

MODÜL 3: İlk Yapay Zeka (Lineer Regresyon)

Artık soyut matematikten çıkıp **gerçek veri bilimine** giriyoruz. Şu ana kadar gibi uydurma fonksiyonları optimize ettik. Şimdi gerçek veriye en uygun doğruyu çizen bir model kodlayacağız.



Teori: Hipotez ve Hata

Bir ev fiyatı tahmin modelinde iki değişken vardır:

- **Girdi ()**: Evin metrekaresi.
- **Çıktı ()**: Evin fiyatı.

Amacımız ile arasındaki ilişkiyi kuran şu denklemini (Hipotez) bulmaktır:

$$h(x) = w \cdot x + b$$

- **(Ağırlık/Eğim)**: Metrekarenin fiyata etkisi.
- **(Bias)**: Sabit başlangıç fiyatı.

Bilgisayar başlangıçta w ve b 'yi rastgele sallar. Sonra senin az önce yazdığın **Gradient Descent** ile hatayı azaltarak en doğru w ve b 'yi bulur.



Hata Fonksiyonu: MSE (Mean Squared Error)

Dağcı örneğinde hatamız idi. Gerçek hayatta hata şudur: **(Gerçek Fiyat - Tahmin Edilen Fiyat)**

Buna **MSE** denir ve formülü şöyledir:

$$Loss = \frac{1}{N} \sum (y_{gerçek} - (w \cdot x + b))^2$$

Karesini alıyoruz ki negatif hatalar pozitif dönsün ve büyük hatalar daha çok cezalandırılsın).



Sıradaki Adım: Modeli Eğitmek (Gradient Descent)

Hatayı hesapladık (çok yüksekti). Şimdi o hatayı kullanarak w ve b değerlerini düzelteceğiz.

Matematiksel Görev: Loss fonksiyonumuzun (MSE) türevini almamız lazım.

Zincir kuralı (Chain Rule) ile türev aldığımızda (buraları senin için sadeleştiriyorum) bilgisayarın kullanacağı güncelleme formülleri şunlardır:

1. **w'nin türevi (dw):** $\frac{-2}{N} \sum x \cdot (y - tahmin)$
2. **b'nin türevi (db):** $\frac{-2}{N} \sum (y - tahmin)$

Bu formüller bize "Hangi yöne gitmeliyiz?" sorusunun cevabını verir.

$$Loss = \frac{1}{N} \sum (y - (wx + b))^2$$

Zincir Kuralı (Chain Rule) Mantığı

Bir soğan düşün. Cücüğe (w) ulaşmak için katmanları dışarıdan içeriye doğru soyman gerekir. Zincir kuralı şudur: "**En dıştaki hatanın türevini al, içeriye doğru çarparak ilerle.**"

Bizim yapımız şöyle:

1. **Girdi (w):** Değiştirdiğimiz şey.
2. **Ara Katman (y_tahmin):** değişince bu değişir ().
3. **Çıktı (loss):** Tahmin değişince hata değişir ().

Formül:

$$\frac{\partial Loss}{\partial w} = \frac{\partial Loss}{\partial y_{tahmin}} \cdot \frac{\partial y_{tahmin}}{\partial w}$$

Adım Adım İspat

Basitlik olsun diye toplam sembolünü (\sum) ve ortalamayı ($\frac{1}{N}$) şimdilik kenara koyalım, **tek bir veri noktası** için yapalım. Sonra onları geri ekleriz.

$$1. \text{ Tahmin: } y_{pred} = w \cdot x + b$$

$$2. \text{ Hata (Loss): } L = (y_{gerçek} - y_{pred})^2$$

Amacımız: $\frac{\partial L}{\partial w}$ (Loss'un w 'ye göre değişimi).

ADIM 1: Dış Katman (Hatanın Tahmine Göre Türevi)

Hata fonksiyonumuz bir kare alma işlemidir: u^2 . Türev kuralı: $(u^2)' = 2 \cdot u \cdot u'$.

Burada $u = (y_{gerçek} - y_{pred})$. Zincir kuralını uygulayalım:

$$\frac{\partial L}{\partial y_{pred}} = 2 \cdot (y_{gerçek} - y_{pred}) \cdot (-1)$$

Neden -1? Çünkü parantezin içindeki y_{pred} 'in önünde eksi işareti var. $y_{gerçek}$ sabit bir sayıdır, türevi 0'dır.

$$\text{Sonuç 1: } -2(y_{gerçek} - y_{pred})$$

ADIM 2: İç Katman (Tahminin Ağırlığa Göre Türevi)

Şimdi tahmin fonksiyonuna bakalım: $y_{pred} = w \cdot x + b$

Bunun w 'ye göre türevi nedir? w 'nin katsayısı x 'tir. (b sabittir, türevi 0 olur).

Sonuç 2: x

ADIM 3: Birleştir (Çarpım)

Zincir kuralı gereği Adım 1 ve Adım 2'yi çarpıyoruz:

$$\frac{\partial L}{\partial w} = \text{Sonuç 1} \cdot \text{Sonuç 2}$$

$$\frac{\partial L}{\partial w} = [-2(y_{gerçek} - y_{pred})] \cdot x$$

Düzenlersek:

$$\frac{\partial L}{\partial w} = -2 \cdot x \cdot (y_{gerçek} - y_{pred})$$

ADIM 4: Toplam Sembolünü Geri Getir

Biz bunu tek bir veri için yaptık. MSE (Mean Squared Error) tüm verilerin **ortalamasıdır**. Yani formülün başına ekleriz.

FİNAL FORMÜL (dw):

Kodda yazdığımız hali:

Python

```
# hata_farki = (Y - tahminler)
dw = (-2 / N) * np.sum(X * hata_farki)
```

🧠 Bias (db) İçin Durum Ne?

Mantık birebir aynıdır, sadece Adım 2 değişir. $y_{pred} = w.x + b$

Bunun b'ye göre türevi nedir? b'nin önünde gizli bir 1 vardır. Türevi **1**'dir.

Bu yüzden db formülünde sonda x yoktur (çünkü 1 ile çarpılmıştır):

$$db = \frac{-2}{N} \sum (y_{gerçek} - y_{tahmin})$$

Bu matematiksel altyapı içine sındıysa, şimdi o kod satırlarını çok daha büyük bir güvenle doldurabilirsin.

Önceki koda dönüp w ve b güncelleme satırlarını tamamlayalım mı? Hatırlatma: $yeni = eski - (lr * türev)$