Hafta 01 - Giriş

SİB 552 - Siber Güvenlik İçin Veri Madenciliği

Bilgisayar Mühendisliği
Siber Güvenlik Yüksek Lisans Programı

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

Gebze Teknik Üniversitesi
2019 - Bahar

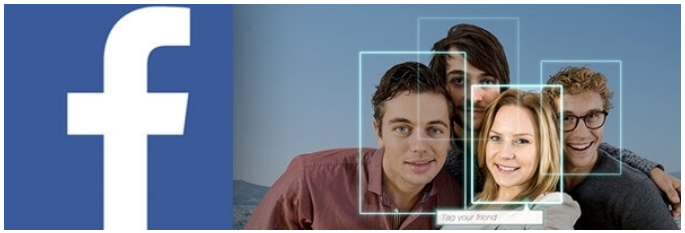
İçindekiler

İçindekiler

Makine Öğrenme Uygulamaları

Şekil: Self-driving cars

Şekil: Öneri Sistemleri



Şekil: Facebook face recognition

Şekil: human-brain-project

Makine Öğrenmesinin Geleceği

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
- *Machine Learning and Security*, Clarence Chio and David Freeman

Ders İçeriği

- Temel Makine Öğrenme Yöntemleri
- Makine Öğrenme Yöntemlerinin Siber Güvenlik Alanında Kullanımı
- Makine Öğrenmesinin Güvenli Hale Getirilmesi

Araçlar I

Python 3.5 (Öneri: 64-Bit)

- **Kütüphaneler**
  - scikit-learn (makine öğrenmesi)
  - keras (derin öğrenme)
  - tensorflow (derin öğrenme)
  - pandas (veri manipulasyonu)
  - matplotlib (veri göreselleştirme)
- **Geliştirme Ortamı**
  - Spyder (Önerilen - MATLAB benzeri ortam)
  - Pycharm (Alternatif)

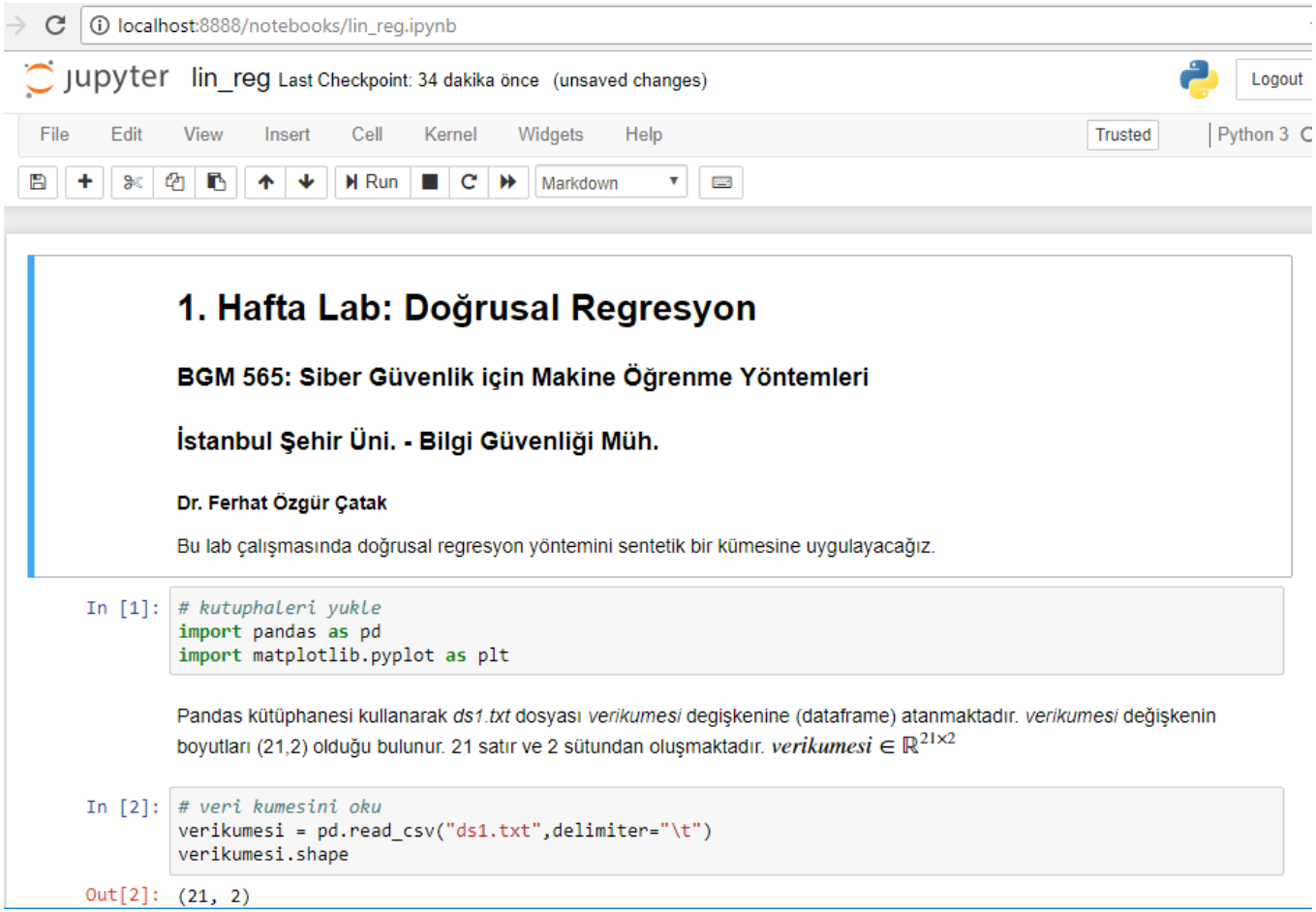
Python kütüphane kurulumu

```
pip install -U kutuphane_adi
```

Araçlar II

Lab Ortamı

- **Jupyter**: komut satırında `jupyter notebook`



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 yeterli olmadığı durumlar (Siber güvenlik, Bio-enformatik)
  - İnsan deneyiminin tam olarak açıklanamadığı durumlar (Ses tanıma, görüntü tanıma)
  - Çözümün zaman içinde değişmesi (saldırı türleri, polimorfik, metamorfik malware)

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
  - İstenmeyen e-posta 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

Makine Öğrenmesi ve Güvenlik I

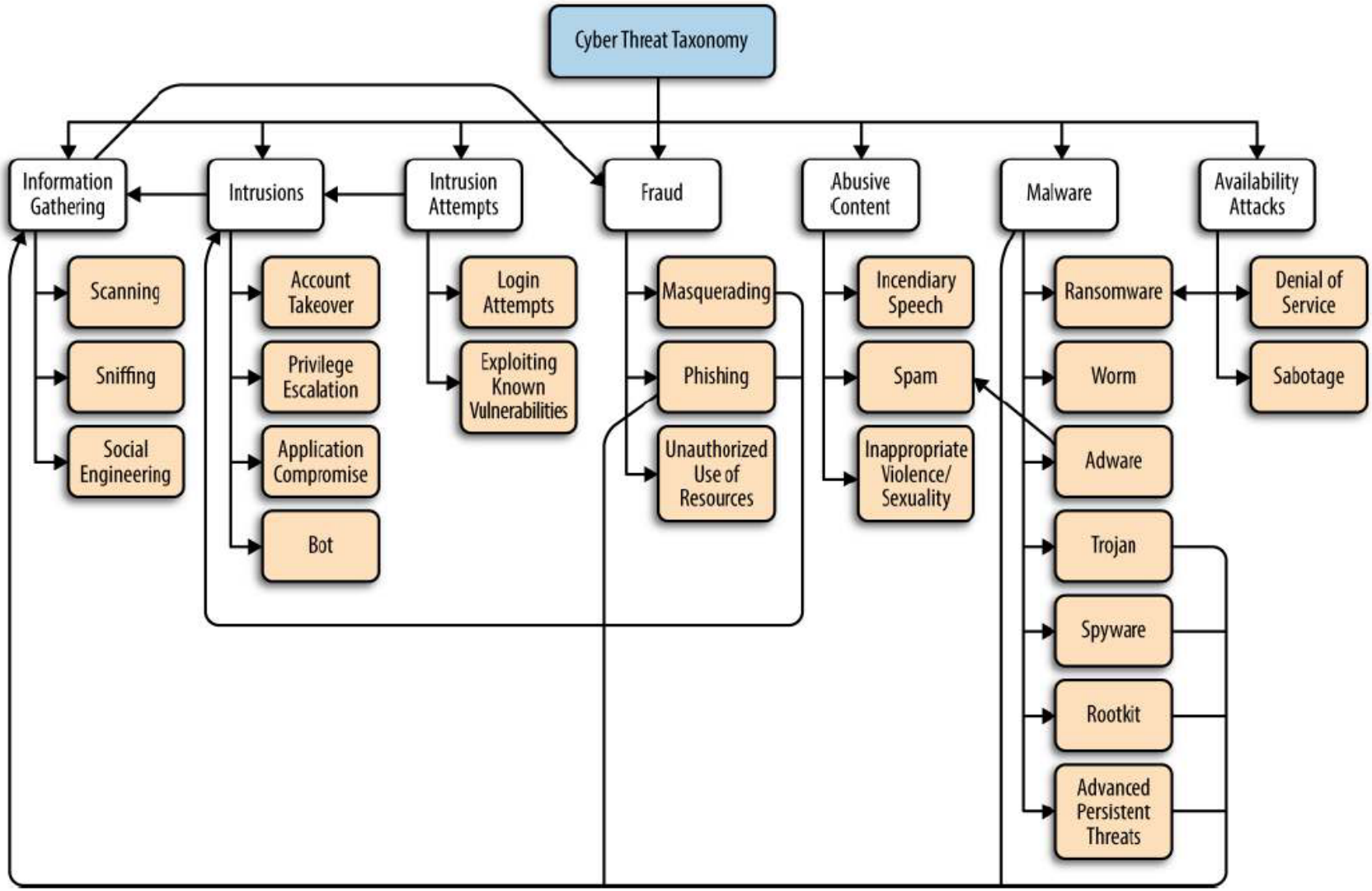
Makine Öğrenmesi ve Güvenlik

- Son yıllarda bilgisayar ve ağ güvenliği tehdit ve alanlar oldukça genişledi
  - intrusion detection
  - web application security
  - malware analysis
  - social network security
  - advanced persistent threats

Siber Güvenlik alanında Makine Öğrenme Kullanımı

- **Pattern Recognition**
  - Verilerde gizlenmiş açık veya gizli özelliklerin keşfedilmesi
  - Bu özellikler, aynı özellikleri gösteren verinin diğer formlarını tanıyan bir algoritmayı öğretmek için kullanılabilir.
  - Örnekler: Spam detection, Malware detection, botnet detection, **user authentication, behavior analysis**,
- **Anomaly Detection**
  - belirli bir veri kümesinin çoğunu (% 95'ten fazla) tanımlayan bir normallik kavramı oluşturmaktır.
  - **user authentication, behavior analysis**

Makine Öğrenmesi ve Güvenlik II



İçindekiler

Tanım I

Makine Öğrenmesi

- Paradigma değişikliği
- **Klasik yöntem (Klasik Programlama)**: Kurallar + Veri $\Rightarrow$ Cevaplar
- **Makine Öğrenmesi**: Veri + Cevaplar $\Rightarrow$ Kurallar
- Bir makine öğrenme sistemi **programlanmaz**, **eğitilir**.

Makine Öğrenmesi - Teori

- *Gözlem veri kümesi*: $\mathcal{X}$
- *Parametreler*: θ
- Öğrenme modeli: $f_\theta : \mathcal{X} \rightarrow \mathcal{Y}$
- *Öğrenme hatası*: $\mathbb{E}(f_\theta(\mathcal{X}), \mathcal{Y})$
  - Makine öğrenme algoritmaları hatayı minimize etmeye çalışırlar.
  - Tahmin edilen çıktı, $f_\theta(\mathcal{X})$, ile gözlemlenen veri kümesi $\mathcal{X}$ arasında bulunan fark.
- θ değiştirilerek $\mathbb{E}(f_\theta(\mathcal{X}), \mathcal{Y})$ minimize edilmeye çalışılır.

Tanım II

Notasyon

- Sayısal değerler (scalar) küçük harf: x, λ, η, örnek $x = 3.14$
- vektörler kalın harf: $\mathbf{x}$, örnek $\mathbf{x} = [1.2, 2.4, \cdots 0.2]$
- Bir vektörün i. elemanı: $\mathbf{x}^{(i)}$
- Girdi veri kümesi : $\mathcal{X} \in \mathbb{R}^{m \times n} \Rightarrow m$ adet satır (örnek, instance) ve n boyutlu uzay (nitelik sayısı, kolon sayısı)
- k boyutlu sınıf etiket kümesi: $C = \{C_1, \cdots C_k\}$
  - C : Ağ saldırıları: {DDoS, SQL Injection, XSS v.b.}
  - C : Zararlı Yazılım: {Trojan, Backdoor, Virus, Botnet v.b.}
- Girdi örnekleri ve hedef etiket çiftleri: $X = \{(\mathbf{x}_i, y_i) | \mathbf{x} \in \mathbb{R}^{m \times n}, y \in \{C_1, \cdots C_k\}\}_{i=1}^{n}$

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 veya daha fazla elemanlı küme, genellikle $\{0, 1\}$ veya $\{-1, +1\}$. $\mathcal{Y}$ olabilecek etiketler kümesini göstersin. Zararlı yazılım analizi için $\mathcal{Y} \in \{0, 1\}$. 0: Normal, 1: zararlı yazılım
  - **Eğitim verisi (Training data)**: $S = ((\mathbf{x}_1, y_1), \cdots (\mathbf{x}_m, y_m))$. Sonlu bir dizi çift $\mathcal{X} \times \mathcal{Y}$. Diğer adlandırmalar: *Eğitim veri kümesi, training set*
- **Öğrenmenin çıktıları**: Tahmin kuralı üretmesi istenir $h : \mathcal{X} \to \mathcal{Y}$. Hipotez, sınıflandırıcı, model. Tahmin: $h(\mathbf{x}) \to \hat{y}$. Gerçek sınıflandırıcı $f : \mathcal{X} \to \mathcal{Y}$, $f(\mathbf{x}) \to y$. $\hat{y}$: tahmin edilen sınıf (zararlı yazılım veya değil), y: gerçek sınıf

Formal Model II
İstatiksel 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.

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

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

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

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

Makine Öğrenme Problemlerinin Sınıflandırılması

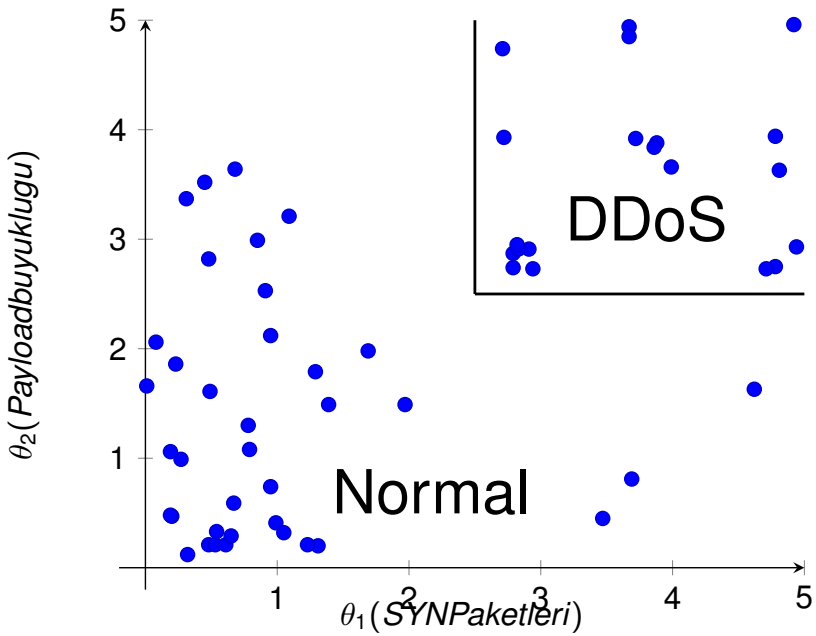
Öğrenme Tipleri

- **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 ödülü en üst düzeye çıkarma

İçindekiler

Sınıflandırma (Classification) I

Örnek: DDoS Saldırısı



- Bir sunucuya olan bağlantıların DDoS saldırı riski kararı için SYN paket sayısı ve Payload büyüklüğüne bakılması
- θ_1 : SYN Paket sayısı
- θ_2 : Payload büyüklüğü

```
if SYN > 25 and payload > 25:
    print("DDoS attack")
else:
    print("Normal traffic")
```

Sınıflandırma (Classification) II

Uygulama Alanları

- **Yüz tanıma (Face recognition)**: Işık, açı, gözlük, makyaj ve saç stilinden bağımsız olarak algılama
- **Karakter tanıma**: Farklı el yazısı tarzından bağımsız
- **Ses tanıma (Speech recognition)**: farklı ortamlarda seslerin tanınması
- **Hastalık Teşhisi**: semptomlardan hastalık tespit edilmesi
- **Biometrik**: Kişinin fiziksel veya davranış karakteristiklerinden tanıma: yüz, iris, imza gibi bilgilerden

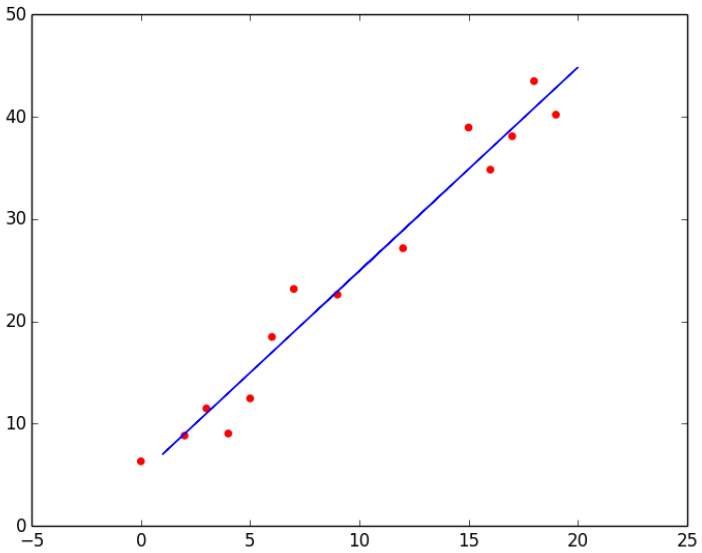
Danışmanlı Öğrenme Örnekleri

Bir kurumda siber güvenlik alanında çalıştığınızı kabul edin. Sizden bir problem için çeşitli algoritmalar kullanarak model oluşturulması istensin.

- **Problem 1**: Kullanılan kimlik doğrulama sistemi üzerinde bulunan her bir hesap için normal/ele geçirilmiş (compromised) olarak karar vermek istiyorsunuz.
  - **Sınıflandırma Problemi**

Regresyon (Regression) I

Örnek: Bir ağ içerisinde toplam paket sayısının İstemci sayısıyla 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

Sıralama (Ranking) I

Sıralama (Ranking)

- $\mathcal{X}$ öğelerinin belirli bir listesini sıralayan bir fonksiyon bulunması
- **Çift yönlü yaklaşım**
  - Sınıflandırma problemi olarak: $\{x_1, x_2\}$ doğru şekilde sıralı mı?
- **Noktasal yaklaşım**
  - Regresyon problemi olarak: $h(x) oyle\, ki\, x_1 < x_2 \Leftrightarrow f(x_1) < f(x_2)$
- **Liste bazında yaklaşım**
  - En iyi liste seçimi.

Kullanım alanları

- Öneri sistemleri
  - Kullanıcıların bereaber aldıkları ürünler
  - Kullanıcılara video önerileri (Youtube)
- Bilgi çıkarımı
  - Döküman
  - Arama motorları

İçindekiler

Danışmansız Öğrenme (Unsupervised Learning) I

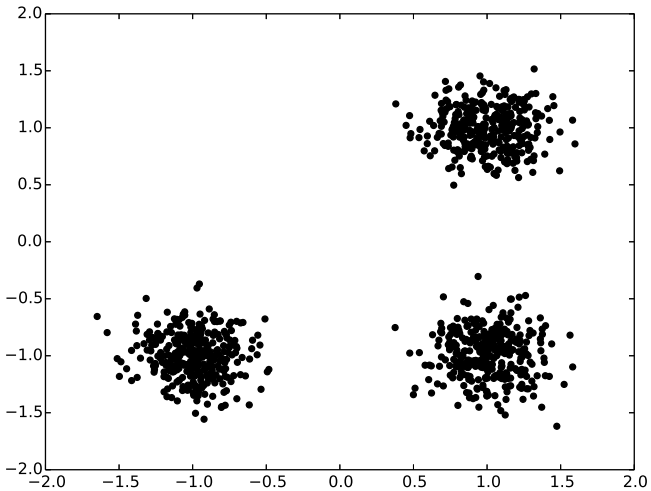
Danışmansız Öğrenme (Unsupervised Learning)

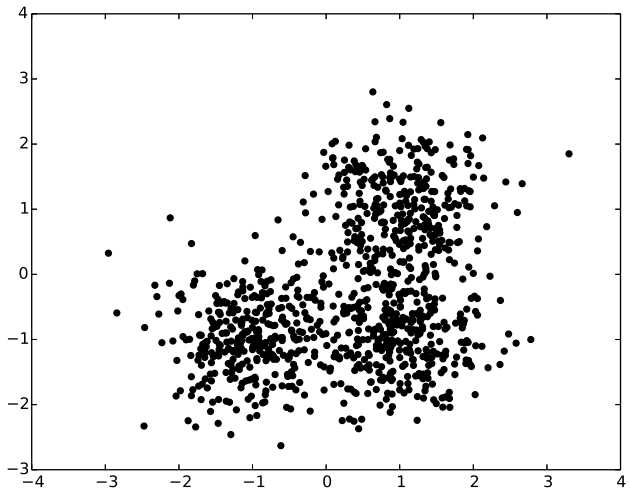
- **Eğitim veri kümesinin çıktı etiketleri mevcut değil**
- **Yoğunluk tahmini (Density estimation)**: Verideki yapının bulunması
- **Kümeleme (Clustering)**
- **Boyut azaltma (Dimensionality reduction)**

Kümeleme (Clustering) I

Kümeleme

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





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

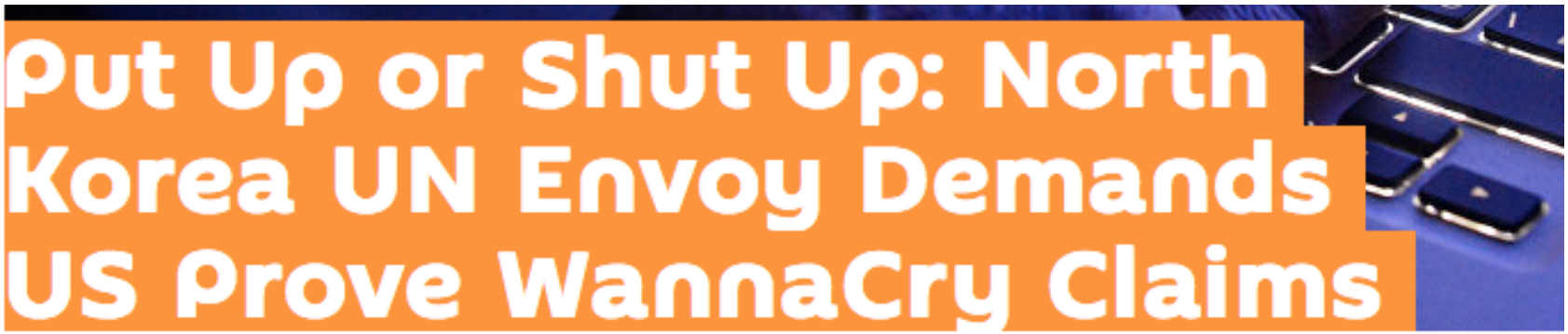

CSO Australia


Sputnik Intern...


FileHippo News

View all

Kümeleme (Clustering) III



ASIA & PACIFIC 02:21 27.12.2017 (updated 02:22 27.12.2017) Get short URL 8 26 0

Fed up with allegations by US Homeland Security Adviser Tom Bossert, the North Korean ambassador to the United Nations called on Washington late Monday to reveal evidence showing that Pyongyang was behind the WannaCry ransomware attack, as it claims.

Ambassador Pak Song Il, speaking to AP in a phone interview, told the outlet that Bossert's statement, which was published in the Wall Street Journal last Monday, was simply an an effort by the US to further create an "extremely confrontational atmosphere."

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

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

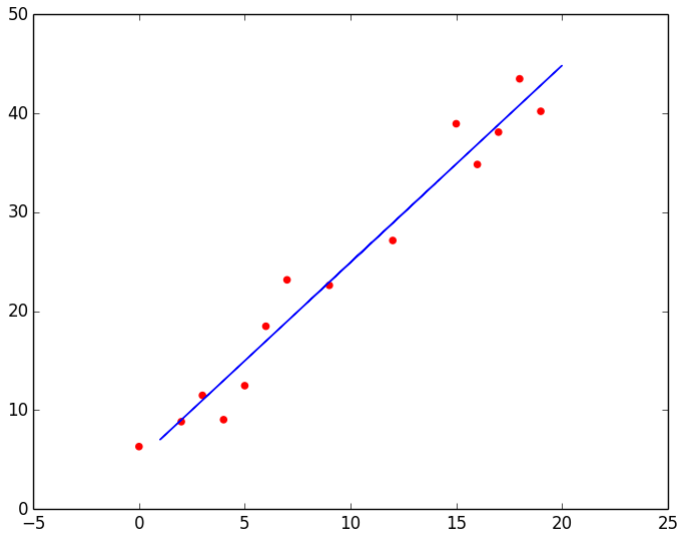
Tek Değişkenli Doğrusal Regresyon I

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
...	...

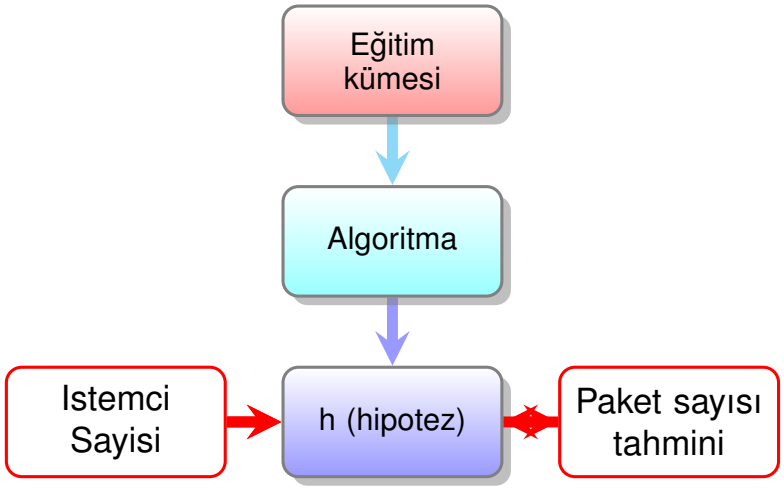
Notasyon: $X \in \mathbb{R}^{m \times n}$

m: örnek sayısı (satır sayısı)

x : girdi değişkeni/nitelikler

y : çıktı/hedef değişken

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

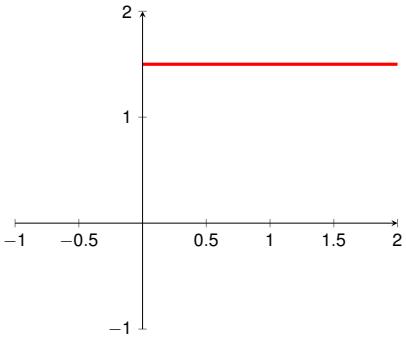


hipotez h gösterilim: $h(x) = w_0 + w_1 x$

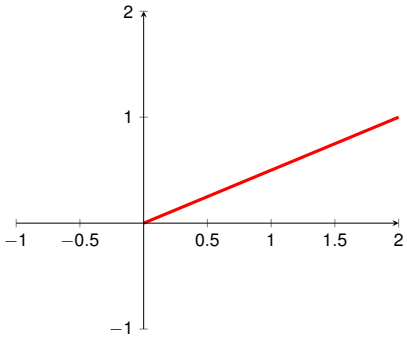
Maliyet fonksiyonu I
Cost function

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

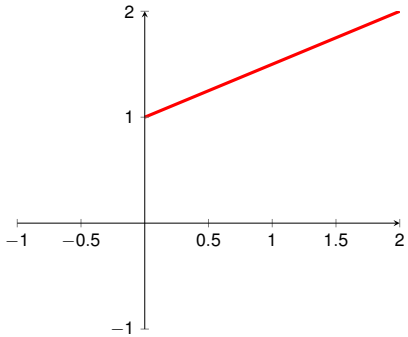
- Hipotez: $h_w = w_0 + w_1 x$
- w_i : parametreler
- w_i nasıl seçilecek?
- Hata: $\epsilon = y - h(\mathbf{x})$



Şekil: $h = 1.5 + 0x$

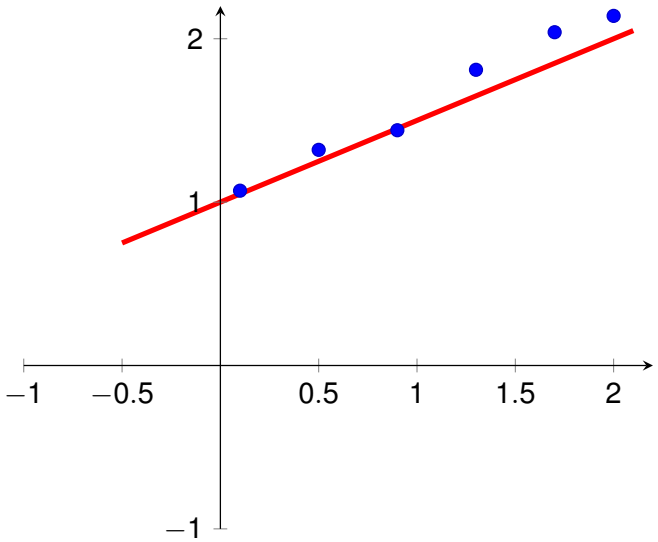


Şekil: $h = 0 + 0.5x$



Şekil: $h = 1 + 0.5x$

Maliyet fonksiyonu II
Cost function



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

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

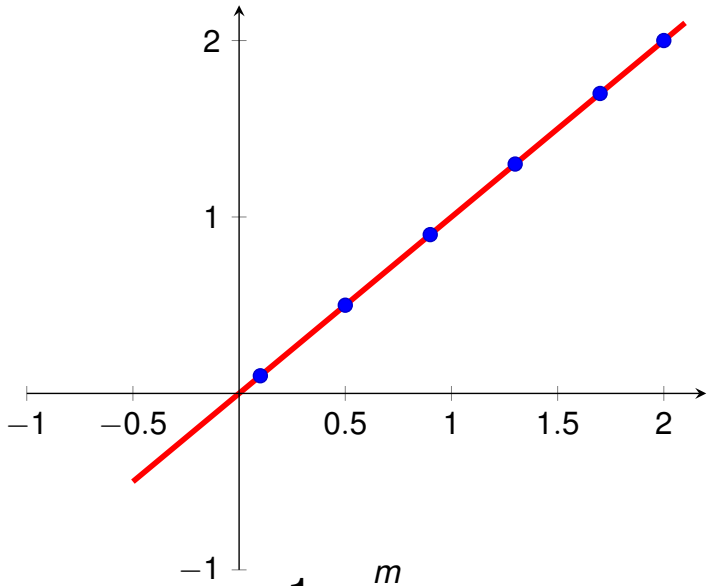
$$\underset{w_0,w_1}{minimize}\ J(w_0, w_1)$$

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

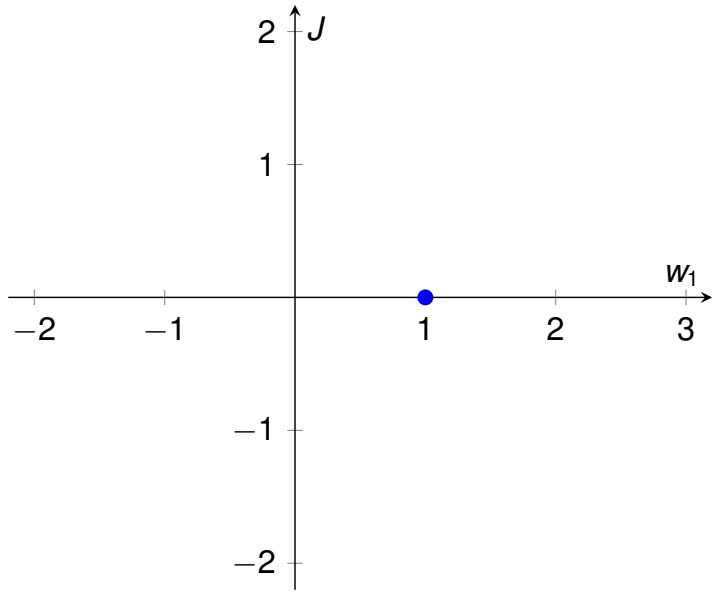
$J(w_0, w_1)$: Maliyet fonksiyonu (Cost function)

Maliyet fonksiyonu III
Cost function

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



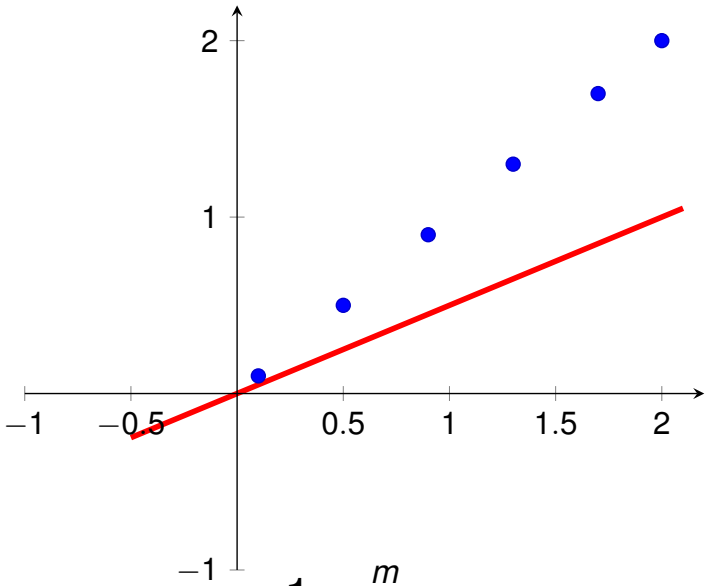
$$J(w_0, w_1)$$



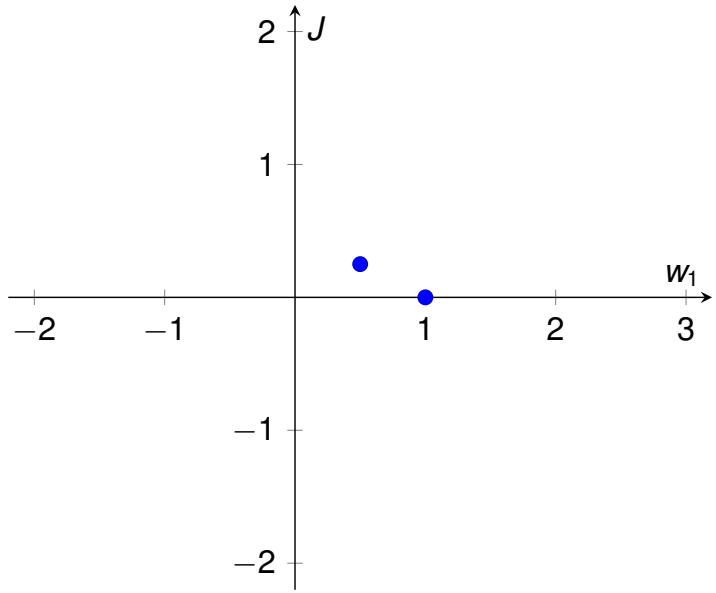
$$\begin{aligned} J(w_0, w_1) &= \frac{1}{2m} \sum_{i=1}^{m} \left(h(x^{(i)}) - y^{(i)} \right)^2 \\ &= \frac{1}{2m} \left(0^2 + 0^2 + \cdots + 0^2 \right) \\ &= 0^2 \end{aligned} \tag{2}$$

Maliyet fonksiyonu IV
Cost function

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



$$J(w_0, w_1)$$



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

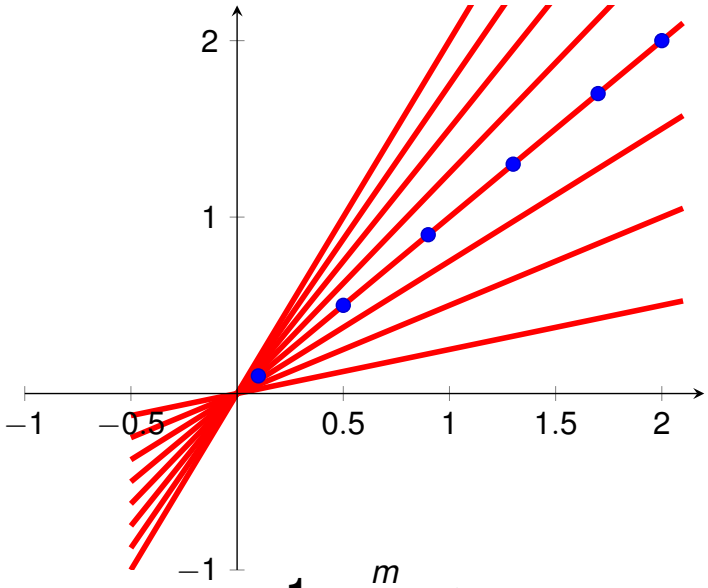
$$= \frac{1}{2 \times 6} \left((0.25 - 0.5)^2 + (0.5 - 1)^2 + (1 - 2)^2 + \cdots + (1.5 - 3)^2 \right)$$

$$= 0.25$$

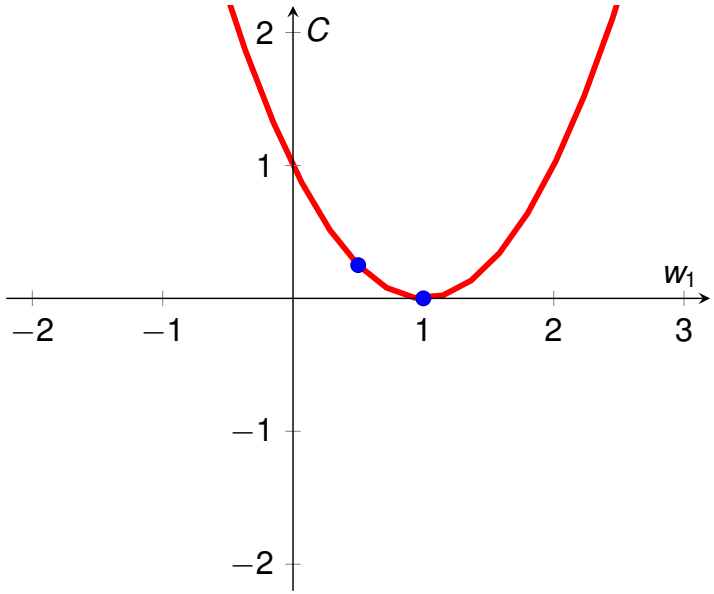
(3)

Maliyet fonksiyonu V
Cost function

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



$$J(w_0, w_1)$$



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

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

$$= 0.25$$

(4)

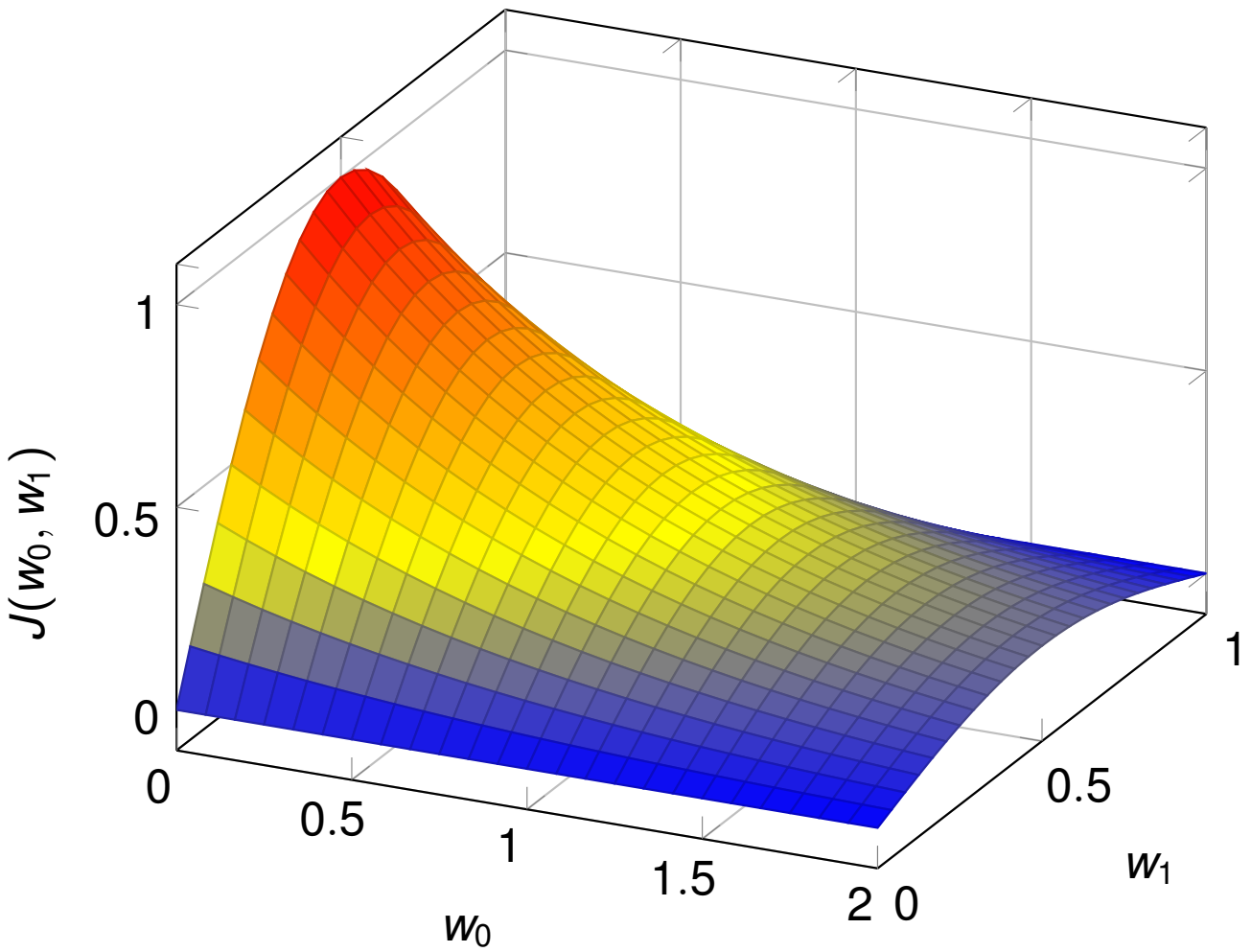
Gradient Descent I

Gradient Descent

Fonksiyon: $J(w_0, w_1)$ veya $J(\mathbf{w}) = J(w_0, w_1, \cdots w_n)$
Amaç: $\min_{w_0, w_1} J(w_0, w_1)$
Özet:

- w_0, w_1 için herhangi bir değerler başlayın (Örnek: $w_0 = 0, w_1 = 0$)
- w_0, w_1 değerlerini değiştirerek, $J(w_0, w_1)$ 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.

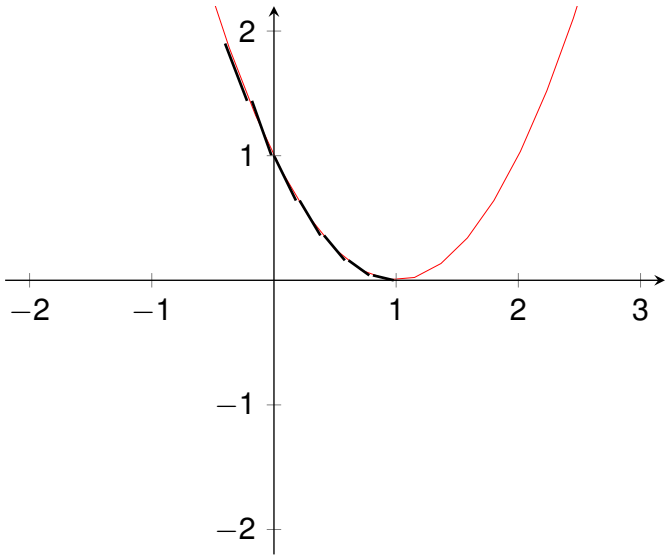
değer yakınsaya kadar {
$w_j = w_j - \alpha \frac{\partial}{\partial w_j} J(w_j)$
}

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

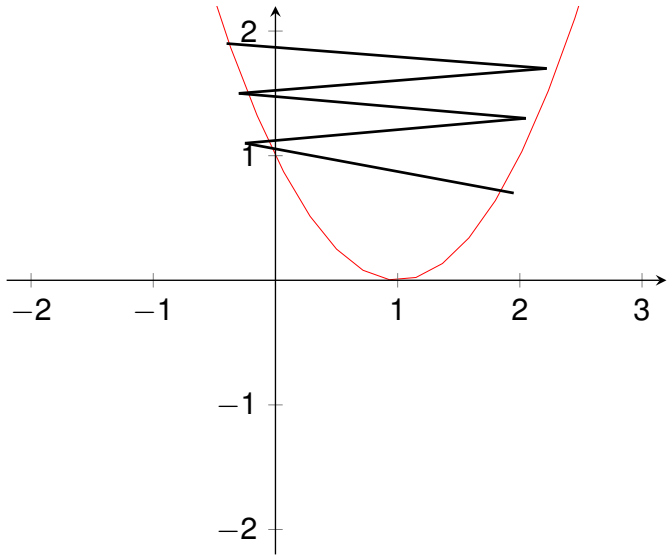
Gradient Descent IV

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

Küçük α değeri



Büyük α değeri



Gradient Descent V

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

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

Hatırlatma

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

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

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

(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("ds1.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