Çok Amaçlı En Küçük Kareler

T.C. Trakya Üniversitesi
Mühendislik Fakültesi
Elektrik - Elektronik Mühendisliği Bölümü
Kontrol Anabilim Dalı

Dr. Öğr. Üyesi Işık İlber Sırmatel
sirmatel.github.io

Kaynak (source)

Lecture Slides for Introduction to Applied Linear Algebra: Vectors, Matrices, and Least Squares.
Stephen Boyd, Lieven Vandenberghe

Konu listesi

Bölüm 1

Çok amaçlı en küçük kareler problemi

Çok amaçlı en küçük kareler problemi

- norm kare formunda k adet amaç fonksiyonu

$$J_1 = \|A_1 x - b_1\|^2, \; \ldots, \; J_k = \|A_k x - b_k\|^2$$

  şeklinde verilsin. bunların hepsinin küçük değerler almasını sağlayacak şekilde n-vektör x'i seçmek istiyoruz
- A_i $m_i \times n$ matris, b_i m_i-vektör $(i = 1, 2, \ldots, k)$
- bu probleme çok amaçlı (*multi-objective*) en küçük kareler problemi denir
- bu problem

$$\underset{x}{\text{minimize}} \quad \sum_{i=1}^{k} \|A_i x - b_i\|^2$$

  formunda bir çok amaçlı optimizasyon problemidir

Çok amaçlı en küçük kareler problemi

- bu tip problemlere çok ölçütlü (*multi-criterion*) problem de denir
- J_i terimleri çok amaçlı optimizasyon problemindeki amaçlardır (*objectives*)
- x'i herhangi bir J_i'yi minimize edecek şekilde seçebiliriz, ancak biz hepsinin küçük değerler almasını sağlayacak bir x bulmak istiyoruz

Ağırlıklı toplam amaç fonksiyonu

- pozitif ağırlıklar $(\lambda_1, \ldots, \lambda_k)$ seçelim ve

$$\begin{aligned} J &= \lambda_1 J_1 + \ldots + \lambda_k J_k \\ &= \lambda_1 \|A_1 x - b_1\|^2 + \ldots + \lambda_k \|A_k x - b_k\|^2 \end{aligned}$$

  şeklinde bir "ağırlıklı toplam amaç" (*weighted sum objective*) fonksiyonu oluşturalım
- J'yi minimize edecek x'i seçmek istiyoruz
- $\lambda_1 = 1$ olarak alıp J_1'e "birincil amaç" (*primary objective*) diyebiliriz
- λ_i'lerin yorumlanması: J_i'nin küçük değerler almasını birincil amaca oranla ne kadar önemsiyorsak λ_i'yı o kadar büyük seçmeliyiz
- iki ölçütlü bir problem için amaç fonksiyonu:

$$J_1 + \lambda J_2 = \|A_1 x - b_1\|^2 + \lambda \|A_2 x - b_2\|^2$$

İstifleme ile ağırlıklı toplam minimizasyonu

- ağırlıklı toplam amaç fonksiyonunu şu şekilde yazalım

$$J = \left\| \begin{bmatrix} \sqrt{\lambda_1}(A_1 x - b_1) \\ \vdots \\ \sqrt{\lambda_k}(A_k x - b_k) \end{bmatrix} \right\|^2$$

- buradan, amaç fonksiyonu

$$\tilde{A} = \begin{bmatrix} \sqrt{\lambda_1} A_1 \\ \vdots \\ \sqrt{\lambda_k} A_k \end{bmatrix}, \qquad \tilde{b} = \begin{bmatrix} \sqrt{\lambda_1} b_1 \\ \vdots \\ \sqrt{\lambda_k} b_k \end{bmatrix}$$

  tanımlarıyla $J = \|\tilde{A}x - \tilde{b}\|$ formunda oluşur
- dolayısıyla J'yi minimize etmek için standart (tek ölçütlü) en küçük kareler yöntemini kullanabiliriz

Ağırlıklı toplam çözümü

- $\tilde{A}$'nın sütunlarının doğrusal bağımsız olduğunu varsayalım
- $J = \|\tilde{A}x - \tilde{b}\|$ için en küçük kareler çözümü:

$$\begin{aligned}\hat{x} &= (\tilde{A}^T\tilde{A})^{-1}\tilde{A}^T\tilde{b} \\ &= (\lambda_1 A_1^T A_1 + \cdots + \lambda_k A_k^T A_k)^{-1}(\lambda_1 A_1^T b_1 + \cdots + \lambda_k A_k^T b_k)\end{aligned}$$

- $\hat{x}$'i $\tilde{A}$'nın QR ayrıştırmasıyla hesaplayabiliriz
- A_i matrisleri geniş olabilir, veya sütunları doğrusal bağımlı olabilir

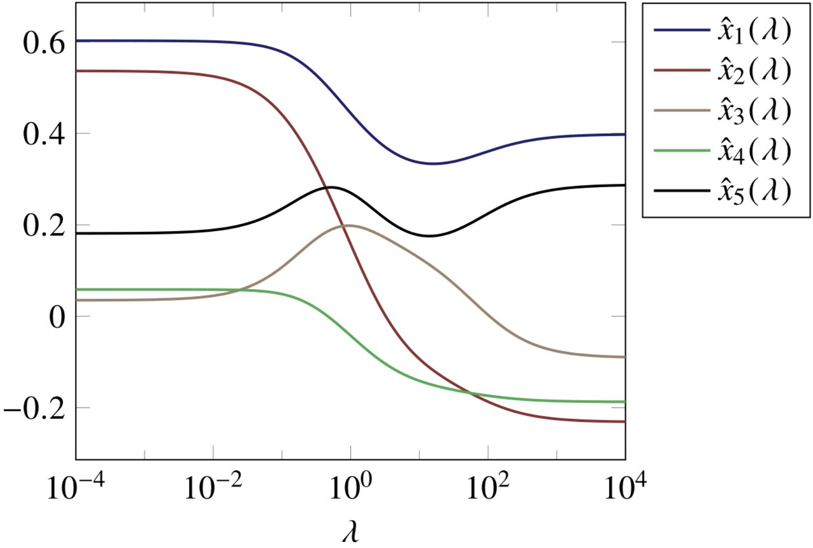
Optimal ödünleşim eğrisi

- amaçları J_1 ve J_2 olan iki ölçütlü probleme bakalım
- $\hat{x}(\lambda)$ $J_1 + \lambda J_2$'nin minimize eden x değeri olsun
- $J_1(z) < J_1(\hat{x}(\lambda))$ ve $J_2(z) < J_2(\hat{x}(\lambda))$ eşitsizliklerini sağlayan bir z noktası mevcut değilse $\hat{x}(\lambda)$'ya "Pareto optimal" denir
- farklı şekilde söylersek: her iki amaç için de $\hat{x}(\lambda)$'ten daha iyi bir x noktası yoktur
- optimal ödünleşim eğrisi (*optimal trade-off curve*):

$$(J_1(\hat{x}(\lambda)),\, J_2(\hat{x}(\lambda))) \quad (\lambda > 0 \text{ için})$$

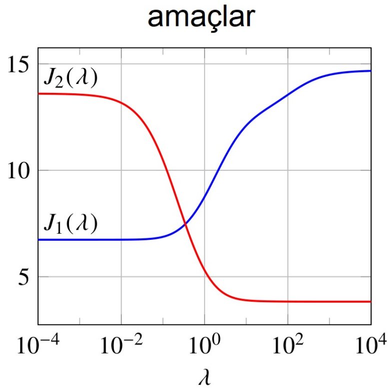
Optimal ödünleşim eğrisi - örnek

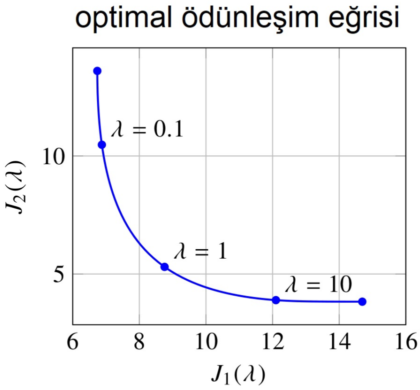
10×5 A_1 ve A_2 matrisleri için $J_1 + \lambda J_2$ amaçlı bir problem



Optimal ödünleşim eğrisi - örnek

10×5 A_1 ve A_2 matrisleri için $J_1 + \lambda J_2$ amaçlı bir problem





Çok amaçlı en küçük kareleri kullanmak

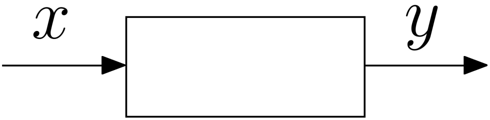
aşağıdaki prosedürü uygulayarak elimizdeki problemi bir çok amaçlı en küçük kareler problemi olarak formüle edebiliriz

- birincil amacı belirleyelim
  - birincil amaç, minimize etmek istediğimiz temel büyüklüktür
- bir veya daha fazla ikincil (*secondary*) amaç seçelim
  - bunlar mümkünse küçük olmasını istediğimiz büyüklüklerdir
  - örneğin: x'in uzunluğu, pürüzlülüğü, verilen bir noktaya uzaklığı
- beğendiğimiz (veya hoş görebildiğimiz) bir $\hat{x}(\lambda)$ bulana kadar ağırlıkları (λ_i) ayarlayalım. örneğin, $J_1 + \lambda J_2$ amaçlı iki ölçütlü bir problem için:
  - J_2 fazla büyükse λ'yı arttır
  - J_1 fazla büyükse λ'yı azalt

Bölüm 2

Örnek: Kontrol

Kontrol



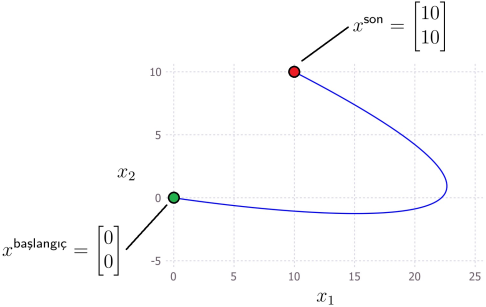
- n-vektör x: girişler (*inputs*) veya eylemler (*actions*)
- m-vektör y: çıkışlar (*outputs*) veya sonuçlar (*results*)
- giriş ve çıkışlar arası ilişkiyi

  $$y = Ax + b$$

  formundaki afin giriş-çıkış modeli ile ifade ediyoruz
- bu model, gerçek bir sistemin bir matematik modeli
- sistem: uygulanan girişlere çıkışlar ile tepki veren "şey"
- A ve b biliniyor (analitik modellerden, uydurmadan vb.)
- kontrol problemi (optimizasyon bakış açısıyla): x'in ve y'nin fonksiyonu olan birden fazla amaç fonksiyonunu optimize edecek şekilde (y'yi belirleyen) x'i seçmek

Örnek: Minimum enerjili durum geçişi

standart (tek ölçütlü) problem formülasyonu



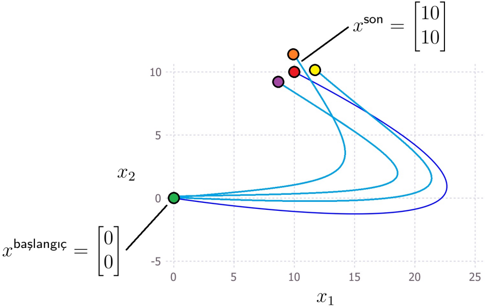
$$
\begin{array}{ll}
\underset{u}{\text{minimize}} & \|u\|^2 \\
\text{bağlı} & x_{t+1} = Ax_t + Bu_t, \quad t = 1, 2, \ldots, T-1 \\
& x_0 = x^{\text{başlangıç}} \\
& x_T = x^{\text{son}}
\end{array}
$$

Çok amaçlı kontrol

- tipik birincil amaç: $J_1 = \|y - y^{\mathsf{des}}\|^2$ (burada y^{des} verilen bir hedef çıkış)
- tipik ikincil amaçlar:
  - x küçük olsun: $J_2 = \|x\|^2$
  - x bir nominal girişten uzak olmasın: $J_2 = \|x - x^{\mathsf{nom}}\|^2$

Örnek: Minimum enerjili durum geçişi

çok amaçlı (iki ölçütlü) problem formülasyonu



$$
\begin{array}{ll}
\underset{u}{\text{minimize}} & \|x_T - x^{\text{son}}\|^2 + \lambda \|u\|^2 \\
\text{bağlı} & x_{t+1} = Ax_t + Bu_t, \quad t = 1, 2, \ldots, T-1 \\
& x_0 = x^{\text{başlangıç}}
\end{array}
$$

Dayanıklı kontrol

- dayanıklı (*robust*) kontrol: kontrol yöntemlerini belirsizliklere (*uncertainties*) dayanıklı hale getirmek amacıyla belirsizlik modellerinin kontrol yöntemi formülasyonuna eklenmesini içeren kontrol dalı
- (genel olarak) dayanıklı XYZ: XYZ yöntemlerini belirsizliklere dayanıklı hale getirmek amacıyla belirsizlik modellerinin yöntem formülasyonuna eklenmesini içeren XYZ dalı (örnekler: dayanıklı optimizasyon, dayanıklı istatistik, dayanıklı bağlanım)

Dayanıklı kontrol

- not: dayanıklı kontrolde birçok yöntem vardır, burada en küçük kareler formunda çok sade bir yöntemi inceliyoruz
- K adet farklı giriş-çıkış modelimiz (veya, senaryomuz) var:

$$y^{(k)} = A^{(k)}x + b^{(k)}, \qquad k = 1, 2, \ldots, K$$

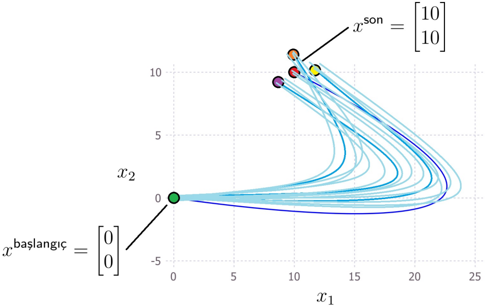
- bu modeller sistemdeki belirsizliği temsil eder
- k. model doğru ise $y^{(k)}$ giriş x için olan çıkıştır
- bütün modeller üzerinden ortalama birincil amaç:

$$\frac{1}{K}\sum_{k=1}^{K}\|y^{(k)} - y^{\mathsf{des}}\|^2$$

- x için de terimler ekleyebiliriz, örneğin $\lambda\|x\|^2$
- bu problem formülasyonu bütün senaryolarda iyi başarım gösteren bir x değeri seçmemizi sağlar

Örnek: Minimum enerjili durum geçişi

dayanıklı, çok amaçlı (iki ölçütlü) problem formülasyonu



$$\underset{u}{\text{minimize}} \quad \frac{1}{K}\sum_{k=1}^{K} \|x_T^{(k)} - x^{\text{son}}\|^2 + \lambda\|u\|^2$$

$$\text{bağlı} \quad k = 1,\, 2,\, \ldots,\, K \text{ için:}$$

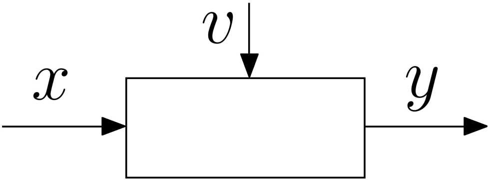
$$x_{t+1}^{(k)} = A x_t^{(k)} + B u_t, \quad t = 1,\, 2,\, \ldots,\, T-1$$

$$x_0^{(k)} = x^{\text{başlangıç}}$$

Bölüm 3

Örnek: Kestirme ve evirme

Kestirme (*estimation*)



- n-vektör x kestirmek istediğimiz parametreleri içerir
- m-vektör y ölçümleri (*measurements*) içerir
- m-vektör v (bilinmeyen) gürültü (*noise*) veya ölçüm hataları terimlerini içerir
- ölçüm modeli:

$$y = Ax + v$$

- $n \times n$ matris A parametrelerle ölçümleri ilişkilendirir
- temel en küçük kareler kestirme problemi: v'nin küçük olduğunu (ve A'nın sütunlarının bağımsız olduğunu) varsayalım. $J_1 = \|Ax - y\|^2$'yi minimize eden x'i bularak kestirme (x'i tahmin etme) yapabiliriz

Düzenlileştirilmiş evirme (*inversion*)

- x ile ilgili önsel bilgiyi (*prior information*) kestirme problemine dahil ederek çok daha iyi sonuçlar elde edebiliriz. x ile ilgili önsel bilgiye örnekler:
  - x çok büyük olmamalı ($\|x\|$ küçük olmalı)
  - x pürüzsüz olmalı ($\|Dx\|$ küçük olmalı)
- bunları ikincil amaçlar olarak ifade edelim:
  - $J_2 = \|x\|^2$ (Tikhonov düzenlileştirmesi (*regularization*))
  - $J_2 = \|Dx\|^2$ ($\|Dx\|^2$: Dirichlet enerjisi)
- problem: $J_1 + \lambda J_2$ amacını minimize etmek
- beğendiğimiz sonuçları elde edinceye kadar λ'yı değiştirebiliriz
- λ'nın fonksiyonu olarak $\hat{x}(\lambda)$ eğrisine düzenlileştirme yolu (*regularization path*) denir
- Tikhonov düzenlileştirmesi ile, A matrisinin sütunları doğrusal bağımlı (örneğin, A geniş matris) olduğunda bile çözüm bulunabilir

Örnek: Görüntü netleştirme

- x: bir görüntüyü
- A matrisi: bulanıklaştırma (*blurring*) operatörü
- $y = Ax + v$; y: bulanıklaştırılmış, gürültülü bir görüntü
- en küçük kareler netleştirme (*de-blurring*) problemi:

$$\|Ax - y\|^2 + \lambda(\|D_{\mathsf{v}}x\|^2 + \|D_{\mathsf{h}}x\|^2)$$

  amacını minimize edecek x'i seçmek (burada D_{v} ve D_{h} dikey ve yatay fark alma (*differencing*) operatörleri)
- λ netleştirilen görüntünün pürüzsüzleştirme ayarıdır

Örnek: Görüntü netleştirme

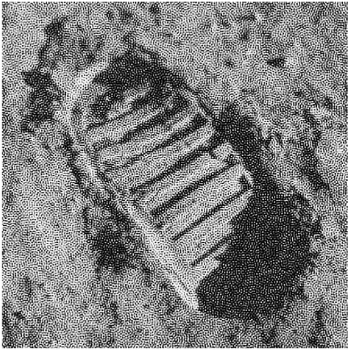
bulanıklaştırılmış,
gürültülü görüntü

$\lambda = 0.007$ ile düzenlileştirilmiş
evirme ile netleştirilmiş görüntü

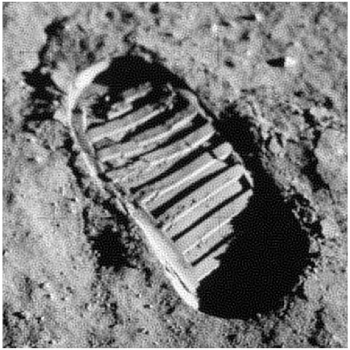
Örnek: Görüntü netleştirme

düzenlileştirme yolu (örnekler)

$\lambda = 10^{-6}$



$\lambda = 10^{-4}$



Örnek: Görüntü netleștirme

düzenlileștirme yolu (örnekler)

$\lambda = 10^{-2}$

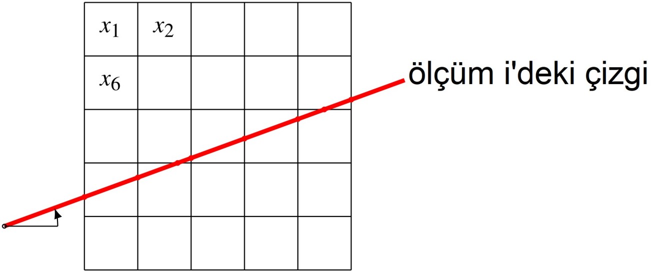
$\lambda = 1$

Örnek: Tomografi

- x: n adet vokselden (veya pikselden) oluşan, ilgi bölgesindeki (*region of interest*) büyüklüklerin değerleri
- $y = Ax + v$; y: bölge içinden geçen çizgiler üzerindeki integrallerin ölçümleri

$$y_i = \sum_{i=1}^{n} A_{ij} x_j + v_i$$

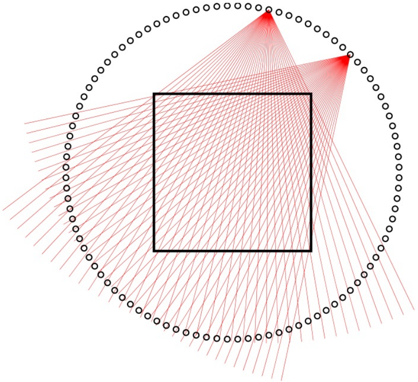
- A_{ij}: ölçüm i'deki çizgiyle voksel j'nin kesişiminin uzunluğu

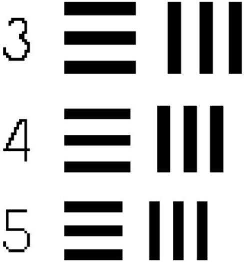


Örnek: Tomografi

- en küçük kareler yöntemini kullanarak tomografik geriçatma (*reconstruction*) yapmak istiyoruz
- birincil amaç: $\|Ax - y\|^2$
- düzenlileştirme terimleri, x ile ilgili önsel bilgileri içerir ve bunları problem formülasyonuna eklememizi sağlar
- örneğin, eğer x bölge üzerinde pürüzsüz şekilde değişiyorsa, her vokseli kendi komşularına bağlayan çizge için Dirichlet enerjisini düzenlileştirme terimi olarak kullanırız

Örnek: Tomografi

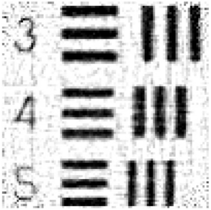




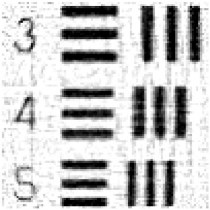
- sol tarafta: 4000 çizgi (100 nokta, nokta başına 40 çizgi)
- sağ tarafta: soldaki kare bölgeye yerleştirilen obje
- ilgi bölgesi 10000 piksele ayrılmış durumda

Örnek: Tomografi

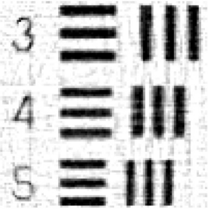
$\lambda = 10^{-2}$



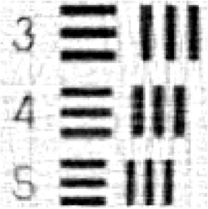
$\lambda = 10^{-1}$



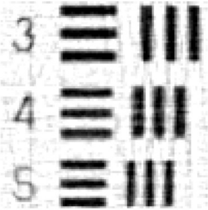
$\lambda = 1$



$\lambda = 5$



$\lambda = 10$



$\lambda = 100$

Bölüm 4

Örnek: Düzenlileştirilmiş veri uydurma

Düzenlileştirme ihtiyacının nedeni

- $f_1(x) = 1$ ile ($y \approx f(x)$ bağıntısı için)

$$\hat{f}(x) = \theta_1 f_1(x) + \cdots + \theta_p f_p(x)$$

  modeli için veri uydurma problemini ele alalım
- θ_i, $\hat{f}(x)$'in $f_i(x)$'e duyarlılığıdır (*duyarlılık*)
- dolayısıyla, θ_i'in büyük olması modelin $f_i(x)$'e çok duyarlı olması anlamına gelir
- (θ_1 istisna çünkü $f_1(x) = 1$ sabit)
- sonuç olarak, $\theta_2,\ \theta_3,\ \ldots \theta_p$ değerlerinin çok büyük olmasını istemeyiz

Düzenlileștirilmiș veri uydurma

- elimizde $x^{(1)}, x^{(2)}, \ldots, x^{(N)}, y^{(1)}, y^{(2)}, \ldots, y^{(N)}$ șeklinde bir eğitim veri kümesi olsun
- veri kümesi üzerinde uydurma hatasını $A\theta - y$ ile ifade edelim
- düzenlileștirilmiș veri uydurma problemi:

$$\|A\theta - y\|^2 + \lambda\|\theta_{2:p}\|^2$$

amacını minimize edecek șekilde θ'yı seçmek

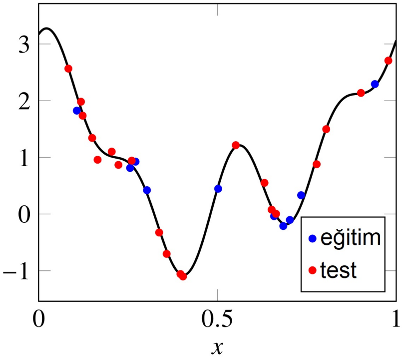
- λ ($\lambda > 0$): düzenlileștirme parametresi
- $\hat{y} = X^T\beta + \nu\mathbf{1}$ formundaki bağlanım modeli için

$$\|X^T\beta + \nu\mathbf{1} - y\|^2 + \lambda\|\beta\|^2$$

amacı minimize edilir

- λ bir test veri kümesi üzerinde geçerleme ile seçilir

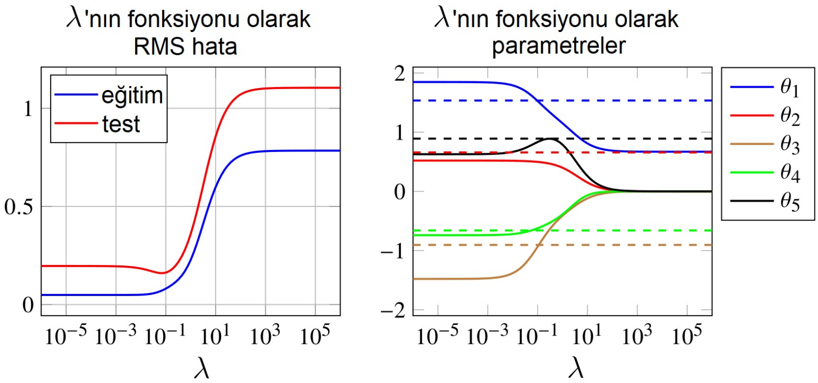
Düzenlileştirilmiş veri uydurma - Örnek



- siyah çizgi: yapay veri oluşturmak için kullanılan eğri
- 10 mavi nokta: eğitim, 20 kırmızı nokta: test
- 5 parametreli bir model uyduruyoruz (ω_k ve ϕ_k veriliyor):

$$\hat{f}(x) = \theta_1 + \sum_{k=1}^{4} \theta_{k+1} \cos(\omega_k x + \phi_k)$$

Düzenlileştirilmiş veri uydurma - Örnek



- minimum test RMS hata değeri $\lambda = 0.08$ ile elde ediliyor
- λ'yı arttırınca parametrelerin $(\theta_2, \ldots \theta_5)$ değerleri azalıyor
- kesikli çizgiler gerçek parametre değerleri
- 0.08'e yakın λ için kestirilen parametreler gerçeğe yakın