Temel Kavramlar

(İktisatçılar İçin) Makine Öğrenmesi (TEK-ES 2020)

Hüseyin Taştan
Yıldız Teknik Üniversitesi

Plan

- Makine Öğrenmesi Problemi
- Öğrenme türleri: Gözetimli Öğrenme vs. Gözetimsiz Öğrenme
- Gözetimli Öğrenme: Regresyon problemleri
- Gözetimli Öğrenme: Sınıflandırma problemleri
- İndirgenemez Hata
- Aşırı Uyum (overfitting)
- Eğitim ve Test verileri
- Sapma-Varyans ilişkisi

Makine Öğrenmesi

Makine Öğrenmesi (ML) bir girdi değişkenleri kümesinden hareketle çıktının kestirilmesi (tahmini) için (istatistiksel) modeller geliştirir.

Genel çerçeve:

- Y_i: Çıktı değişkeni
- $\mathbf{X}_i = \{X_{i1}, X_{i2}, \ldots, X_{ip}\}$: kestirim değişkenleri ya da öznitelikler (features),
- Kestirim modeli:

$$Y_i = f(\mathbf{X}_i) + \epsilon_i, \quad i = 1, 2, \ldots, n$$

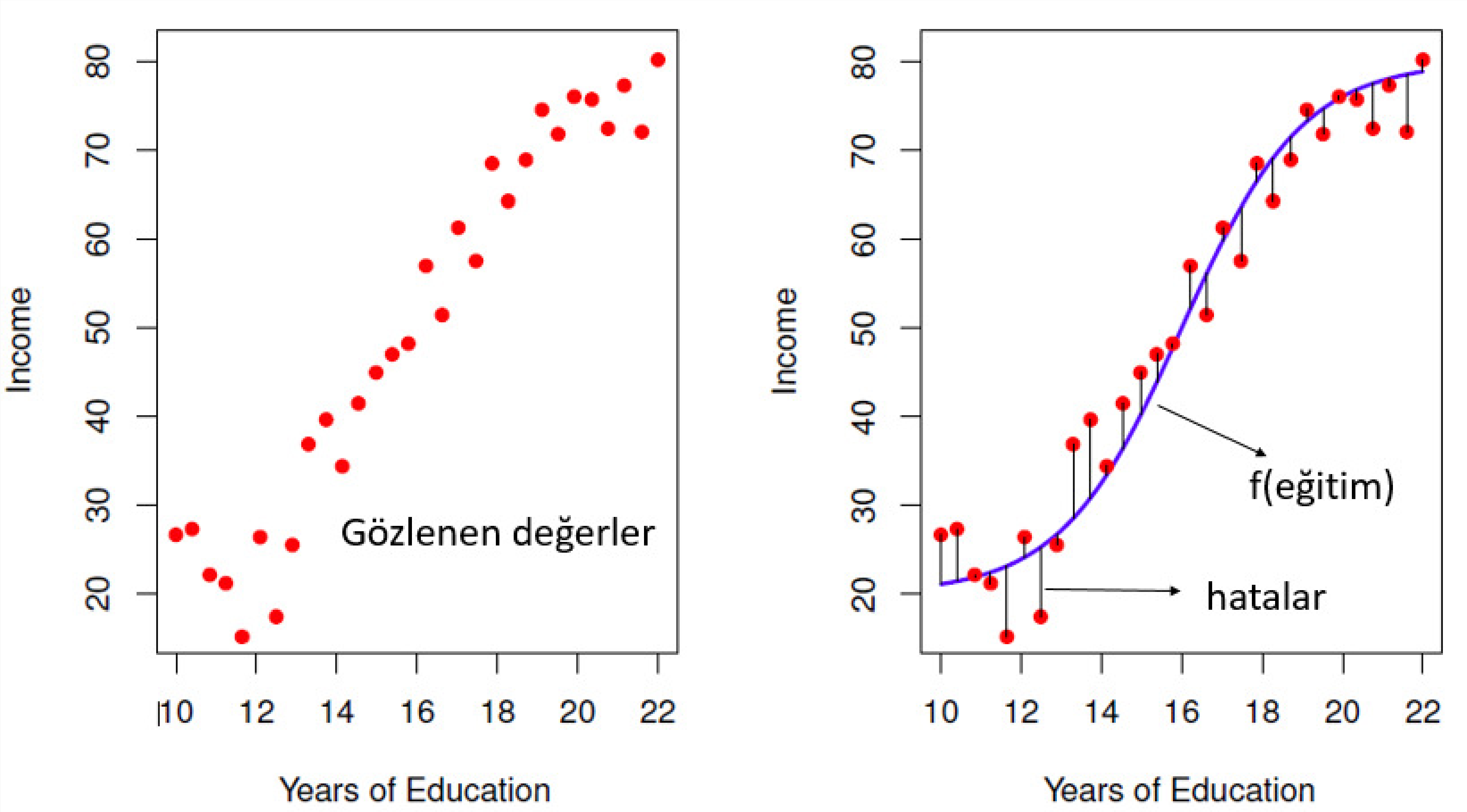
Burada $f(\mathbf{X}_i)$ bilinmeyen bir fonksiyon, ϵ gözlenemeyen bir rassal hata terimidir.

Gözetimli vs. Gözetimsiz Öğrenme

- Gözetimli (supervised) öğrenme: Çıktı değişkeni Y_i gözlemleniyor.
- Y_i sürekli değerler alıyorsa: **regresyon** problemi. Örneğin, evlerin özniteliklerinden hareketle değerinin tahmin edilmesi, borsa endeksinin yarınki kapanış değerinin öngörülmesi, vb.
- Y_i kategorik değişkense: **sınıflandırma** problemi. Örneğin, bir döviz kurunun yarınki hareketinin (aşağı ya da yukarı) öngörülmesi, bir kredi başvurusunun sınıflandırılması, vb.
- Gözetimsiz öğrenme: Verilerde Y_i yok ya da gözlenemiyor. Amaç özniteliklerden hareketle gözlemlerin öbeklenmesi ya da öznitelik boyutunun küçültülmesi olabilir. Örnek: müşterilerin özelliklerinden hareketle piyasa segmentasyonu yapılması.

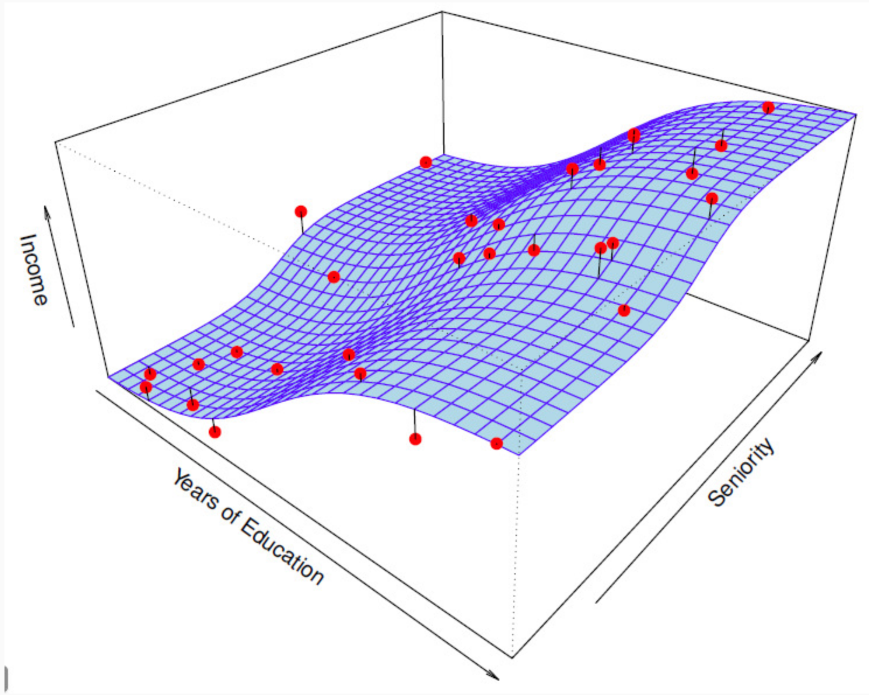
Regresyon Problemi Örnek: Gelir ve eğitim düzeyi

$$Income = \underbrace{\beta_0 + \beta_1 Education}_{f(x)} + \epsilon$$



Örnek: Eğitim ve yaşın bir fonksiyonu olarak gelir

$$income = f(education, seniority) + \epsilon$$



- Kırmızı noktalar: gözlenen gelir düzeyleri
- Mavi yüzey: pratikte genelde bilinmeyen $f(\cdot)$ fonksiyonu.
- Bu örnekte veriler simülasyonla elde edildiği için f tam olarak biliniyor

İndirgenebilir ve İndirgenemez Hata

- Bilinmeyen $f(X)$ fonksiyonunun tahminine $\hat{f}(X)$ diyelim. Bunun sonucunda elde edeceğimiz tahmin $\hat{Y} = \hat{f}(X)$

- Bu $\hat{f}(X)$'nın tahmininde ortaya çıkan hataya **indirgenebilir hata** denir. Bu hatanın azaltılması ancak uygun kestirim fonksiyonunun bulunmasıyla mümkündür.

- $f(X)$ pratikte bilinmez. Ancak bunu bilsek ve kestirimi buna göre oluştursak $\hat{Y} = f(X)$ bile bu kestirim hata içerecektir.

- $\epsilon = Y - f(X)$: indirgenemez hata (irreducible error). Bu hata X değişkenleri kullanılarak tahmin edilemez.

- Toplam değişkenlik iki parçaya ayrılabilir:

$$E[(Y - \hat{f}(X))^2 | X = x] = (f(x) - \hat{f}(x))^2 + Var(\epsilon)$$

Burada $Var(\epsilon)$ indirgenemez hatanın varyansıdır.

Kestirim modelinin tahmini

- n gözlemden oluşan bir eğitim (training) veri setimiz olsun.
- Öznitelikler: $\mathbf{X}_i = \{X_{i1}, X_{i2}, \ldots, X_{ip}\}, i = 1, 2, \ldots, n$
- Çıktı değerleri: $Y_1, Y_2, \ldots, Y_n$
- Amaç eğitim verilerinden hareketle $f(X)$ kestirim modelinin (kara kutu?) tahmini.
- Kullanabileceğimiz istatistiksel öğrenme yöntemleri iki ana gruba ayrılabilir:
  - **Parametrik Yöntemler**: modelin formuna ilişkin bir varsayım gerektirir (doğrusal, karesel, kübik, vb)
  - **Parametrik olmayan yöntemler**: f'in fonksiyonel kalıbına ilişkin bir varsayım yapılmaz. Kestirimlerin gözlemlenen değerlere mümkün olduğunca yaklaştırılması amaçlanır

Parametrik Yöntemler

Kabaca iki adımdan oluşur:

1.ADIM: Kestirim fonksiyonu f( )’in şekline ilişkin varsayım yapmamız gerekir, örneğin doğrusal, karesel, kübik, vs.

Örneğin: Lineer kalıp

$$f_L(X) = \beta_0 + \beta_1 X + \epsilon$$

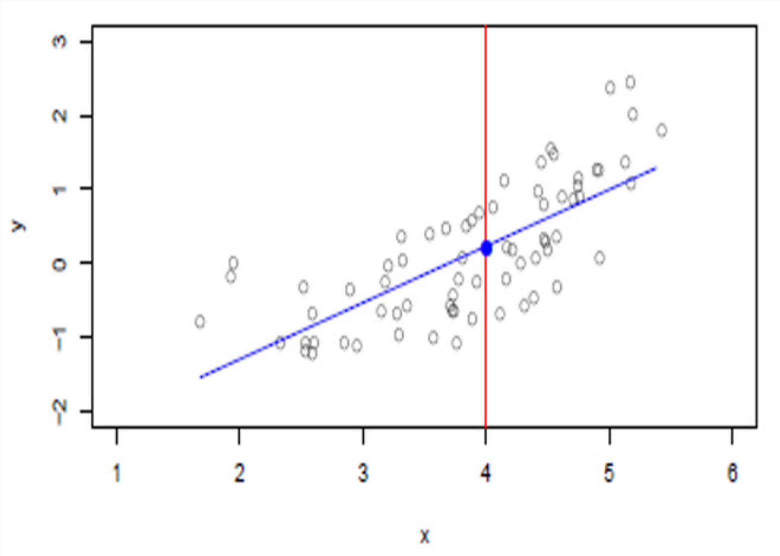
Karesel Kalıp:

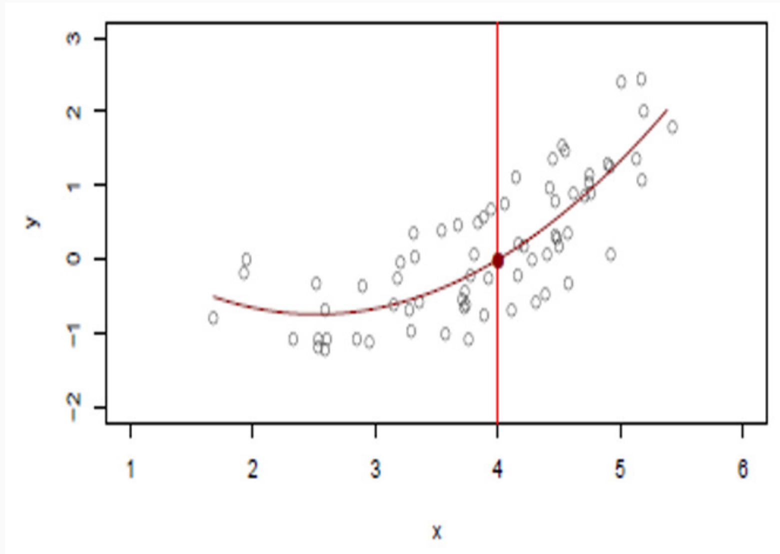
$$f_Q(X) = \beta_0 + \beta_1 X + \beta_2 X^2 + \epsilon$$

2.ADIM: Eğitim verilerinden hareketle modelin tahmini (eğitimi, uyumu) için bir yöntemin uygulanması. Örneğin, doğrusal regresyon modeli için sıradan en küçük kareler yöntemini kullanabiliriz.

Örnek: Doğrusal ve Karesel Modeller

$$\hat{f}_L(X) = \hat{\beta}_0 + \hat{\beta}_1 X \qquad \hat{f}_Q(X) = \hat{\beta}_0 + \hat{\beta}_1 X + \hat{\beta}_2 X^2$$



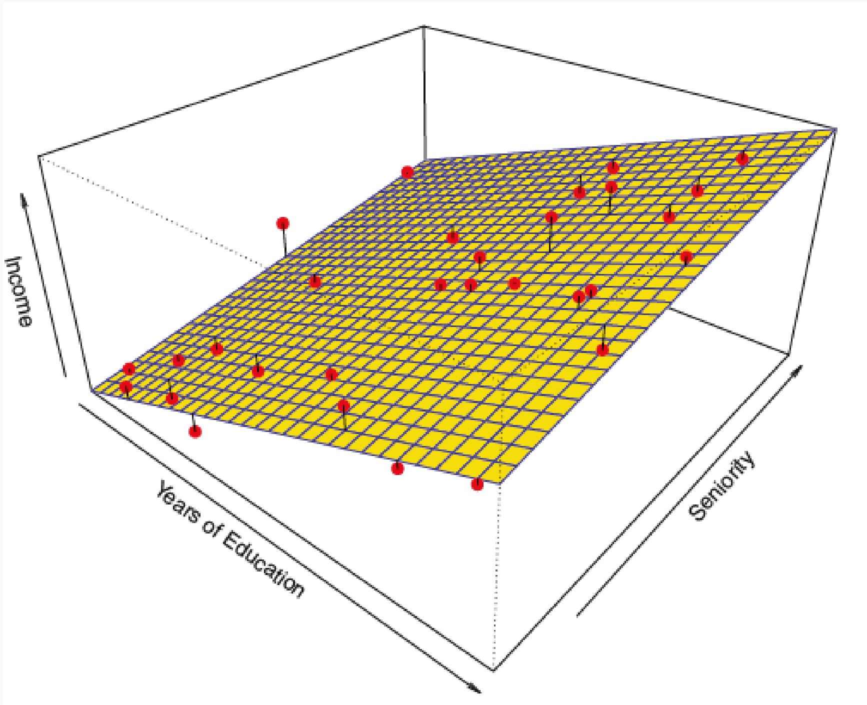


Doğrusal kestirim (solda) aşağı yukarı kabul edilebilir bir yaklaştırım sunsa da karesel model daha başarılı görünmektedir.

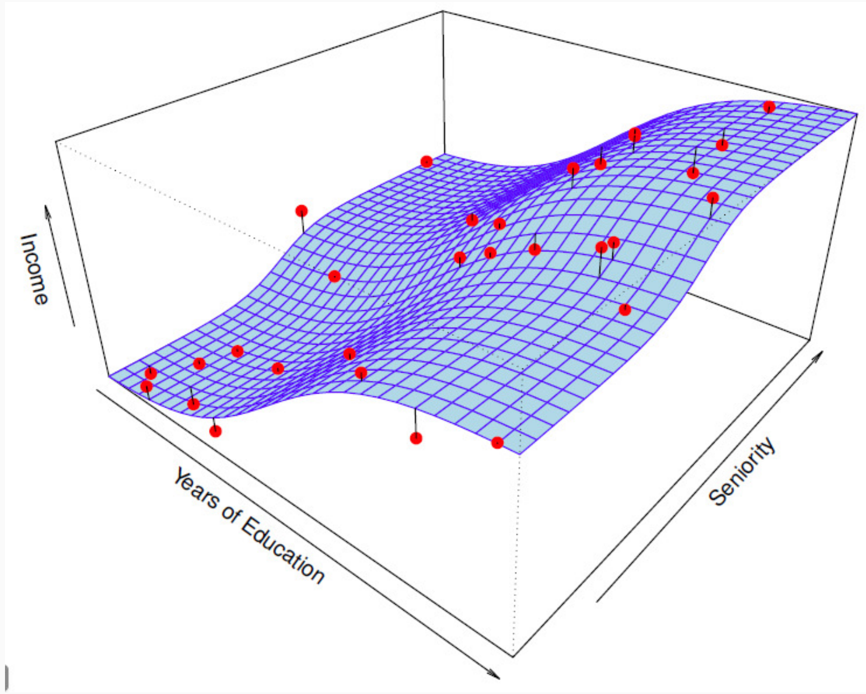
Örnek: Doğrusal Regresyon Tahmini

$$\hat{f}_L(eğitim, yaş) = \hat{\beta}_0 + \hat{\beta}_1 eğitim + \hat{\beta}_2 yaş$$

Kestirim:



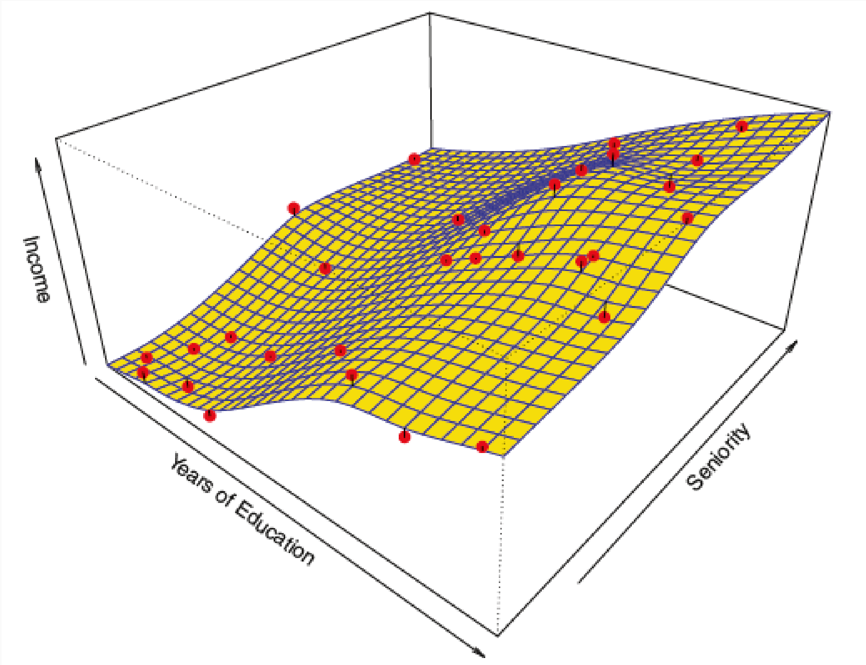
Gerçek ilişki:



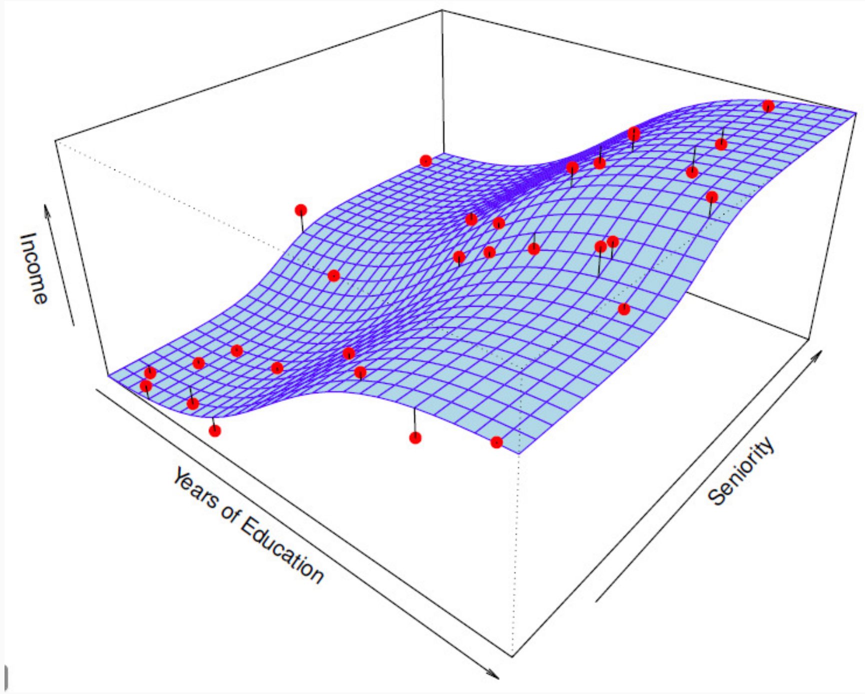
Parametrik Olmayan Modeller

Avantaj: esneklik ve kesinlik düzeyi yüksek tahmin

Kestirim:



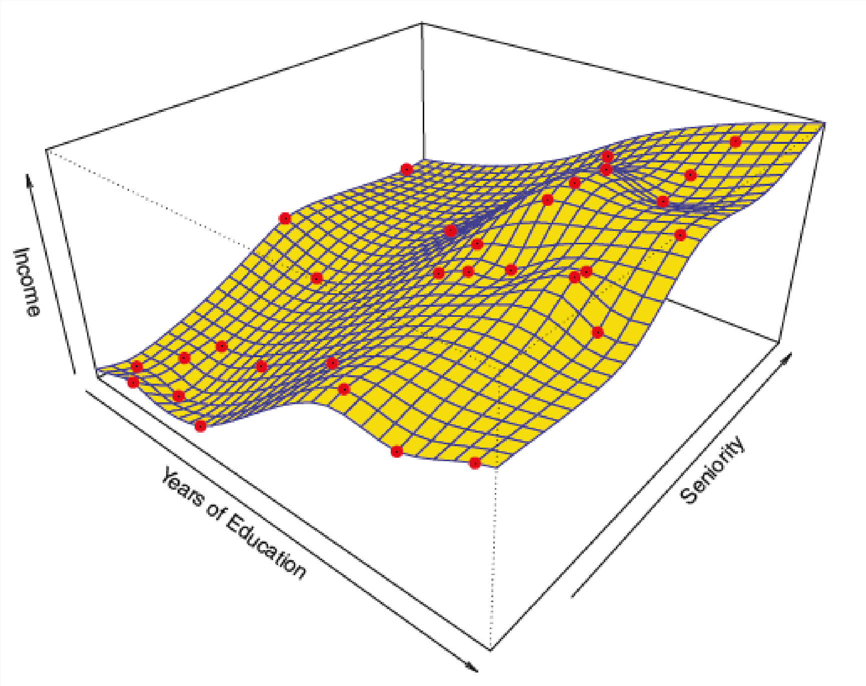
Gerçek ilişki:



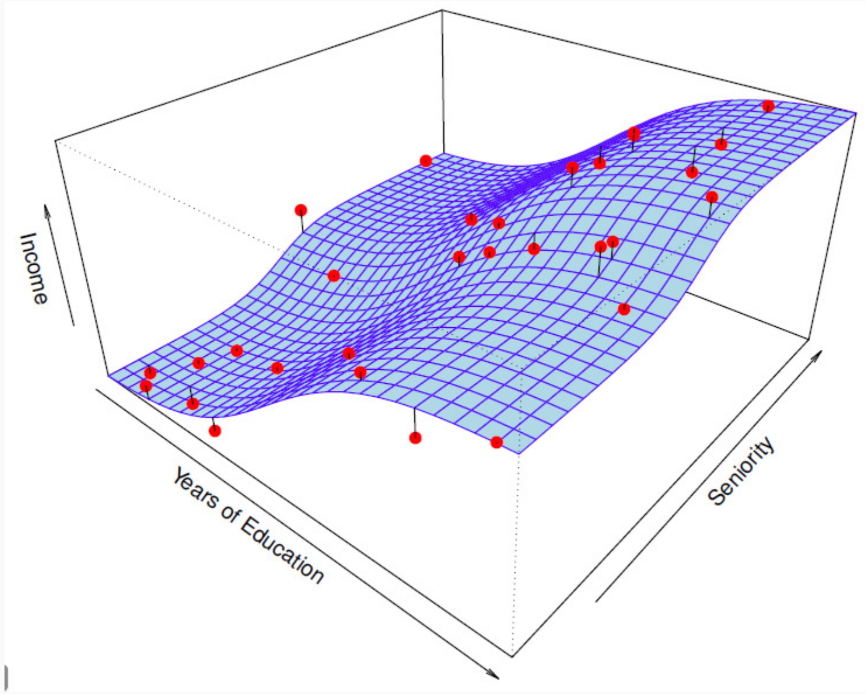
Tehlike: Fazla uyum (over-fitting)

Over-fitting (Aşırı uyum)

Kestirim:



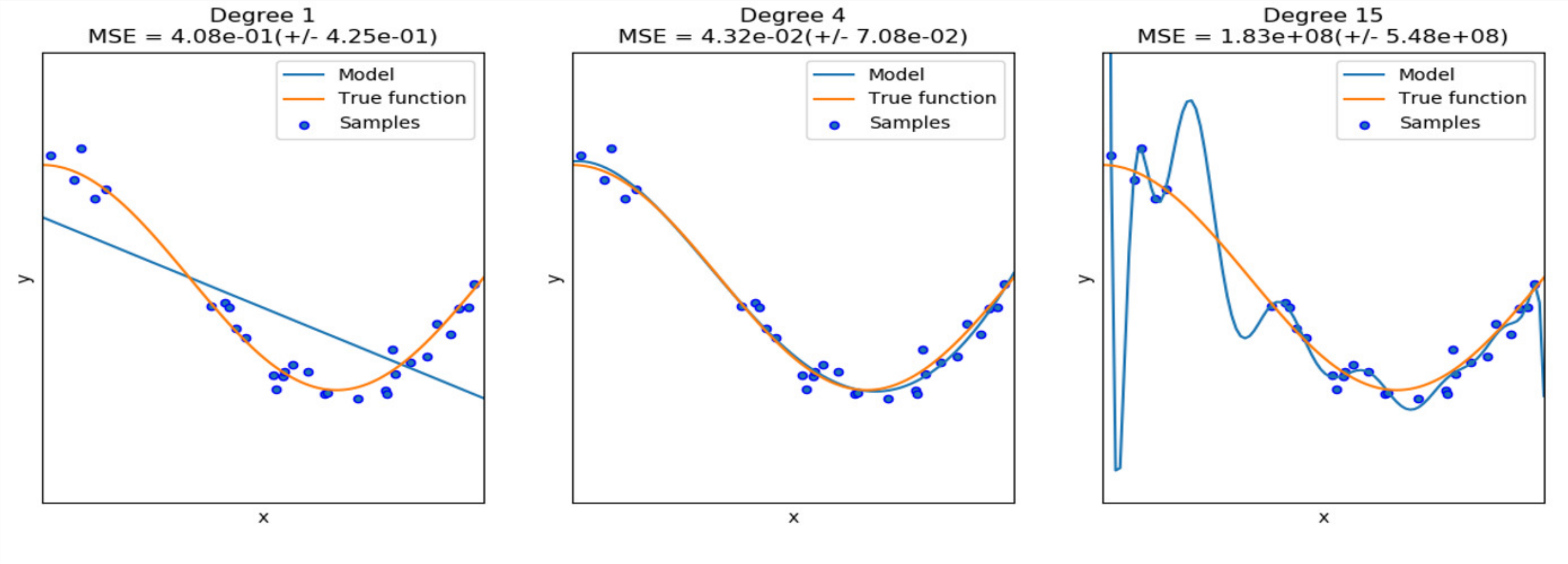
Gerçek ilişki:



Düzgünleştirme parametresini azaltarak mükemmel uyum sağladık. Ama bu kestirimlerin başarılı olacağını garanti etmez. Aslında verilerdeki gürültüyü (noise) de modelledik.

Aşırı uyum: iki boyutlu örnek

Bu grafikte, sırasıyla, lineer model, 4ncü derece polinom, ve 15nci derece polinom tahminleri gösteriliyor. Aşırı uyumun en önemli göstergesi tahminlerin hızlı hareket ederek zikzaklar çizmesidir.



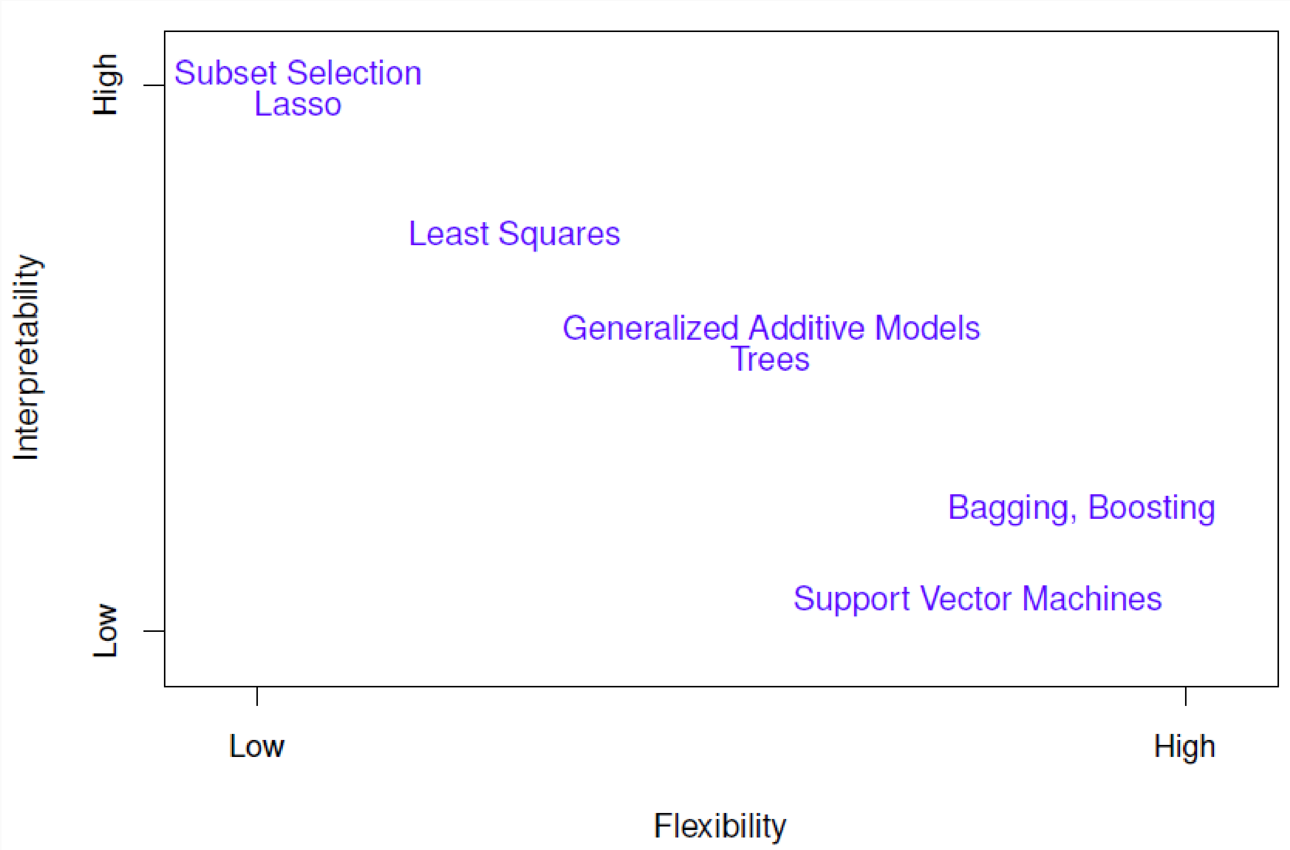
Aşırı Uyum

Doğruluk vs. Yorumlanabilirlik

- Kısıtlayıcı bir model yerine neden daha esnek bir tahmin modeli kullanmıyoruz?
- Bunun başlıca iki nedeni vardır:

1. Doğrusal regresyon gibi kısıtlayıcı varsayımlara dayanan modellerde yorumlama ve istatistiksel çıkarsama çok daha kolaydır. Örneğin, regresyon modelinde β_j katsayıları X_j'nin çıktı üzerindeki marjinal etkisini ölçer.
2. Modelin yorumlanabilirliği ikinci planda olsa bile, fonksiyon kalıbı esnek olmayan modeller daha yüksek öndeyi (kestirim, prediction) başarısına sahip olabilirler.

Esneklik ve Yorumlanabilirlik arasındaki ödünüm



(Kaynak: James et al., An Introduction to Statistical Learning, Figure 2.7, s. 25)

Modelin Doğruluğu Nasıl Ölçülür

- Tahmin doğruluğu (accuracy) tipik olarak Ortalama Hata Karesi (Mean Squared Error - MSE) ile ölçülür
- Modelin $y = f(x) + \epsilon$ olduğunu, tahminin ise $\hat{f}(x)$ ile gösterildiğini varsayalım.
- Böyle bir regresyon problemi için Ortalama Hata Karesi (MSE) aşağıdaki gibi tanımlanabilir:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{f}(x_i))^2$$

- Burada n gözlemden oluşan bir **eğitim** (training) veri seti kullanılmıştır.

MSE iyi bir ölçüt mü?

- Tipik olarak bir gözetimli öğrenme probleminde eğitim verisinde MSE en küçük olacak şekilde tahmin yapılır. Örnek: Sıradan En Küçük Kareler tahmininde kalıntı kareleri toplamını minimum yapan katsayı tahminleri bulunur.

- Bir makine öğrenmesi uygulamasında asıl amaç eğitim verisinde modelin performansının ne olduğu değildir. Önemli olan tahminde (eğitimde) kullanılmamış yeni bir veri setinde nasıl performans gösterdiğidir.

- Eğitimde kullanılmayan, sadece kestirim performansının (doğruluğunun) değerlendirilmesinde kullanılan veri setine **test** verileri denir.

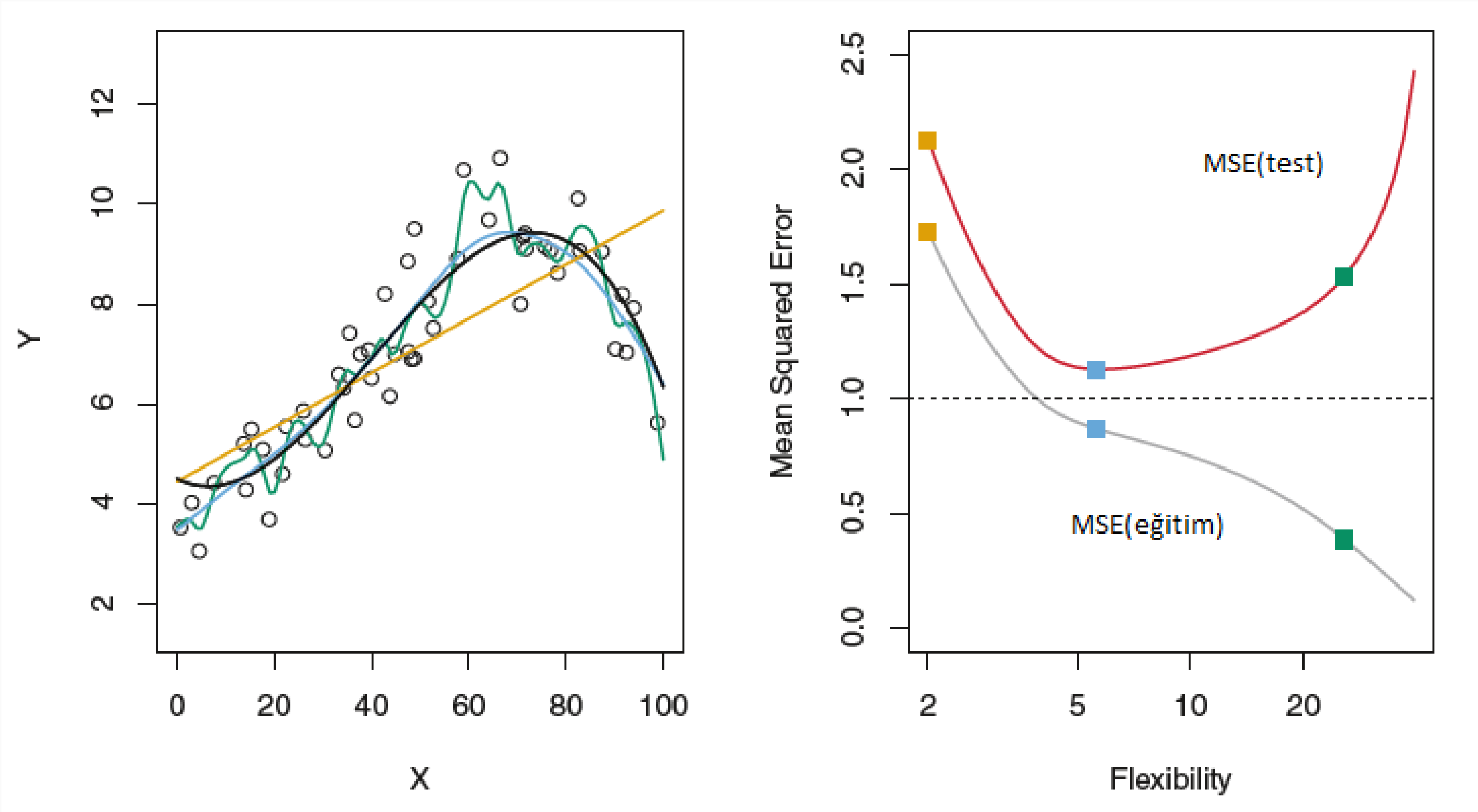
- Eğitim MSE'nin en küçük olması test MSE'nin de en küçük olacağı anlamına gelmez.

Test MSE

- Modelin esnekliği arttıkça MSE(eğitim) azalır.
- Eğitim Verileri: $\{Y_i, \mathbf{X}_i\}_{i=1}^{n}$
- Test Verileri: $\{Y_{0i}, \mathbf{X}_{0i}\}_{i=1}^{m}$
- Test MSE:

$$MSE_{test} = \frac{1}{m}\sum_{i=1}^{m}(y_{0i} - \hat{f}\,(x_{0i}))^2$$

- Modelin eğitim verilerinden hareketle tahmininden sonra test verileri ile tahmin yapılarak kolayca hesaplanabilir.
- Test verileri nereden geliyor?



Siyah eğri: gerçek ilişki, Tahmin edilen modeller: doğrusal model (kavuniçi), smoothing spline (mavi), daha esnek smoothing spline (yeşil) (Kaynak: James et al., Figure 2.9, s.31)

Sapma-Varyans Ödünümü

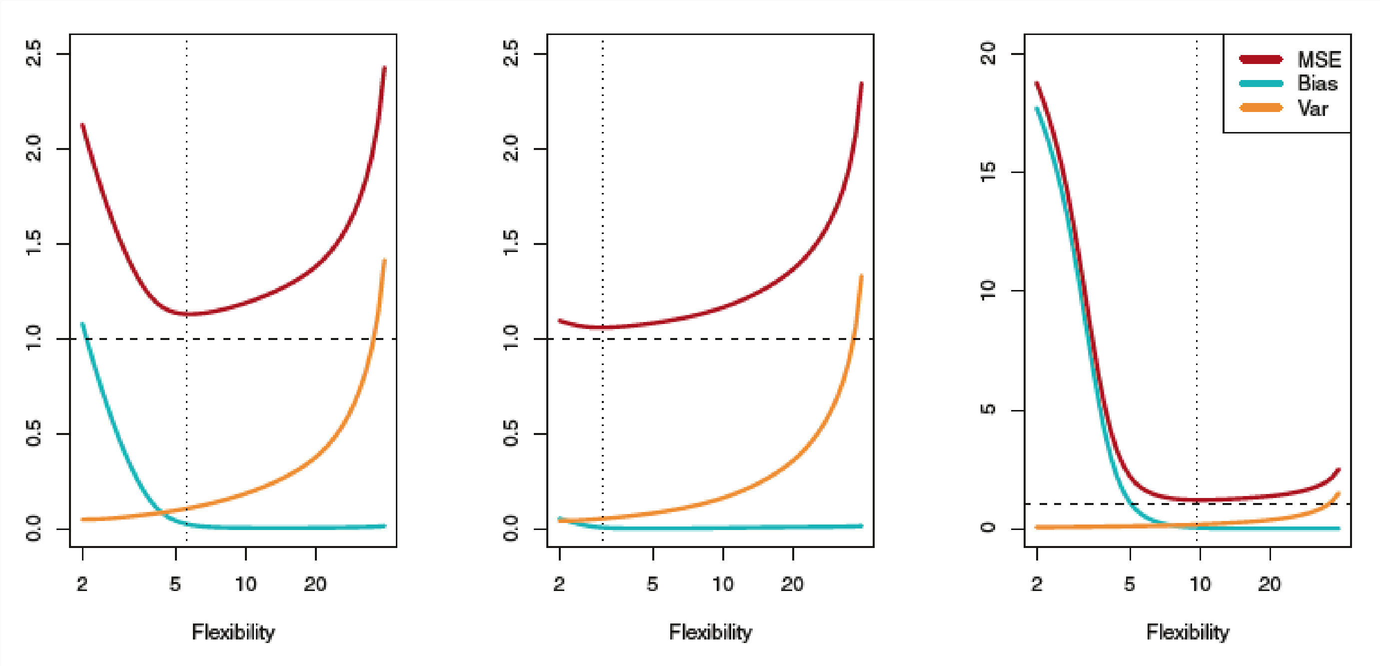
- Test verilerindeki beklenen Ortalama Hate Karesi aşağıdaki gibi yazılabilir:

$$E(MSE_{test}) = E\Big(y_0 - \hat{f}\ (x_0)\Big)^2 = \mathrm{Var}\Big(\hat{f}\ (x_0)\Big) + \Big[\mathrm{Bias}\Big(\hat{f}\ (x_0)\Big)\Big]^2 + \mathrm{Var}(\epsilon)$$

- Bias = Sapma (yanlılık)
- Beklenen test hatasının azaltılabilmesi için eşanlı olarak hem düşük varyanslı hem de düşük sapmalı öğrenme yönteminin seçilmesi gerekir.
- Modelin esnekliği (karmaşıklığı) arttıkça varyans artar sapma azalır.

Beklenen MSE(test)

Test MSE (kırmızı) = Modelin varyansı (kavuniçi) + Sapma Kare (mavi) + İndirgenemez Hata Varyansı (yatay kesikli çizgi)



Dikey kesikli çizgi: en küçük test MSE değerini veren karmaşıklık düzeyi (serbestlik derecesi - degrees of freedom) (Kaynak: James et al., Figure 2.12, s.36)

Sınıflandırma Problemleri

- Çıktı değişkeni: kategorik (ikili ya da çoklu olabilir)
- Yaptığımız tanımlamalar sınıflandırma problemleri için de geçerli. Ancak bazı ufak değişiklikler yapmak gerekebilir.
- Örneğin, modelin kestirim başarısını değerlendirmede MSE yerine hata oranını kullanacağız.
- Eğitim verilerindeki hata oranı **yanlış** sınıflandırılan gözlemlerin toplamdaki payıdır:

$$\frac{1}{n}\sum_{i=1}^{n} I\left(y_i \neq \hat{y}_i\right)$$

  Burada $I\left(\cdot\right)$ $y_i \neq \hat{y}_i$ ise 1, aksi durumda 0 değerini alan bir ikili değişkendir (indicator function).

- Aynı formülden hareketle (y_0, x_0) gibi bir test verisinden hareketle test hata oranı hesaplanabilir.

Bayes Sınıflandırıcısı

- Test hata oranını nasıl en düşük yapabiliriz?

- Bayes Sınıflandırıcısı (classifier): gözlemleri olasılığı en yüksek olan gruba sınıflandır.

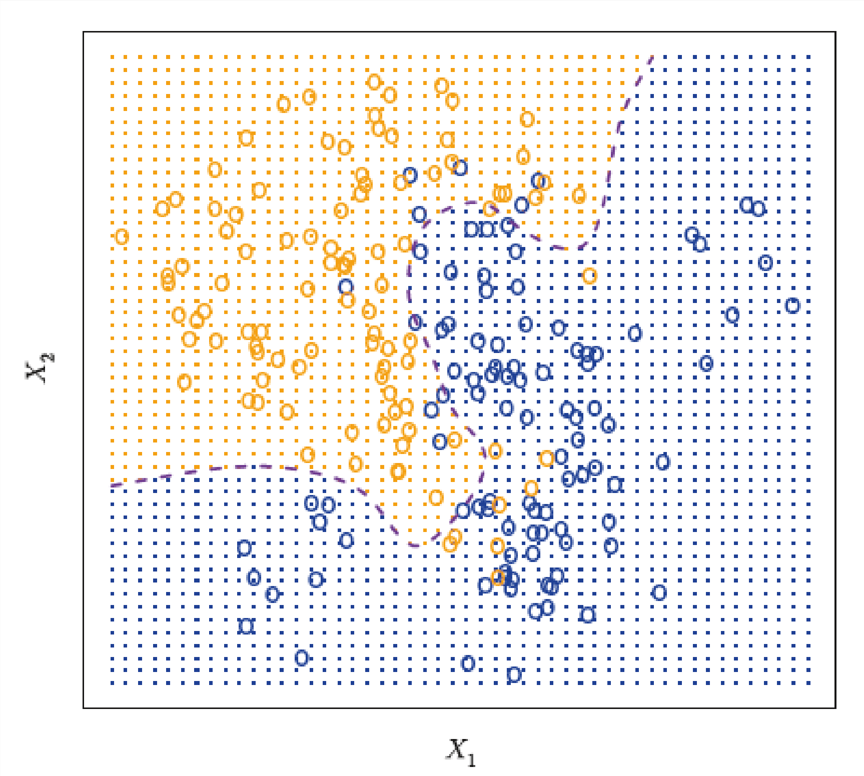
- Bu herhangi bir x_0 test verisi için koşullu olasılığın

$$\Pr(Y = j \mid X = x_0)$$

  en yüksek olduğu sınıfa atamanın yapılacağı anlamına gelir.

- Örneğin, ikili bir sınıflandırma probleminde (grup 1, grup 2),
  $\Pr(Y = 1 \mid X = x_0) > 0.5$ ise grup 1, değilse grup 2'ye sınıflandırma yapılır.

Bayes sınırı



- mor kesikli çizgi: Bayes sınıflandırma sınırı
- Olasılık 0.5'den büyükse kavuniçi gruba, değilse mavi gruba atama yapılır.

Bayes Hata Oranı

- Bayes sınıflandırıcısı olanaklı en düşük hata oranını verir.

- Bayes Hata Oranı:

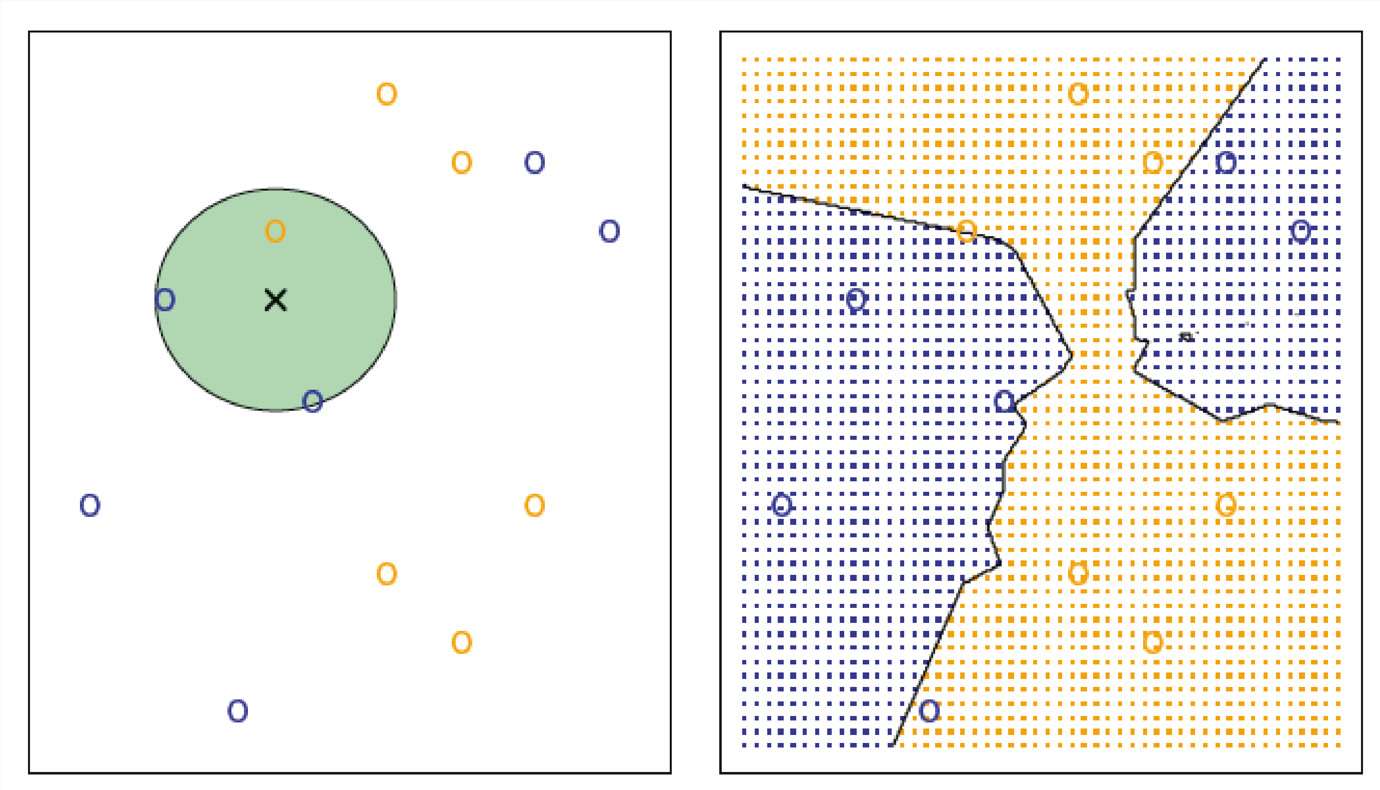
$$1 - E\left(\max_j \Pr(Y = j \mid X)\right)$$

- Bayes hata oranı daha önce gördüğümüz indirgenemez hata gibi düşünülebilir. Koşullu dağılım bilinmediği için Bayes hata oranı da bilinemez.

KNN Sınıflandırıcısı

- Pratikte test verilerinde Bayes hata oranından daha düşük hata oranı elde edilemez.
- Bayes sınıflandırıcısını kullanabilmemiz için her grubun koşullu olasılığını verilerden hareketle tahmin etmemiz gerekir. Böyle bir model bulduktan sonra en yüksek olasılıklı sınıf seçilebilir.
- Koşullu olasılık dağılımının kestiriminde kullanılabilecek bir yöntem K-en yakın komşu (K-Nearest Neighbor, KNN) yöntemidir.
- KNN yönteminde x_0 test noktasına eğitim verisinde en yakın K nokta belirlenir. Daha sonra bu K nokta içinde en fazla sıklığa sahip olan gruba atama yapılır.

KNN Örnek

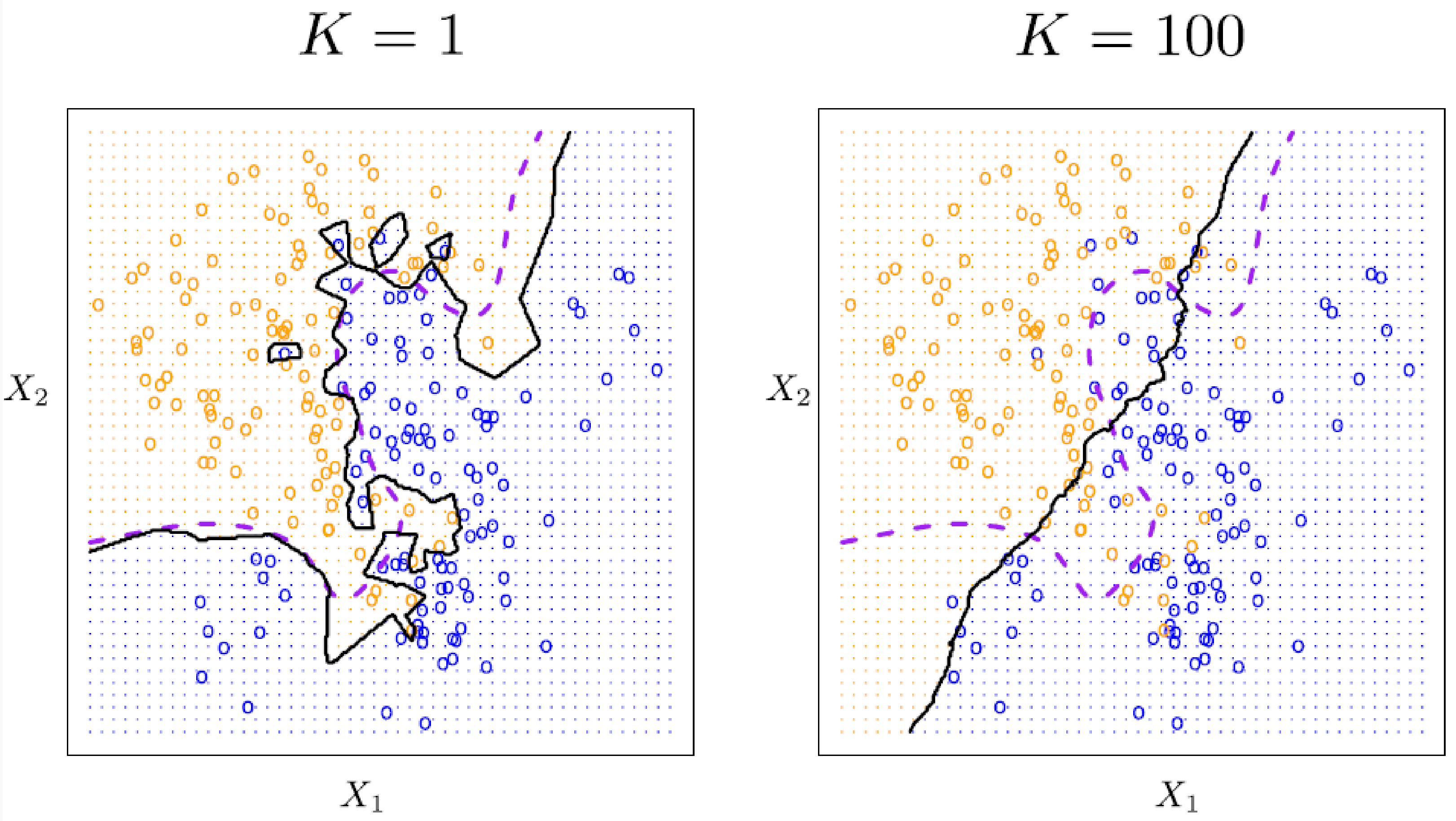


Notlar: $K = 3$ için (bkz. soldaki grafik) x noktasına en yakın değerler içinde en fazla sıklığa sahip olan mavi gruptur. KNN karar sınırı sağ tarafta gösterilmiştir. (Kaynak: James et al., Figure 2.14, s.40)

Aşırı uyum tehlikesi

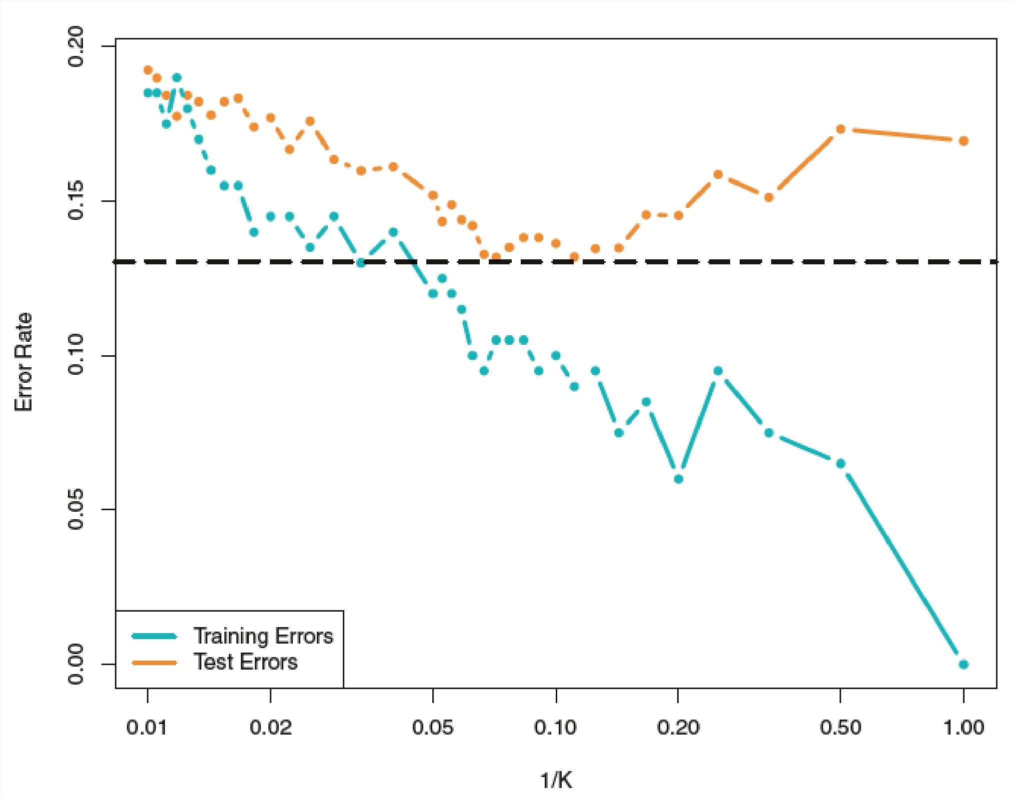
- KNN sınıflandırıcısında K parametresi modelin performansını önemli ölçüde etkiler.
- K arttıkça komşuluk içine giren nokta sayısı artar ve model daha **az esnek** hale gelir.
- K azaldıkça modelin esnekliği artar.
- Örnek olarak $K = 1$ ve $K = 100$ durumlarını ele alalım.

KNN'de Aşırı uyum



(Kaynak: James et al., Figure 2.16, s.41)

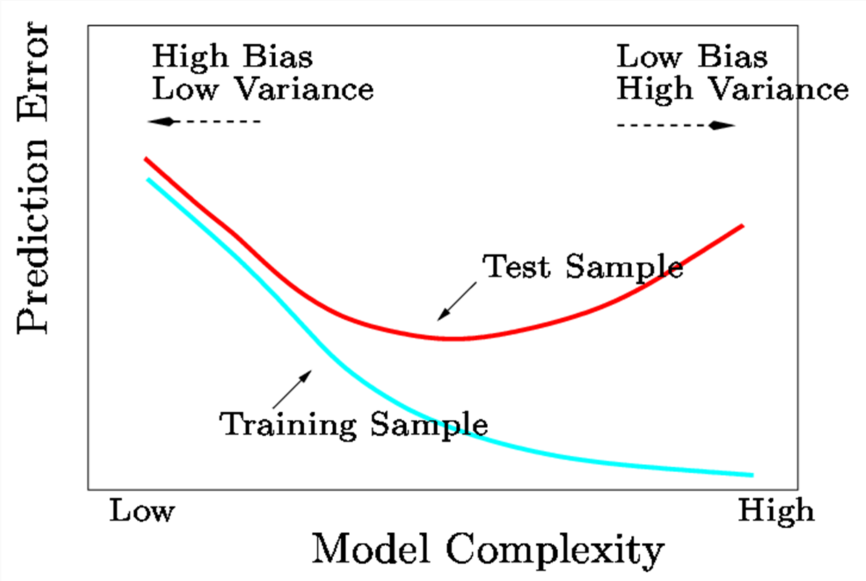
KNN Eğitim ve Test Hata Oranları



Not: Kesikli siyah çizgi Bayes hata oranıdır (veriler simülasyonla üretildiği için biliniyor)

(Kaynak: James et al., Figure 2.17, s.42)

Makine Öğrenmesinde Hata Davranışı



- Model karmaşıklığı arttıkça eğitim verisindeki kestirim hatası azalmaya devam eder.
- Model karmaşıklığı arttıkça test verisindeki kestirim hatası bir noktaya kadar azalır, daha sonra artmaya başlar. Sapma düşük olsa da varyans çok yüksektir (aşırı uyum)
- Test kestirim hatasının en düşük olduğu model için sapma ve varyans en optimal düzeydedir.