

Hafta 01 - Giriş

BGM 565 - Siber Güvenlik için Makine Öğrenme Yöntemleri

Bilgi Güvenliği Mühendisliği
Yüksek Lisans Programı

Dr. Ferhat Özgür Çatak
ozgur.catak@tubitak.gov.tr

İstanbul Şehir Üniversitesi
2018 - Bahar

- ◀ ◻ ▶ ◀ ◻ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

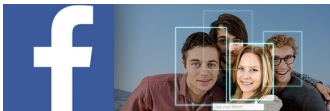
İçindekiler

- 1 Makine Öğrenmesi
 - Makine Öğrenme Uygulamaları
 - Makine Öğrenmesinin Geleceği
 - Ders Hakkında Bilgiler
 - Uygulama Alanları
- 2 Yöntemler
 - Tanım
 - Formal Model
 - Makine Öğrenme Problemlerinin Sınıflandırılması
- 3 Danışmanlı öğrenme (Supervised learning)
 - Sınıflandırma
- 4 Danışmansız Öğrenme (Unsupervised Learning)
 - Giriş
 - Kümeleme
 - Boyut Azaltımı
- 5 Regresyon
 - Tek Değişkenli Doğrusal Regresyon
 - Maliyet fonksiyonu (Cost function)
 - Gradient Descent
 - Doğrusal Regresyon - Python
- Regresyon
 - Sıralama (Ranking)

Makine Öğrenme Uygulamaları



Şekil: Self-driving cars



Şekil: Facebook face recognition



Şekil: Öneri Sistemleri



Şekil: human-brain-project

Ders

Notlandırma

- ▶ Derse katılım : %10
- ▶ Ödevler : %30
- ▶ Proje : %25
- ▶ Final : %35

Kaynaklar

- ▶ *Applications of Data Mining in Computer Security* Daniel Barbara and Sushil Jajodia
- ▶ *Machine Learning and Data Mining for Computer Security* Marcus A. Maloof

Araçlar I

Python 3.5 (Öneri: 64-Bit)

► Kütüphaneler

- scikit-learn (makine öğrenmesi)
- keras (derin öğrenme)
- tensorflow (derin öğrenme)
- pandas (veri manipülasyonu)
- matplotlib (veri görselleştirme)

► Geliştirme Ortamı

- Spyder (Önerilen - MATLAB benzeri ortam)
- Pycharm (Alternatif)

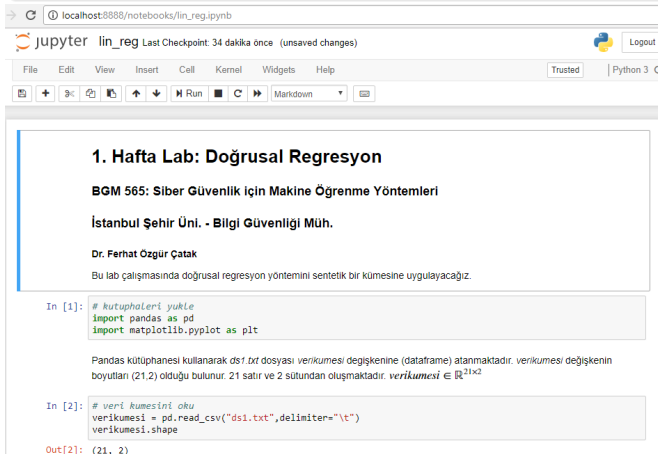
Python kütüphane kurulumu

```
pip install -U kutuphane_adi
```

Araçlar II

Lab Ortamı

► Jupyter: komut satırında jupyter notebook



localhost:8888/notebooks/lin_reg.ipynb

jupyter lin_reg Last Checkpoint: 34 dakika önce (unsaved changes) Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 C

1. Hafta Lab: Doğrusal Regresyon

BGM 565: Siber Güvenlik için Makine Öğrenme Yöntemleri

İstanbul Şehir Üni. - Bilgi Güvenliği Müh.

Dr. Ferhat Özgür Çatak

Bu lab çalışmasında doğrusal regresyon yöntemini sentetik bir kümesine uygulayacağız.

```
In [1]: # kutuphaları yükle
import pandas as pd
import matplotlib.pyplot as plt
```

Pandas kütüphanesi kullanarak `ds1.txt` dosyası `verikumesi` değişkenine (dataframe) atanmaktadır. `verikumesi` değişkenin boyutları (21,2) olduğu bulunur. 21 satır ve 2 sütundan oluşmaktadır. $verikumesi \in \mathbb{R}^{21 \times 2}$

```
In [2]: # veri kümesini oku
verikumesi = pd.read_csv("ds1.txt", delimiter="\t")
verikumesi.shape
```

Out[2]: (21, 2)

Makine Öğrenmesi I

Genel Tanım

- **Tanım:** Öğrenme modelini, örnek verilerden otomatik olarak çıkarmanın ve genelleştirmenin hesaplama sürecidir.
- Öğrenme modelleri, **veri ve nedensellikler arasındaki bağımlılıkları ve girdi ile çıktı arasındaki korelasyonları** tanımlamak için istatistiksel fonksiyonları veya kuralları kullanmaktadır.

Makine Öğrenmesi II

Neden Öğrenme?

- ▶ Örnek bir veri kümesi kullanarak bilgisayarları bir performans kriterini optimize etmek için programlamak.
- ▶ Makine öğrenimi her olay için gerekli değildir.
 - ▶ Bordro hesabı
- ▶ Öğrenme örnekleri
 - ▶ İnsan deneyiminin olmadığı durumlar (Bio-enformatik)
 - ▶ İnsan deneyiminin tam olarak açıklanamadığı durumlar (Ses tanıma, görüntü tanıma)
 - ▶ Çözümün zaman içinde değişmesi (saldırı türleri)

Makine Öğrenmesi III

Makine Öğrenmesine Ne Zaman Gerek Var?

- ▶ Programlamak için karmaşık görevler:
 - ▶ Canlılar tarafından yapılan görevler
 - ▶ araba kullanmak
 - ▶ ses tanıma
 - ▶ resim tanıma
 - ▶ İnsan yeteneklerinin ötesinde görevler (Büyük veri)
 - ▶ Astronomi verileri
 - ▶ DNA analizleri
 - ▶ Arama motorları
- ▶ Uyarlanabilirlik (Adaptivity)
 - ▶ Programlar yazıldıktan sonra değiştirilemezler
 - ▶ İstenmeye e-posra algılama programları

Örnek Model

Örnek Model

- ▶ **Öğrenme:** örneklerden (veri) genel modellerin çıkarılması
 - ▶ **Veri:** erişimi kolay ve etrafımızda dolu
 - ▶ **Bilgi:** Pahalı ve sınırlı
- ▶ **Örnek:**
 - ▶ Müşteri alışveriş hareketlerinden (transactions) müşteri davranışının bulunması
 - ▶ "Da Vinci Şifresi"ni satın alanlar "Kayıp Sembol"de satın aldı.
- ▶ **Hedef:** Veriden anlamlı tahminler yapan modeller oluşturmak.

Uygulama Alanları

Uygulama Alanları

- ▶ **Perakende:** Sepet analizi, müşteri ilişkileri yönetimi
- ▶ **Finans:** kredi skorları, sahtekarlık tespiti (fraud)
- ▶ **Üretim:** Optimizasyon, kontrol
- ▶ **Haberleşme:** spam filtreleri, saldırı tespiti,
- ▶ **Bilim:** yüksek boyutlu fizik verilerinin analiz edilmesi, biyoloji
- ▶ **Web:** Arama motorları

İçindekiler

- 1 Makine Öğrenmesi
 - Makine Öğrenme Uygulamaları
 - Makine Öğrenmesinin Geleceği
 - Ders Hakkında Bilgiler
 - Uygulama Alanları
- 2 Yöntemler
 - Tanım
 - Formal Model
 - Makine Öğrenme Problemlerinin Sınıflandırılması
- 3 Danışmanlı öğrenme (Supervised learning)
 - Sınıflandırma
 - Regresyon
 - Sıralama (Ranking)
- 4 Danışmansız Öğrenme (Unsupervised Learning)
 - Giriş
 - Kümeleme
 - Boyut Azaltımı
- 5 Regresyon
 - Tek Değişkenli Doğrusal Regresyon
 - Maliyet fonksiyonu (Cost function)
 - Gradient Descent
 - Doğrusal Regresyon - Python

Tanım II

Notasyon

- ▶ Sayısal değerler (scalar) küçük harf: x, λ, η , örnek $x = 3.14$
- ▶ vektörler kalın harf: \mathbf{x} , örnek $\mathbf{x} = [1.2, 2.4, \dots 0.2]$
- ▶ Bir vektörün i . elemanı: $\mathbf{x}^{(i)}$
- ▶ Girdi veri kümesi : $\mathcal{X} \in \mathbb{R}^{m \times n} \Rightarrow m$ adet satır (örnek) ve n boyutlu uzay (nitelik sayısı, kolon sayısı)
- ▶ k boyutlu sınıf etiket kümesi: $\mathcal{C} = \{C_1, \dots C_k\}$
 - ▶ C : Ağ saldırıları: {DDoS, SQL Injection, XSS v.b.}
 - ▶ C : Zararlı Yazılım: {Trojan, Backdoor, Virus, Botnet v.b.}
- ▶ Girdi örnekleri ve hedef etiket çiftleri:

$$X = \{(\mathbf{x}_i, y_i) | \mathbf{x} \in \mathbb{R}^{m \times n}, y \in \{C_1, \dots C_k\}\}_{i=1}^n$$

Formal Model I

İstatiksel Öğrenme Teorisi

- ▶ **Öğrenmenin girdileri**
 - ▶ **Alan kümesi (Domain set):** Etiketlenmek istenen veri kümesi, \mathcal{X} . Zararlı yazılım öğrenme problemi. Nitelikler (DNS, API). Diğer isimlendirmeler: *örnekler, örnek uzayı*
 - ▶ **Etiket kümesi (Label set):** iki elemanlı küme, genellikle $\{0, 1\}$ veya $\{-1, +1\}$. \mathcal{Y} olabilecek etiketler kümesini gösterebilir. Zararlı yazılım analizi için $\mathcal{Y} \in \{0, 1\}$. 0: Normal, 1: zararlı yazılım
 - ▶ **Eğitim verisi (Training data):** $S = ((\mathbf{x}_1, y_1), \dots, (\mathbf{x}_m, y_m))$. Sonlu bir dizi çift $\mathcal{X} \times \mathcal{Y}$. Diğer adlandırmalar: *Eğitim veri kümesi, training set*
- ▶ **Öğrenmenin çıktıları:** Tahmin kuralı üretmesi istenir $h : \mathcal{X} \rightarrow \mathcal{Y}$.
Hipotez, sınıflandırıcı, model. Tahmin: $h(\mathbf{x}) \rightarrow \hat{y}$. Gerçek sınıflandırıcı $f : \mathcal{X} \rightarrow \mathcal{Y}$, $f(\mathbf{x}) \rightarrow y$. \hat{y} : tahmin edilen sınıf (zararlı yazılım veya değil), y : gerçek sınıf

Formal Model II

İstatistiksel Öğrenme Teorisi

- **Başarı ölçütleri:** Sınıflandırıcı hatası, \mathcal{D} 'den (Eğitim veri kümesi) rastgele seçilen bir örneğin \mathbf{x} kullanılarak $h(\mathbf{x}) \neq f(\mathbf{x})$ ifadesinin olasılığıdır.

$$L_{\mathcal{D},f}(h) = \mathbb{P}_{\mathbf{x} \sim \mathcal{D}} [h(\mathbf{x}) \neq f(\mathbf{x})]$$

$L_{\mathcal{D},f}(h)$ *genelleştirme hatası (generalization error), risk, h 'in gerçek hatası*

- **Deneyisel risk minimizasyonu:** f bilinmemesi sebebiyle gerçek hatanın hesaplanması imkansız. Bunun yerine *eğitim hatası* hesaplanabilir.

$$L_{\mathcal{D}}(h) = \frac{|\{i \in [m] : h(\mathbf{x}_i) \neq y_i\}|}{m}$$

- ▶ **İlişkilendirme kuralları (Association rule learning):** değişkenler arasında ilişkinin öğrenilmesi
- ▶ **Danışmanlı öğrenme (Supervised learning):** Değişkenlerden çıktı tahmini
 - ▶ Sınıflandırma (Classification)
 - ▶ Regresyon (Regression)
 - ▶ Sıralama (Ranking)
 - ▶ Sıralı kategoriler (Puanlar)
- ▶ **Danışmansız öğrenme (Unsupervised learning):** Veri üzerinde bulunan desenin çıkarılması
 - ▶ Kümeleme (Clustering)
 - ▶ Boyut azaltma (Dimensionality reduction)
- ▶ **Yarı danışmanlı öğrenme (Semi-supervised learning):** Büyük miktarda etiketlenmemiş veriyle etiketlenmiş küçük bir miktarda veri
- ▶ **Güçlendirme öğrenimi (Reinforcement learning):** Kümülatif ödül en üst düzeye çıkarma

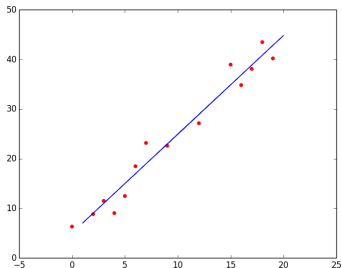
İçindekiler

- 1 Makine Öğrenmesi
 - Makine Öğrenme Uygulamaları
 - Makine Öğrenmesinin Geleceği
 - Ders Hakkında Bilgiler
 - Uygulama Alanları
- 2 Yöntemler
 - Tanım
 - Formal Model
 - Makine Öğrenme Problemlerinin Sınıflandırılması
- 3 Danışmanlı öğrenme (Supervised learning)
 - Sınıflandırma
 - Regresyon
 - Sıralama (Ranking)
- 4 Danışmansız Öğrenme (Unsupervised Learning)
 - Giriş
 - Kümeleme
 - Boyut Azaltımı
- 5 Regresyon
 - Tek Değişkenli Doğrusal Regresyon
 - Maliyet fonksiyonu (Cost function)
 - Gradient Descent
 - Doğrusal Regresyon - Python

Hafta 01 - Giriş

Regresyon (Regression) I

Örnek: Bir ağ içerisinde toplam paket sayısının İstemci sayısı ile değişimi



- ▶ x : istemci sayısı
- ▶ y : paket sayısı
- ▶ $y = wx + b$
- ▶ $y = h_{\theta}(x)$
 - ▶ h : model
 - ▶ θ : parametreler
 - ▶ $\theta = (w, b)$

Regresyon (Regression) II

Uygulama Alanları

- ▶ Bilgisayar sayısının ağ trafiğine etkisi
- ▶ Ürün fiyatlarının satış üzerine etkisi
- ▶ Bir hastalığın başlangıç yaşı
- ▶ Bir robot kolunun kinematiği

Hafta 01 - Giriş

İçindekiler

- 1 Makine Öğrenmesi
 - Makine Öğrenme Uygulamaları
 - Makine Öğrenmesinin Geleceği
 - Ders Hakkında Bilgiler
 - Uygulama Alanları
- 2 Yöntemler
 - Tanım
 - Formal Model
 - Makine Öğrenme Problemlerinin Sınıflandırılması
- 3 Danışmanlı öğrenme (Supervised learning)
 - Sınıflandırma
 - Regresyon
 - Sıralama (Ranking)
- 4 Danışmansız Öğrenme (Unsupervised Learning)
 - Giriş
 - Kümeleme
 - Boyut Azaltımı
- 5 Regresyon
 - Tek Değişkenli Doğrusal Regresyon
 - Maliyet fonksiyonu (Cost function)
 - Gradient Descent
 - Doğrusal Regresyon - Python

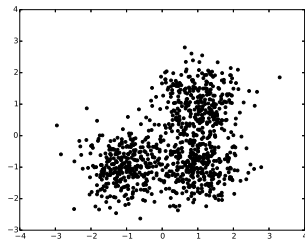
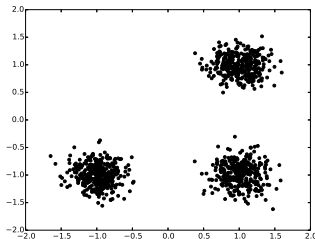
Danışmansız Öğrenme (Unsupervised Learning) I

- ▶ Eğitim veri kümesinin çıktı etiketleri mevcut değil
- ▶ Yoğunluk tahmini (Density estimation): Verideki yapının bulunması
- ▶ Kümeleme (Clustering)
- ▶ Boyut azaltma (Dimensionality reduction)

Kümeleme (Clustering) I

Kümeleme

- ▶ **Amaç:** Nesneleri kümelere gruplamak.
- ▶ Aynı küme içerisinde yer alan nesneler, diğer kümelerde yer alan nesnelere göre daha benzer olacaktır.



Kümeleme (Clustering) II



North Korea Is Demanding the US Prove Its Claim That Pyongyang ...

TIME - Dec 25, 2017

(TOKYO) — North Korea's envoy in charge of U.S. affairs at the United Nations demanded Washington provide evidence to back up its claim Pyongyang was behind the **WannaCry** ransomware attack, an allegation has said was a "baseless provocation" being used to generate tensions. Pak Song Il told The ...

'Show us the evidence': N. Korea invites US to prove Pyongyang's ...

RT - Dec 25, 2017

North Korea asks US for proof of **WannaCry** claim

iTWire - Dec 26, 2017

Should we believe the White House when it says North Korea is ...

CSO Australia - Dec 25, 2017

Put Up or Shut Up: North Korea UN Envoy Demands US Prove ...

Local Source - Sputnik International - Dec 26, 2017

North Korea UN ambassador demands US prove **Wannacry** ...

International - Fox News - Dec 25, 2017



RT



Fox News



CSO Australia



Sputnik Intern...



FileHippo News

[View all](#)

Hafta 01 - Giriş

Kümeleme (Clustering) IV

WannaCry ransomware: North Korea labels US accusation as "absurd"

North Korea says the US has no basis for laying the blame for the global ransomware attack at its door.



By [Liam Tung](#) | December 21, 2017 -- 12:54 GMT (12:54 GMT) | Topic: [Security](#)

Recommended Content:

Downloads: Kaspersky Endpoint Security for Cloud

Forty per cent of businesses say increased infrastructure complexity is pushing budgets to their limits. Kaspersky Endpoint Security Cloud helps small and medium-sized businesses simplify security management for Mac and Windows endpoints, mobile - ...

Free Trial

4

f 30

in 19

t

e

a

RECOMMENDED FOR YOU

Business Resiliency: The Need for 24/7/365 Operations (Japanese)

Boyut Azaltma (Dimensionality Reduction) I

Boyut Azaltımı

- ▶ **Amaç:** Girdi veri kümesinde yer alan değişken sayısının azaltılması
- ▶ **Nitelik seçimi (Feature selection)**
 - ▶ Sadece ilişkili niteliklerin seçilmesi
- ▶ **Nitelik çıkarımı (Feature extraction):**
 - ▶ Veri kümesini daha az boyuta sahip bir uzaya çevirilmesi
 - ▶ Saklama alanı ve hesaplama zamanının azaltılması
 - ▶ Doğrusallıkların kaldırılması
 - ▶ Görselleştirme (2 veya 3 boyutla)
 - ▶ Yorumlanabilir hale getirme

İçindekiler

- 1 Makine Öğrenmesi
 - Makine Öğrenme Uygulamaları
 - Makine Öğrenmesinin Geleceği
 - Ders Hakkında Bilgiler
 - Uygulama Alanları
- 2 Yöntemler
 - Tanım
 - Formal Model
 - Makine Öğrenme Problemlerinin Sınıflandırılması
- 3 Danışmanlı öğrenme (Supervised learning)
 - Sınıflandırma
 - Regresyon
 - Sıralama (Ranking)
- 4 Danışmansız Öğrenme (Unsupervised Learning)
 - Giriş
 - Kümeleme
 - Boyut Azaltımı
- 5 Regresyon
 - Tek Değişkenli Doğrusal Regresyon
 - Maliyet fonksiyonu (Cost function)
 - Gradient Descent
 - Doğrusal Regresyon - Python

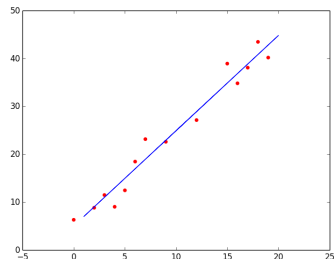
Tek Değişkenli Doğrusal Regresyon I

Linear regression with one variable

Danışmanlı öğrenme (Supervised learning)

Regresyon Problemi:

- Sürekli değerler
- Sınıflandırma: Discrete-value



Tek Değişkenli Doğrusal Regresyon II

Linear regression with one variable

İstemci sayısı (x)	Paket sayısı (y)
80	100
85	100
110	130
...	...

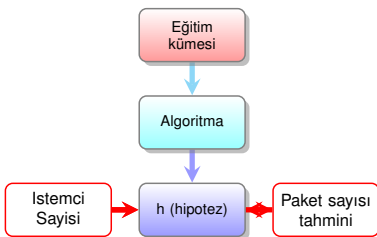
Notasyon: $X \in \mathbb{R}^{m \times n}$

m: örnek sayısı (satır sayısı)

x : girdi değişkeni/nitelikler

y : çıktı/hedef değişken

$(\mathbf{x}^{(i)}, y^i)$: i. eğitim girdisi



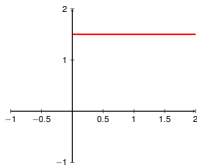
hipotez h gösterilim: $h(x) = w_0 + w_1 x$

Maliyet fonksiyonu I

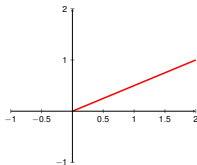
Cost function

İstemci sayısı (x)	Paket sayısı (y)
80	100
85	100
110	130
...	...

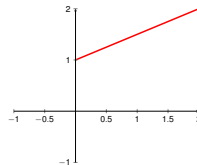
- Hipotez: $h_w = w_0 + w_1 x$
- w_i : parametreler
- w_i nasıl seçilecek?
- Hata: $\epsilon = y - h(\mathbf{x})$



Şekil: $h = 1.5 + 0x$



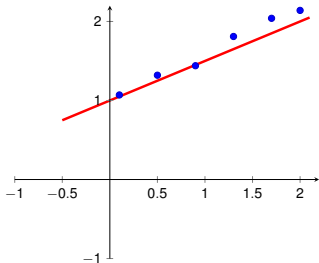
Şekil: $h = 0 + 0.5x$



Şekil: $h = 1 + 0.5x$

Maliyet fonksiyonu II

Cost function



$$\underset{w_0, w_1}{\text{minimize}} = \frac{1}{2m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right)^2$$

$$C(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right)^2 \quad (1)$$

$$\underset{w_0, w_1}{\text{minimize}} C(w_0, w_1)$$

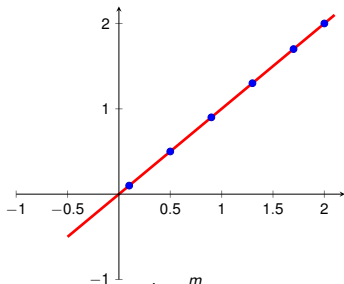
Çözüm: w_0 ve w_1 değerlerinin seçimiyle (x, y) eğitim örneklerinin $h(x)$ değerinin y 'ye yakın olması.

$C(w_0, w_1)$: Maliyet fonksiyonu (Cost function)

Maliyet fonksiyonu III

Cost function

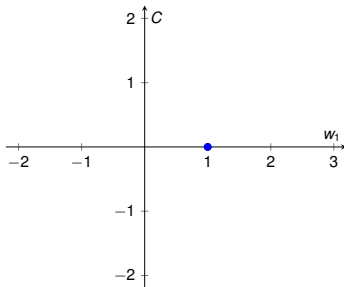
$$h(x) = 0 + x$$



$$\begin{aligned} C(w_0, w_1) &= \frac{1}{2m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right)^2 \\ &= \frac{1}{2m} \left(0^2 + 0^2 + \dots + 0^2 \right) \\ &= 0^2 \end{aligned}$$

(2)

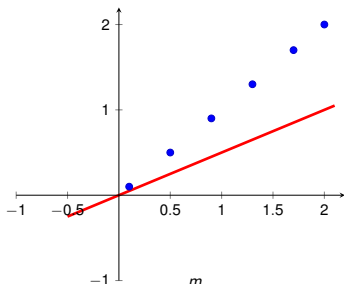
$$C(w_0, w_1)$$



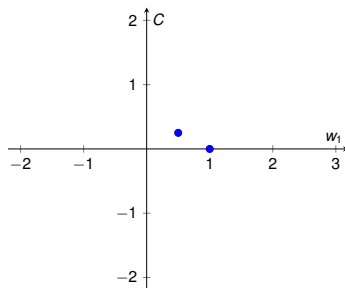
Maliyet fonksiyonu IV

Cost function

$$h(x) = 0 + 0.5x$$



$$C(w_0, w_1)$$



$$C(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right)^2$$

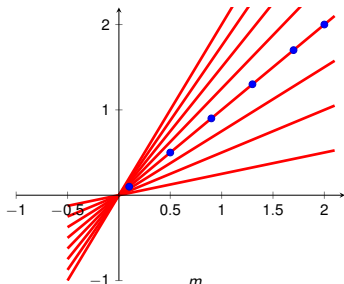
$$= \frac{1}{2 \times 6} \left((0.25 - 0.5)^2 + (0.5 - 1)^2 + (1 - 2)^2 + \dots + (1.5 - 3)^2 \right)$$
$$= 0.25$$

(3)

Maliyet fonksiyonu V

Cost function

$$h(x) = 0 + x$$

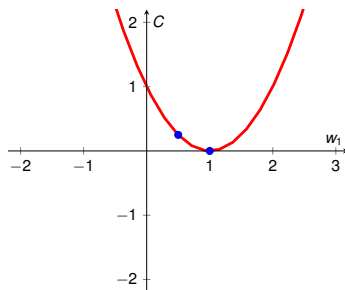


$$C(w_0, w_1) = \frac{1}{2m} \sum_{i=1}^m (h(x^{(i)}) - y^{(i)})^2$$

$$= \frac{1}{2 \times 6} \left((0.25 - 0.5)^2 + (0.5 - 1)^2 + (1 - 2)^2 + \dots + (1.5 - 3)^2 \right)$$
$$= 0.25$$

(4)

$$C(w_0, w_1)$$



Gradient Descent I

Gradient Descent

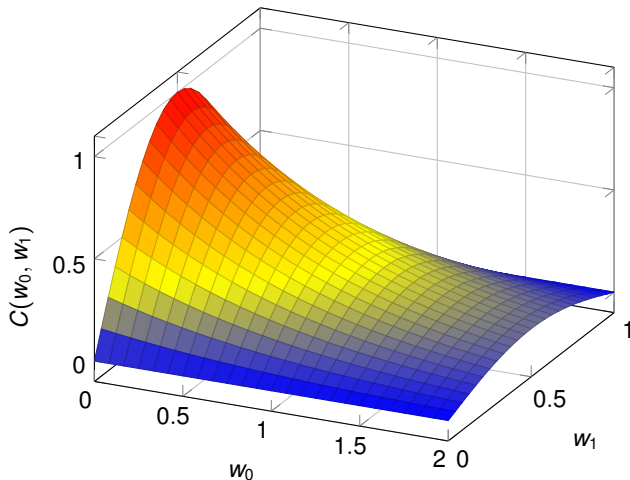
Fonksiyon: $C(w_0, w_1)$ veya $C(\mathbf{w}) = C(w_0, w_1, \dots, w_n)$

Amaç: $\min_{w_0, w_1} C(w_0, w_1)$

Özet:

- ▶ w_0, w_1 için herhangi bir değerler başlayın (Örnek: $w_0 = 0, w_1 = 0$)
- ▶ w_0, w_1 değerlerini değiştirerek, $C(w_0, w_1)$ maliyet fonksiyon değerini minimize etmeye çalışın.
- ▶ $C(w_0, w_1)$ 'un minimum değerini bulun.

Gradient Descent II



Gradient Descent III

Gradient descent algoritması herhangi bir değerle başlayıp, w değerlerinde her bir iterasyonda güncellemeler yapar.

değer yakınsaya kadar {
$$w_j = w_j - \alpha \frac{\partial}{\partial w_j} C(w_j)$$

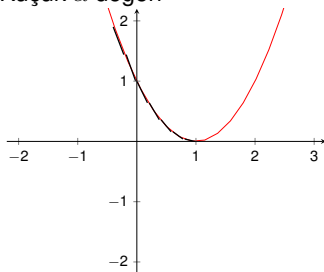
}

α : **Learning rate**
$$\frac{\partial}{\partial w_j} C(w_j)$$

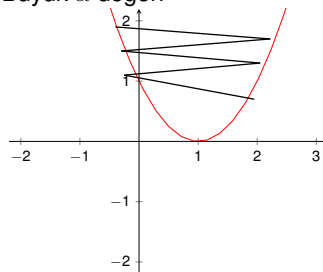
Gradient Descent IV

$$w_1 = w_1 - \alpha \frac{\partial}{\partial w_1} C(w_1)$$

Küçük α değeri



Büyük α değeri



Gradient Descent V

$$\frac{\partial}{\partial w_j} C(w_0, w_1) = \frac{\partial}{\partial w_j} \frac{1}{2m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right)^2$$

$$= \frac{\partial}{\partial w_j} \frac{1}{2m} \sum_{i=1}^m \left(w_0 + w_1 x^{(i)} - y^{(i)} \right)^2$$

Hatırlatma

$$(f \cdot g)' = f'g + g'f$$

$$j = 0 \Rightarrow \frac{\partial}{\partial w_0} C(w_0, w_1) = \frac{1}{m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right)$$

$$j = 1 \Rightarrow \frac{\partial}{\partial w_1} C(w_0, w_1) = \frac{1}{m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right) x^{(i)}$$

(5)

değer yakınsaya kadar {

$$w_0 = w_0 - \alpha \frac{1}{m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right)$$

$$w_1 = w_1 - \alpha \frac{1}{m} \sum_{i=1}^m \left(h(x^{(i)}) - y^{(i)} \right) \cdot x^{(i)}$$

}

Dogrusal Regresyon - Python I

```
import pandas as pd
import matplotlib.pyplot as plt

# veri kumesini oku
verikumesi = pd.read_csv("dsl.txt", delimiter="\t")

X = verikumesi.iloc[:, -1].values
y = verikumesi.iloc[:, 1].values

# veri kumesini egitim ve test olarak parcala
from sklearn.cross_validation import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)

# dogrusal regresyon modeli
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train, y_train)

# tahmin
y_pred = regressor.predict(X_test)

# veri gorsellestirme
plt.scatter(X_train, y_train, color='red')
plt.plot(X_test, y_pred, color='blue')
plt.show()
```

Dogrusal Regresyon - Python II

