etkileşim: durum ve ödül
Çıktı	en iyi ya da ona yakın çözüm	tahmin yapan model	politika: durumdan eyleme kural
Başarı ölçütü	amaç değeri, yakınsama hızı	görülmemiş veride hata	toplam beklenen ödül
Matematik yükü	kuramın kendisi	yüksek: doğrusal cebir, olasılık, optimizasyon	yüksek: olasılık, dinamik programlama
Veri ihtiyacı	gerekmez	orta ve çok	etkileşimle üretilir

- **Makine Öğrenmesi:** MÖ eğitim kaybını küçültürken yeni veride başarıyı da ister; optimizasyon yalnız verilen amacı en iyi yapar.
- **Pekiştirmeli Öğrenme:** Pekiştirmeli öğrenme ödülü deneme yanılmayla en büyükler; bu derste amaç fonksiyonu baştan açıkça yazılıdır.

UYGULAMA

Uygulama Alanları

Model eğitimi
Kaybı en küçükleyen ağırlıkları bulmak.

SVM
En geniş aralıklı ayırıcı: konveks problem.

Hiperparametre arama
Öğrenme hızı ve düzenlileştirme seçimi.

Lojistik
Rota, taşıma ve depo yerleşimi.

Çizelgeleme
Vardiya, üretim ve ders programı.

Portföy seçimi
Getiri ile riski dengelemek.

UYGULAMA

Riskler ve Etik

Yanlış amaç fonksiyonu

Ölçülebileni en iyilemek asıl hedefi bozabilir: yalnız tıklamayı en büyükleyen sistem tık tuzağını öne çıkarır.

Unutulan kısıt

Modele yazılmayan kısıt yok sayılır: en ucuz rota yasak yolu, en iyi vardiya planı yasal dinlenmeyi çiğneyebilir.

Kırılgan en iyi çözüm

Kestirilmiş parametrelerle bulunan en iyi çözüm küçük bir sapmada çok kötüleşebilir; Markowitz portföyü buna duyarlıdır.

Kötü ölçekleme

Birimleri çok farklı değişkenler çözücüyü yavaşlatır ya da yanlış "uygun çözüm yok" sonucuna götürebilir.

Ayrıntılı tartışma: Açıklanabilir Yapay Zekâ ve Yapay Zekâ Etiği.

UYGULAMA

Araçlar ve Örnek Veriler

Araçlar ve kütüphaneler

SciPy (scipy.optimize) | CVXPY | Google OR-Tools | PuLP | Gurobi | PyTorch (torch.optim)

Klasik örnek problemler

Diyet problemi | Taşıma problemi | Sırt çantası problemi | Gezgin satıcı | Markowitz portföyü | Rosenbrock fonksiyonu

UYGULAMA

Sık Karıştırılanlar

Yaygın düşünce	Doğrusu
Gradyan inişi her zaman en iyi çözümü bulur.	Konveks olmayan problemlerde yerel bir en iyide ya da eyer noktasında kalabilir.
Öğrenme hızı ne kadar büyükse o kadar hızlı.	Çok büyük adım minimumu aşar ve kayıp ıraksar; çok küçüğü ise eğitimi uzatır.
Tam sayılı problemler yuvarlayarak çözülür.	Doğrusal çözümü yuvarlamak kısıtları bozabilir ya da en iyiden çok uzak kalabilir.
Kayıp en küçükse model en iyisidir.	Eğitim kaybını sıfırlamak aşırı uydurma demek olabilir; YZ'de amaç genellemedir.

SONRASI

Bu Dersten Sonra

Bu dersi temel alan ya da genişleten dersler

Makine Öğrenmesi | Yapay Sinir Ağları

İlgili meslekler

Yöneylem araştırmacısı | Makine öğrenmesi mühendisi | Lojistik ve planlama analisti | Kantitatif analist

Bu dersten sonra makine öğrenmesi ve sinir ağlarında eğitimin nasıl işlediği açıklık kazanır. Türev kullanılamayan problemler için evrimsel ve metasezgisel optimizasyona, ardışık kararlar için pekiştirmeli öğrenmeye, kaynak paylaşımı için yöneylem araştırmasına geçilebilir.

Kaynaklar

- Boyd ve Vandenberghe, *Convex Optimization*, Cambridge University Press, 2004.
- Nocedal ve Wright, *Numerical Optimization*, 2. baskı, Springer, 2006.
- Bertsimas ve Tsitsiklis, *Introduction to Linear Optimization*, Athena Scientific, 1997.
- Bottou, Curtis ve Nocedal, "Optimization Methods for Large-Scale Machine Learning", *SIAM Review* 60(2), 2018.

Son

Bölüm 33: Optimizasyon

← Önceki bölüm · Alanın haritası · Sonraki bölüm →