

Doğrusal Olmayan Modeller

(İktisatçılar İçin) Makine Öğrenmesi (TEK-ES-2021)

Hüseyin Taştan
Yıldız Teknik Üniversitesi

Plan

- Doğrusal ve Doğrusal Olmayan Modeller
- Polinom regresyonu
- Adım fonksiyonu
- Baz fonksiyonları
- Spline regresyonu
- Doğal Spline'lar
- Düzleştirme Spline'ları
- Lokal regresyon
- Genelleştirilmiş Toplamsal Modeller (GAMs)
- Yapay Sinir Ağları

Doğrusal ve Doğrusal Olmayan Modeller

Model: $y = f(x) + u$, burada $f(x)$ ilişkinin formunu belirler

Doğrusal modeller:

- $f(x) = \beta_0 + \beta_1 x$
- Hem parametrelerde hem de değişkenlerde doğrusal
- Pratikte iyi bir başlangıç noktası olabilir, ancak kestirim performansı çok başarılı olmayabilir
- Ancak çeşitli genelleştirmelerle iyileştirilebilir.

Doğrusal olmayan modeller

- $f(x)$ x 'in herhangi bir fonksiyonu olabilir. Daha esnek bir çerçeve sunar.
- Polinom regresyonu
- Adım fonksiyonu
- Regresyon spline'ları
- Düzleştirme spline'ları
- Yerel (local) regresyon
- Genelleştirilmiş Toplamsal Modeller (Generalized Additive Models - GAMs)
- ... ve diğerleri (yapay sinir ağları, destek vektör makineleri, ağaçlar, vb.)

Polinom Regresyonu

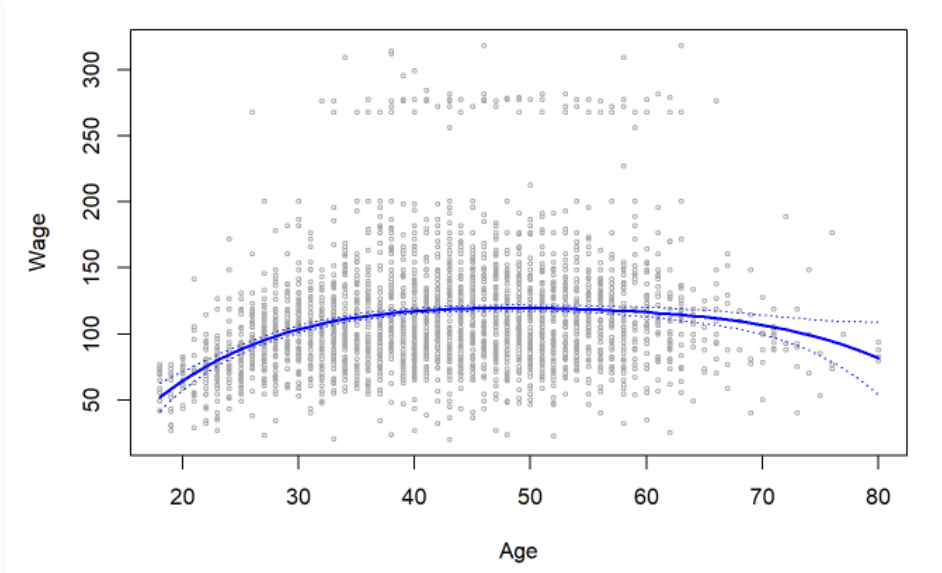
- Basitlik amacıyla sadece bir kestirim değişkeni, x , olsun. Derecesi d olan bir polinom aşağıdaki gibi yazılabilir:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \dots + \beta_d x^d + \epsilon$$

- Modele kestirim değişkeninin d kuvvetini ekledik.
- Model hala parametrelerde doğrusal olduğu için sıradan en küçük kareler (OLS) yöntemi kullanabiliriz.
- Polinom derecesi, d , modelin karmaşıklığını belirleyen bir ayarlanma parametresi olarak düşünülebilir.
- Pratikte d geçerleme yaklaşımı ile seçilebilir.

Örnek: Ücret-yaş ilişkisi için 4. derece polinom

$$\widehat{wage} = \hat{\beta}_0 + \hat{\beta}_1 age + \hat{\beta}_2 age^2 + \hat{\beta}_3 age^3 + \hat{\beta}_4 age^4$$



- Mavi kesiksiz çizgi tahmin edilen eğridir (%95 güven aralığı ile birlikte verilmiştir).
- Yüksek ücrete sahip bir grup mevcuttur (yıllık kazancı 250.000 USD üzerinde olanlar)
- Bu grubu nasıl modelleyebiliriz?

Polinom terimli lojistik regresyon

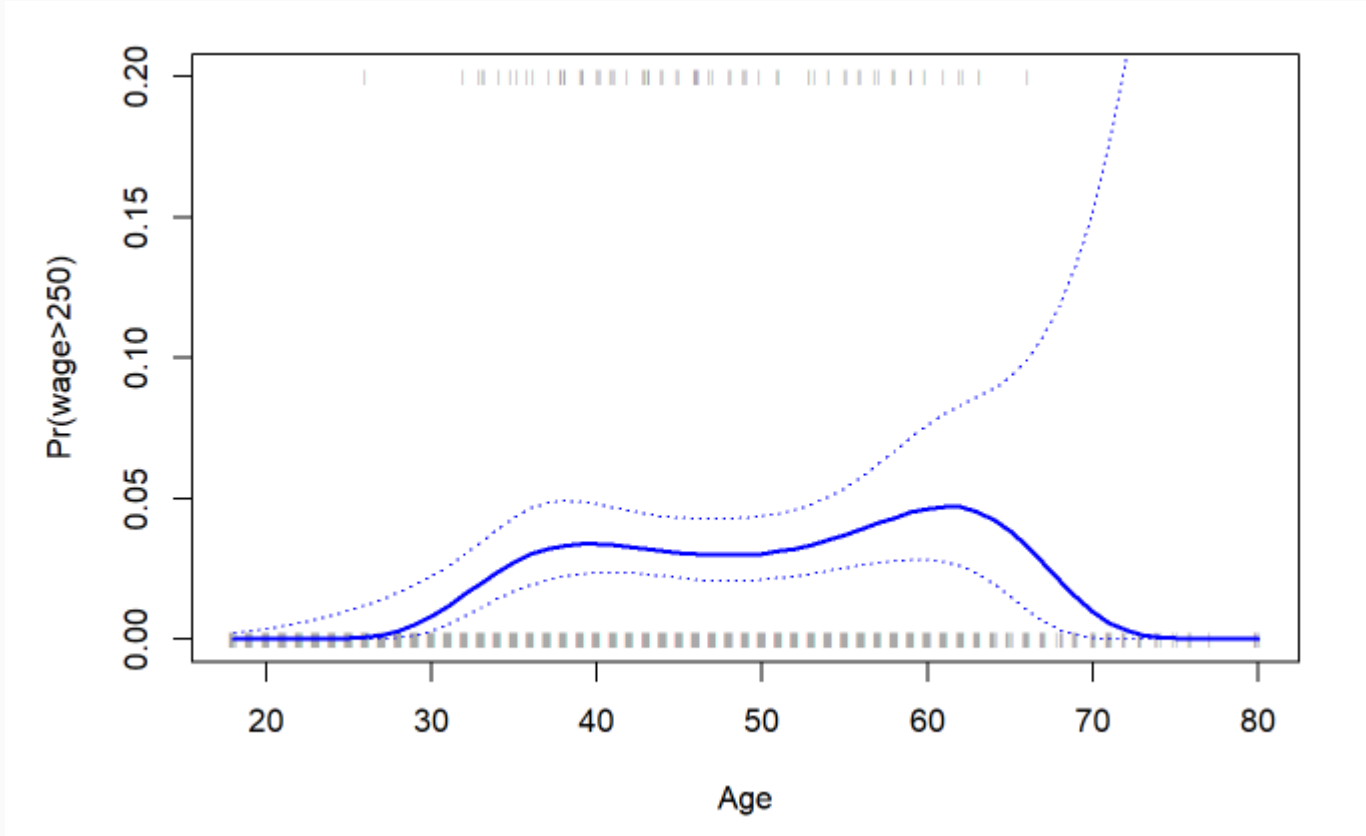
- Polinom regresyonuna benzer şekilde bir lojistik regresyon sınıflandırma modeli kurabiliriz.
- Örneğin yüksek ücretli grup için

$$\Pr(y_i > 250 \mid x_i) = \frac{\exp(\beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots + \beta_d x_i^d)}{1 + \exp(\beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \dots + \beta_d x_i^d)}$$

Burada veriler ücret değişkenine göre kazancı 250 bin doların üzerinde olanlar ve olmayanlar şeklinde iki gruba ayrılmıştır.

- İzleyen grafik tahmin edilen olasılıkları ve %95 güven aralığını göstermektedir.

Örnek: polinom terimli lojistik regresyon



Veri setinde sadece 79 yüksek ücretli gözlem yer almaktadır. Veri setinin küçük olması nedeniyle varyans yüksektir ve güven aralıkları geniştir.

Adım fonksiyonu

Gösterge (indicator) fonksiyonu: X değişkeni için c_1, c_2, \dots, c_k kesme noktalarını kullanarak $K + 1$ kategorik değişken oluşturulabilir:

$$C_0(X) = I(X < c_1)$$

$$C_1(X) = I(c_1 \leq X < c_2)$$

$$C_2(X) = I(c_2 \leq X < c_3)$$

$$\vdots$$

$$C_{K-1}(X) = I(c_{K-1} \leq X < c_K)$$

$$C_K(X) = I(c_K \leq X)$$

- Gösterge fonksiyonu, $I(\cdot)$, parantez içindeki olay doğruysa 1 değilse 0 değerini alır (doğru, yanlış). c kesme değerlerine göre kukla değişkenler yaratır.
- Örnek: $c_1 = 35, c_2 = 50, c_3 = 65$
- $I(\text{age} < 35) = 1$: 35 yaşından küçükler için 1 değerini alır, diğerleri için 0 olur.
Dikkat: her gözlem C_0, C_1, \dots, C_K gruplarından birine girmelidir.

Adım fonksiyonu

- X 'in herhangi bir değeri için

$$C_0(X) + C_1(X) + \dots + C_K(X) = 1$$

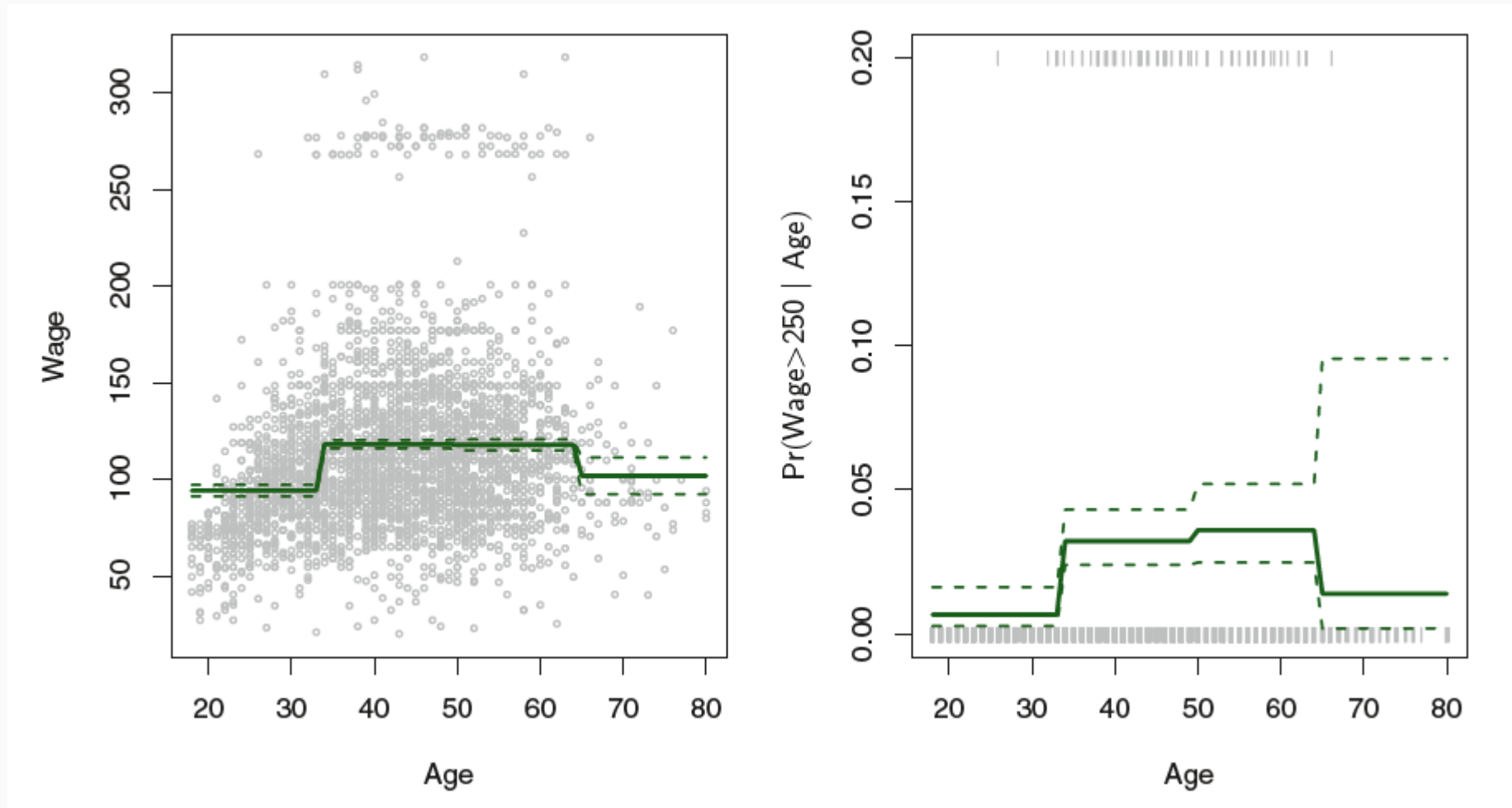
yazılabilir (X değerleri $K + 1$ gruptan birinde yer almalıdır).

- Modelin tahmini OLS ile yapılır. $C_1(X), C_2(X), \dots, C_K(X)$ kukla değişkenleri modele eklenir:

$$y_i = \beta_0 + \beta_1 C_1(x_i) + \beta_2 C_2(x_i) + \dots + \beta_K C_K(x_i) + \epsilon_i$$

- β_0 : $X < c_1$ olduğunda Y 'nin ortalaması
- β_j : $X < c_1$ grubuna kıyasla X in $c_j \leq X < c_{j+1}$ grubunun Y 'nin ortalamasına katkısı.
- İzleyen grafikte 35, 50 ve 65 kesme değerleri için adım fonksiyonu regresyonu gösterilmiştir.

Örnek: adım fonksiyonu



(source: ISLR Fig. 7.2, p.269)

Baz fonksiyonları

- Polinomlar ve parçalı-sabit regresyon modelleri (adım fonksiyonu) baz fonksiyonlarının özel bir durumu olarak düşünülebilir.
- Bir baz fonksiyonu ailesini, $b_1(X), b_2(X), \dots, b_K(X)$ ile göstereceğiz. Bu fonksiyonlar her X değerine uygulanır.
- Modelimizde orijinal X değerlerini değil baz fonksiyonlarını kullanacağız:

$$y_i = \beta_0 + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + \beta_3 b_3(x_i) + \dots + \beta_K b_K(x_i) + \epsilon_i.$$

- Polinom regresyonu için baz fonksiyonu: $b_j(x_i) = x_i^j$
- Parçalı sabit regresyon: $b_j(x_i) = I(c_j \leq x_i < c_{j+1})$
- Çok sayıda alternatif mevcuttur: spline'lar, wavelet'ler, Fourier dizileri, ...

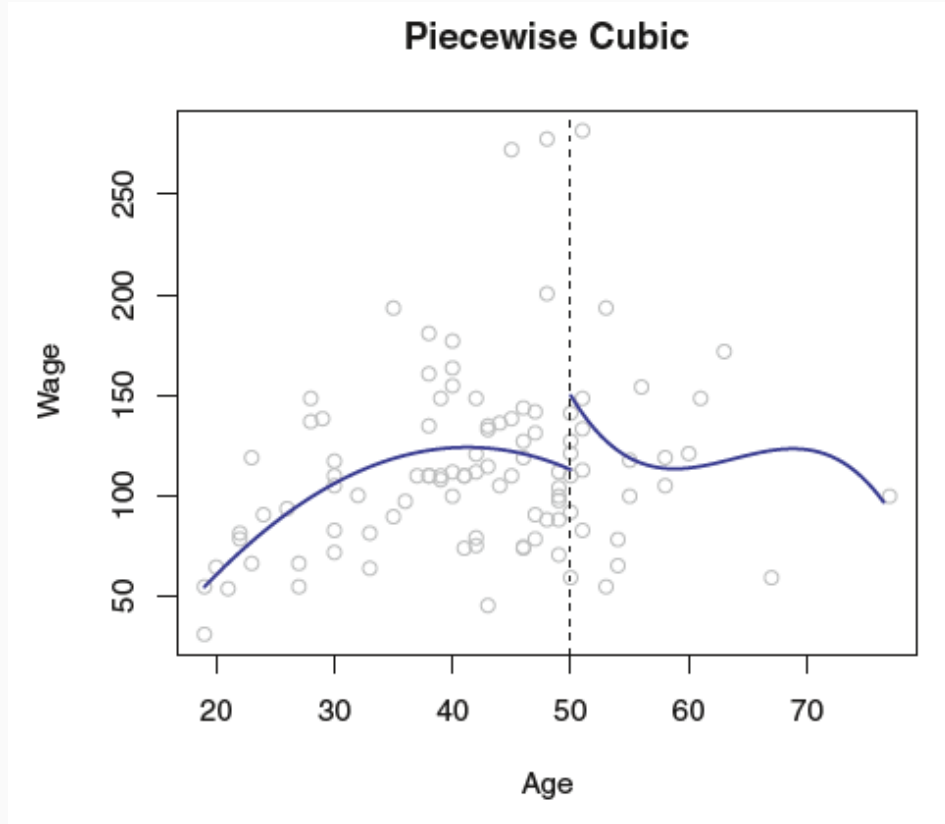
Parçalı polinomlar

- Pratikte X 'in tüm değerleri üzerine bir yüksek dereceli polinom regresyonu genellikle tercih edilmez. Alternatif olarak X 'in bazı değer aralıkları için düşük dereceli polinom tahmin edilmesidir. .
- X 'in değer aralığı adım fonksiyonunda olduğu gibi alt aralıklara bölünür. Kesme noktalarına düğüm (knot) adı verilir.
- Örneğin, c düğüm noktasını kullanarak bir parçalı kübik polinom modeli:

$$y_i = \begin{cases} \beta_{01} + \beta_{11}x_i + \beta_{21}x_i^2 + \beta_{31}x_i^3 + \epsilon_i & \text{if } x_i < c \\ \beta_{02} + \beta_{12}x_i + \beta_{22}x_i^2 + \beta_{32}x_i^3 + \epsilon_i & \text{if } x_i \geq c \end{cases}$$

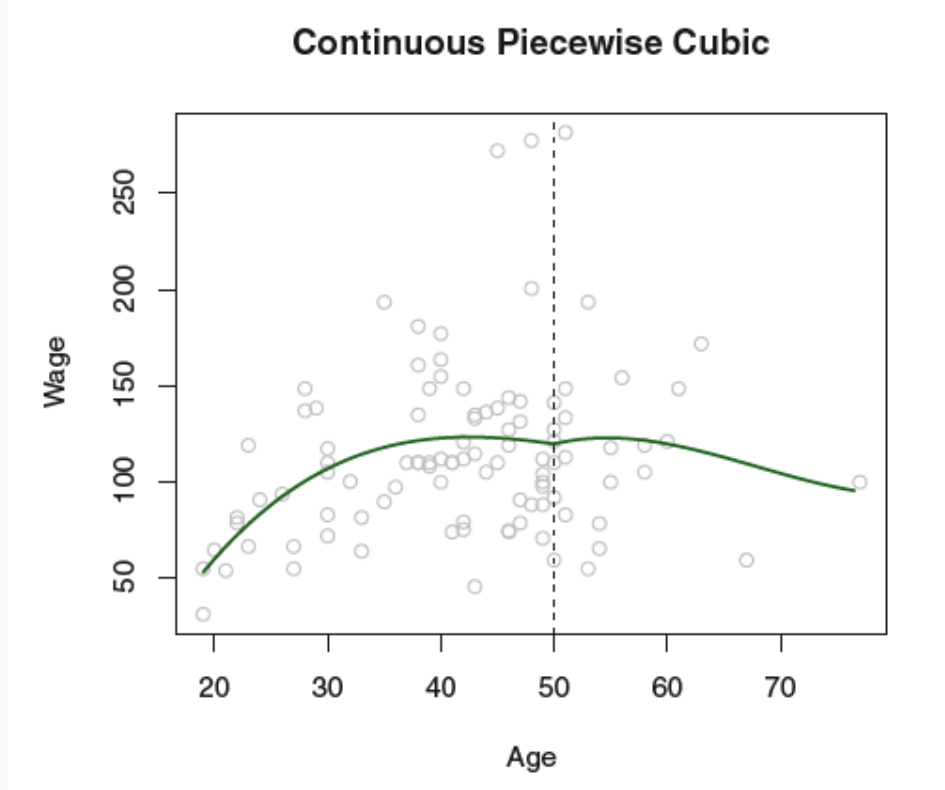
- OLS ile tahmin edilebilir.
- Polinom derecesi değiştirilebilir veya istenen sayıda düğüm noktası oluşturulabilir.

Örnek: Kısıtlanmamış parçalı kübik polinom



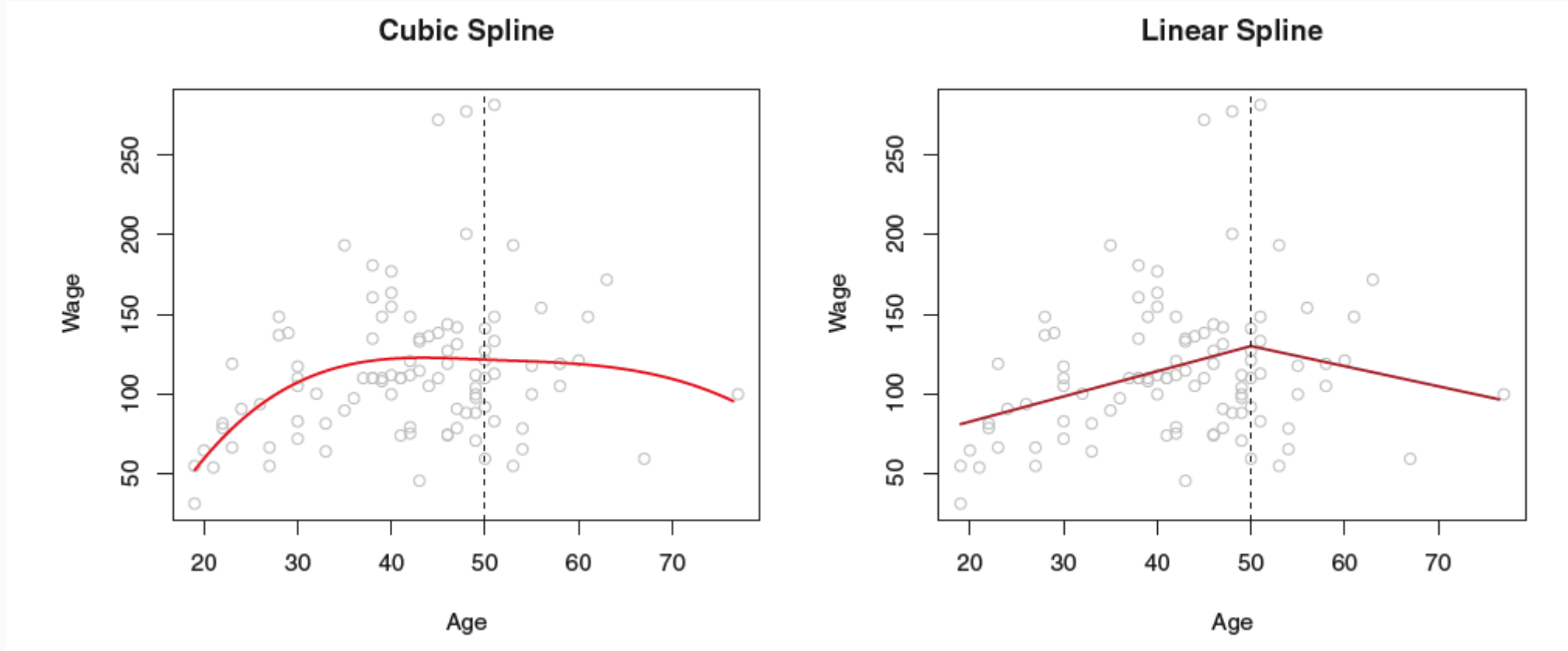
- Ücret-yaş parçalı kübik polinom modeli (düğüm noktası age=50).
- Düğüm noktasında süreksizlik var.
- Polinom modelini düğüm noktasında sürekli olacak şekilde kısıtlayabiliriz.
- Süreklilik kısıtının yanı sıra $d - 1$ türevin sürekli olmasını gerektiren düzgünlük kısıtını da koyabiliriz (smoothness)

Örnek: Kısıtlanmış parçalı kübik polinom



- Düğüm noktasında (age=50) süreklilik kısıtı altında parçalı kübik polinom.
- Düğüm noktasında hala görünür bir değişim mevcut.
- Daha düzgün bir alternatif: spline'lar

Örnek: spline



Kübik spline süreklilik şartını sağlar, ayrıca birinci ve ikinci türevleri de sürekli dir. (Source: ISLR Fig. 7.3, p.272)

Doğrusal Spline

- K düğüm noktası, ξ_k , $k = 1, 2, \dots, K$, için doğrusal spline aşağıdaki gibi yazılabilir:

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 (x_i - \xi_1)_+ + \beta_3 (x_i - \xi_2)_+ + \dots + \beta_{1+K} (x_i - \xi_K)_+ + \epsilon_i$$

Burada $(\cdot)_+$ pozitif kısmı gösterir:

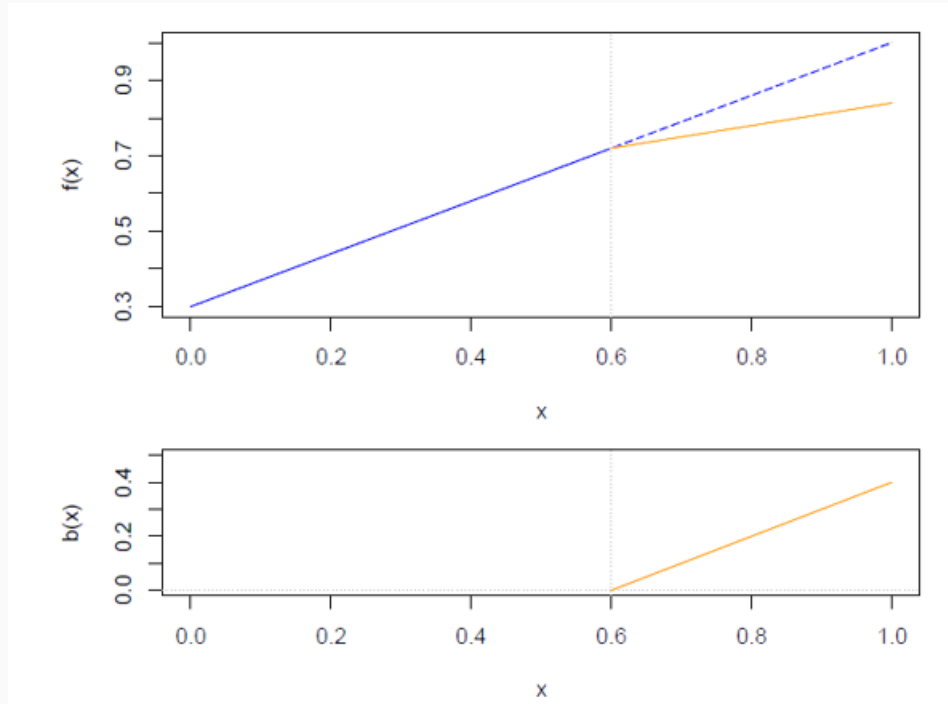
$$(x - \xi)_+ = \begin{cases} (x - \xi) & \text{if } x > \xi \\ 0 & \text{otherwise} \end{cases}$$

- Örneğin, 25, 40 , ve 60 yaşlarında düğüm noktaları ile bir doğrusal spline:

$$wage = \beta_0 + \beta_1 age + \beta_2 (age - 25)_+ + \beta_3 (age - 40)_+ + \beta_4 (age - 60)_+ + \epsilon$$

Burada $K = 3$ düğüm noktası ve 5 parametre vardır.

Doğrusal spline: örnek



- Global doğrusal fonksiyon
 $f(x) = \beta_0 + \beta_1 x$ mavi ile gösterilmiştir.
- Düğüm noktası = 0.6:
 $f(x) = \beta_0 + \beta_1 x + \beta_2(x - 0.6)_+$
- Baz fonksiyonu: $b(x) = (x - 0.6)_+$
(kavuniçi ile gösterilmiştir).

- Dikkat edilirse baz fonksiyonu 0'da başlar ve düğüm noktasında sürekliliği sağlar.
- Global fonksiyon düğüm noktasında eğim değiştirir.

Kübik spline

- d dereceli bir spline fonksiyonu aslında bir parçalı polinom fonksiyonudur ancak düğüm noktalarında türevleri süreklidir.
- Bir kübik spline aşağıdaki gibi yazılabilir:

$$y_i = \beta_0 + \beta_1 b_1(x_i) + \beta_2 b_2(x_i) + \cdots + \beta_{K+3} b_{K+3}(x_i) + \epsilon_i$$

- Kübik spline baz fonksiyonu: modele x, x^2, x^3 ile başlanır ve her bir düğüm noktası için kesilmiş baz kuvvet fonksiyonları eklenir:

$$h(x, \xi) = (x - \xi)_+^3 = \begin{cases} (x - \xi)^3 & \text{if } x > \xi \\ 0 & \text{otherwise} \end{cases}$$

Burada $(\cdot)_+$ pozitif kısmı gösterir.

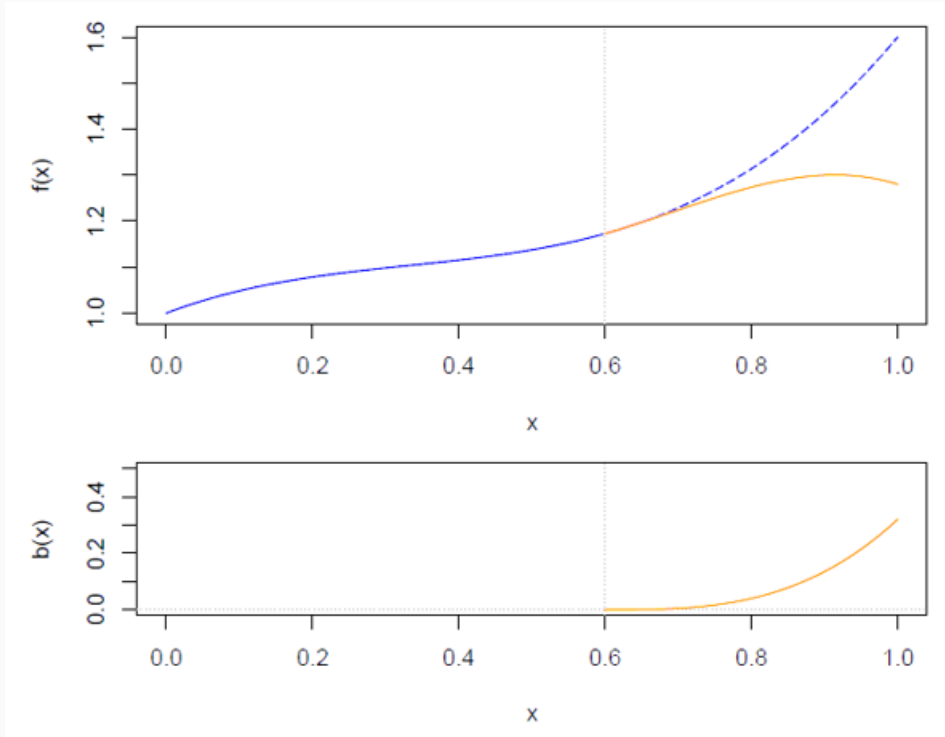
Kübik spline

- K düğüm noktaları belirlenmiş olsun. Bir kübik spline modelini OLS ile tahmin etmek için değişkenin kuvvetlerinin X, X^2, X^3 yanı sıra $h(x, \xi_1), h(x, \xi_2), \dots, h(x, \xi_K)$ değerlerini de kestirim değişkeni olarak kullanırız.
- Bir kübik spline $4 + K$ serbestlik derecesine sahiptir (parametre sayısı)
- Örneğin ücret modelinde 25, 40, 60 düğüm noktaları ile:

$$\begin{aligned} wage = & \beta_0 + \beta_1 age + \beta_2 age^2 + \beta_3 age^3 + \\ & \beta_4 (age - 25)_+^3 + \beta_5 (age - 40)_+^3 + \beta_6 (age - 60)_+^3 + \epsilon \end{aligned}$$

Burada $K = 3$ ve toplamda 7 parametre vardır.

Kübik spline: örnek

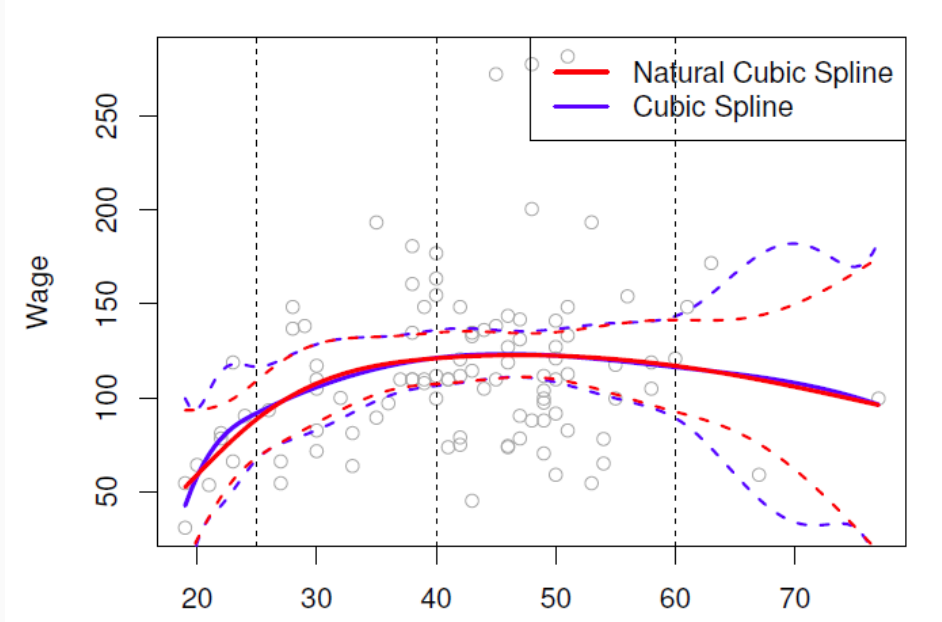


- Kübik spline baz fonksiyonu (kavuniçi) düğün noktasında (0.6) süreklidir.
- Global fonksiyon düğüm noktasında düzgün bir şekilde eğim değiştirir.

- 0.6 düğüm noktası ile bir kübik splineB

$$f(x) = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \beta_4 (x - 0.6)_+^3$$

Doğal spline



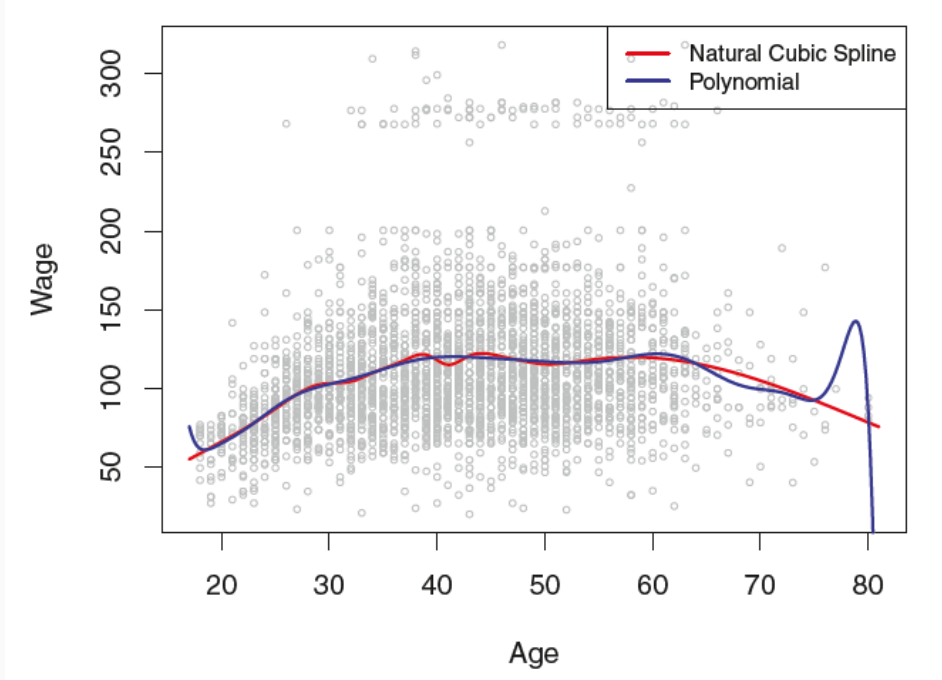
- Spline'lar X değer aralığının uçlarında yüksek değişkenliğe sahiptir.
- Bu değişkenlik grafikte güven bandındaki genişlemeden anlaşılabilir.
- Bir doğal kübik spline sınırdaki düğüm noktalarının ötesinde doğrusal ekstrapolasyon yapar.

- Sınır kısıtları: fonksiyon $\text{age} < 25$ ve $\text{age} > 60$ için doğrusal olmaya zorlanır.
- Bu $4 = 2 \times 2$ ek kısıt getirir.
- Grafikten de görüleceği gibi doğal spline güven bandı daha dardır.

Düğümlerin yeri ve sayısının belirlenmesi

- Pratikte düğüm sayısına ve bu düğümlerin yerine karar vermemiz gerekir.
- Temel ilke: fonksiyonun hızlıca değiştiği bölgelere daha fazla düğüm noktası yerleştir, fonksiyonun daha stabil olduğu bölgelere ise daha az.
- Diğer bir opsiyon düğüm noktasına karar verdikten sonra kullandığımız yazılımın otomatik olarak bunların yerlerini belirlemesidir (örneğin 25, 50, 75 yüzdelerlik dilimler).
- Düğüm sayısının belirlenmesinde çapraz geçerleme yaklaşımı kullanılabilir.

Polinom regresyonu ve spline karşılaştırması



- 15 dereceli bir polinom (mavi) ve doğal kübik spline (kırmızı)
- Polinom regresyonu özellikle sınırlarda yüksek değişkenliğe sahiptir.
- Polinomlara kıyasla spline'lar daha istikrarlıdır.
- Spline'lar dereceyi sabit tutarken düğüm sayısını arttırarak daha esnek bir modelleme sunmaktadır.

(Source: ISLR Fig. 7.7, p.277)

Düzleştirme Spline'ları

- Regresyon spline'ları: bir baz fonksiyon kümesi tanımla ve modeli OLS ile tahmin et.
- Alternatif yaklaşım: verilere en iyi uyumu veren $g(x)$ gibi bir fonksiyon bul. Bunun için kalıntı kareleri toplamı, $RSS = \sum_{i=1}^n (y_i - g(x_i))^2$ en küçük yapılır.
- Ancak böyle bir problemde uygun kısıtlar konmazsa aşırı-uyum problemi ortaya çıkabilir. Bundan kaçınmak için fonksiyonun düzgün (smooth) olmasını isteriz.
- Bunu sağlamanın bir yolu amaç fonksiyonuna bir ceza terimi eklemektir:

$$\sum_{i=1}^n (y_i - g(x_i))^2 + \lambda \int g''(t)^2 dt$$

Burada λ negatif olmayan bir ayarlanma parametresidir. $g''(t)$ ise $g(x)$ 'in ikinci türevidir.

Düzleştirme spline'ları (smoothing splines)

- Problemin kayıp+ceza (Loss+Penalty) yapısına sahip olduğuna dikkat ediniz:

$$\underbrace{\sum_{i=1}^n (y_i - g(x_i))^2}_{Loss} + \underbrace{\lambda \int g''(t)^2 dt}_{Penalty}$$

- Bir fonksiyonun birinci türevi, $g'(t)$, fonksiyonun t noktasındaki eğimini ölçer.
- İkinci türev, $g''(t)$, ise t noktasında eğimin ne kadar değiştiğini gösterir.
- Bu nedenle ikinci türev fonksiyonun ne kadar düzgün olduğunun veya olmadığının göstergesidir (roughness). Büyük değerler aldığında t noktasında fonksiyon hızlıca değişir. Aksi durumda sıfıra yakın değerler alır.
- İkinci türevin karesinin integrali, $\int g''(t)^2 dt$, tüm değerler aralığı için $g'(t)$ fonksiyonundaki toplam değişimi gösterir.

Smoothing Splines

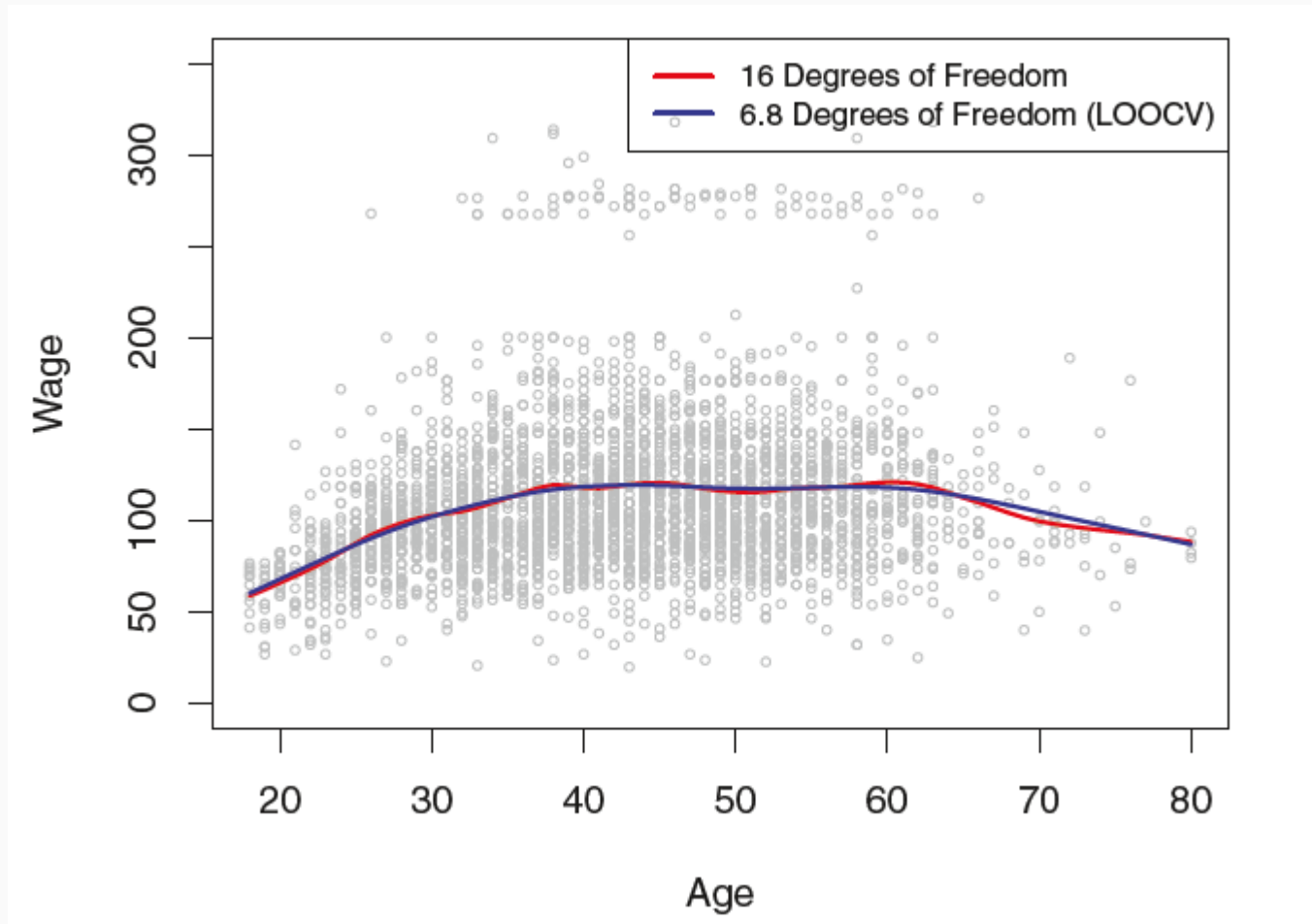
$$\underbrace{\sum_{i=1}^n (y_i - g(x_i))^2}_{Loss} + \underbrace{\lambda \int g''(t)^2 dt}_{Penalty}$$

- $g()$ ne kadar düz ise $\int g''(t)^2 dt$ o kadar küçük değerler alır ve $g'(t)$ sabit olur.
- Diğer taraftan, $g()$ çok değişken ise $g'(t)$ fonksiyonu da değişken olur ve $\int g''(t)^2 dt$ büyük değerler alır.
- Ceza terimi, $\lambda \int g''(t)^2 dt$, fonksiyonun düzgün olmasını sağlar. λ parametresi ne kadar büyükse fonksiyon o kadar düzgün olur.
- $\lambda = 0$ olduğunda bir ceza uygulanmaz; sonuçta mükemmel uyum ortaya çıkar.
- $\lambda \rightarrow \infty$ durumunda $g()$ düzgünleşir ve limitte doğrusal olur.

Düzleştirme parametresinin seçimi

- λ büyüdükçe fonksiyon doğrusala yaklaşır (en küçük kareler doğrusu)
- Ara değerler için fonksiyon $g(\cdot)$ eğitim verilerine düzgün bir şekilde uyumlanmaya çalışır.
- Ayarlanma parametresi λ düzleştirme spline'larının ne kadar değişken olduğunu kontrol eder ve efektif serbestlik derecesi ile ilişkilidir.
- λ 0'dan ∞ 'a doğru değişirken efektif serbestlik derecesi n 'den 2'ye doğru değişir.
- Düğüm noktası seçimine gerek yoktur. Sadece λ veya efektif serbestlik derecesinin seçimi yeterlidir.
- Bunun seçiminde çapraz geçерleme ya da LOOCV kullanılabilir.

Örnek



(Source: ISLR Fig. 7.8, p.280)

Yerel Regresyon

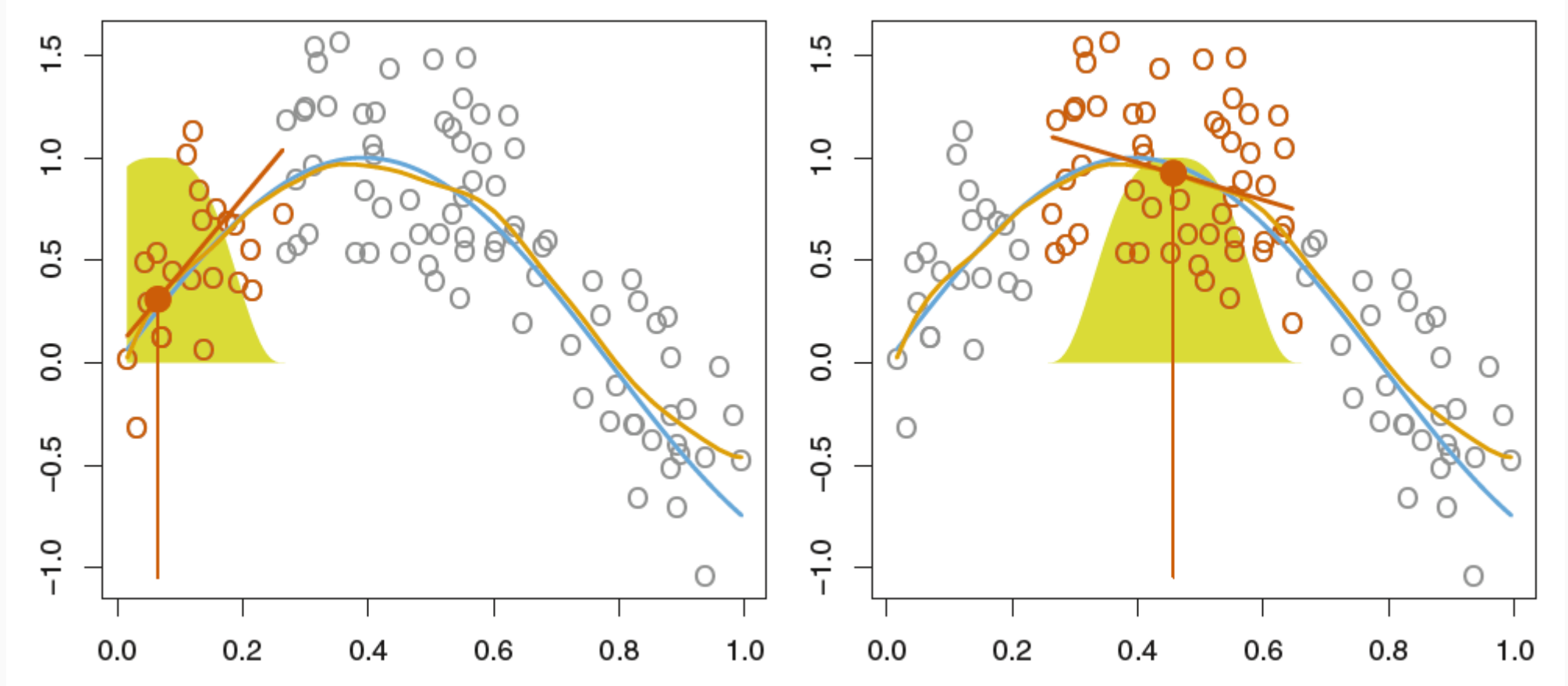
- Adından da anlaşılacağı gibi yerel regresyon hedef x_0 noktasının belirli bir komşuluğundaki gözlemleri kullanır.
- Bir ağırlıklandırma fonksiyonu yardımıyla hedef noktaya yakın gözlemlere daha fazla ağırlık verilir.
- Bu ağırlık fonksiyonun kernel fonksiyonu da denir:

$$K_{i0} = \frac{1}{h} k \left(\frac{x_i - x_0}{h} \right)$$

- Yerel regresyonun amaç fonksiyonu:

$$\sum_{i=1}^n K_{i0} (y_i - \beta_0 - \beta_1 (x_i - x_0))^2$$

Yerel Regresyon

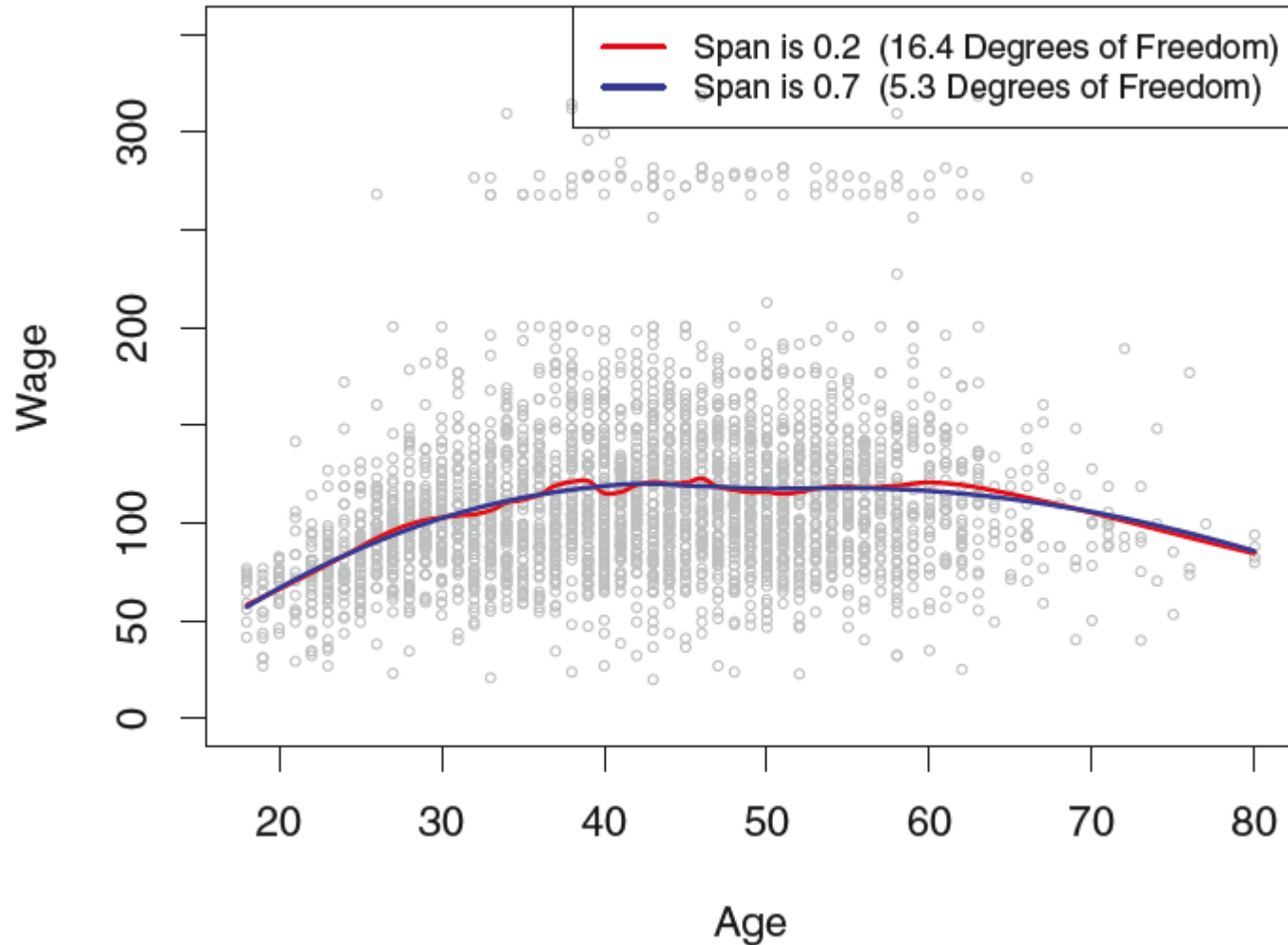


(Kavuniçi noktalar hedef değere yakın (yerel) gözlemlerdir. Mavi eğri gerçek fonksiyon (bu pratikte bilinmez), kavuniçi eğri ise yerel regresyondur. Kaynak: ISLR Fig. 7.9, p.281)

Yerel regresyon

- Pratikte hedef noktasının çevresinde pencere içinde kaç gözlem kullanacağımıza ya da oranına, $s = k/n$, karar vermemiz gerekir. Buna span s adı verilir.
- Burada span aslında λ gibi bir ayarlanma parametresidir. Doğrusal olmayan modelin ne kadar esnek olacağını belirler.
- s ne kadar küçükse fonksiyon o kadar inişli-çıkışlı olur.
- Ters durumda, büyük bir s değeri tüm gözlemlerle bir regresyon doğrusu tahmin etmekle sonuçlanır.
- s doğrudan belirlenebilir veya çapraz geçerleme ile seçilebilir.
- İzleyen grafikte iki farklı span değeri, $s = 0.2$ ve $s = 0.7$, kullanılmıştır.
- Span değeri yüksek olan fonksiyon beklendiği gibi diğerine göre daha düzgündür.

Lokal regresyon: örnek



(Source: ISLR Fig. 7.10, p.283)

Generalized Additive Models (GAMs)

- Genelleştirilmiş Toplamsal Modeller (GAMs) modelin eklemeli yapısını korur ancak her bir X değişkeninin farklı bir doğrusal olmayan yaklaşımla modellenmesine izin verir:

$$\begin{aligned} y_i &= \beta_0 + \sum_{j=1}^p f_j(x_{ij}) + \epsilon_i \\ &= \beta_0 + f_1(x_{i1}) + f_2(x_{i2}) + \cdots + f_p(x_{ip}) + \epsilon_i \end{aligned}$$

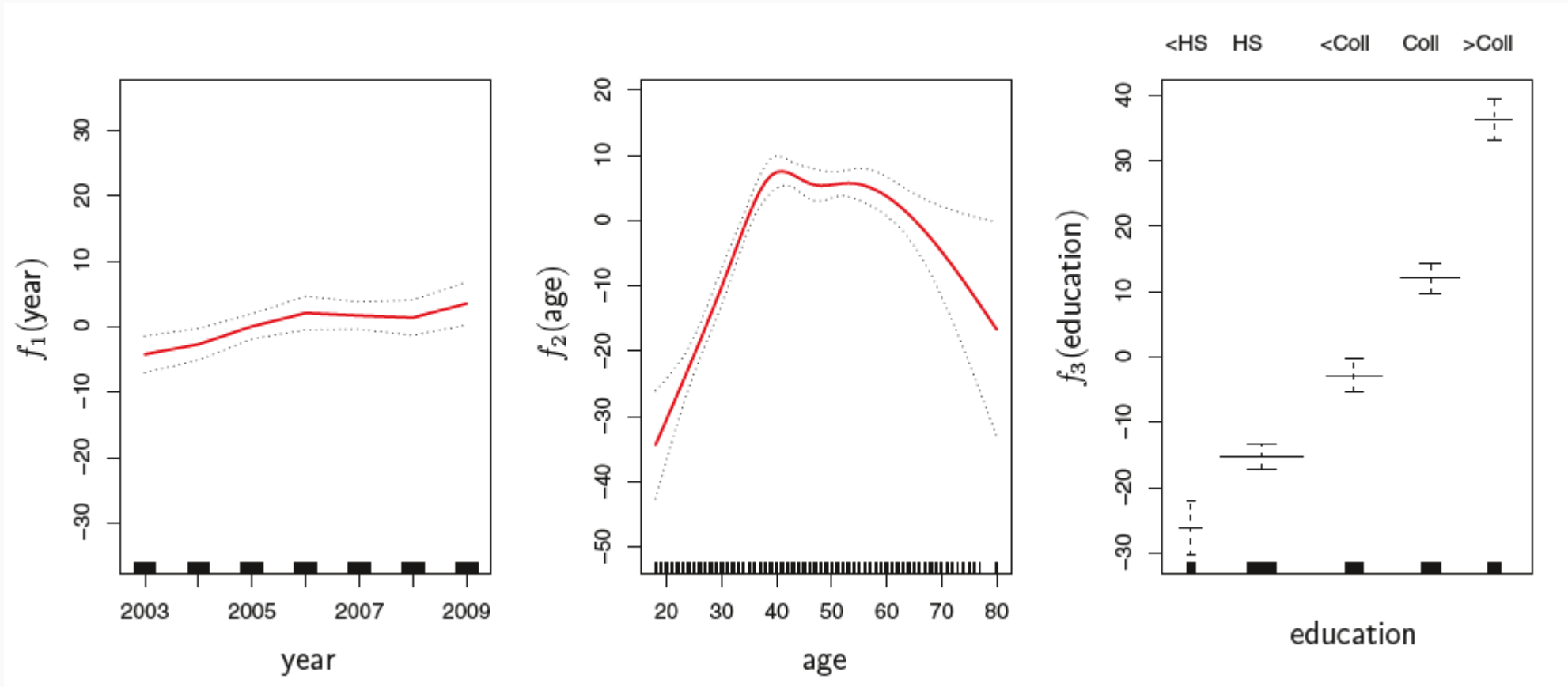
Burada $f_j(x_{ij})$ doğrusal olmayan düzgün bir fonksiyondur.

- Model toplamsal bir yapıya sahiptir çünkü her bir X değişkeni için ayrı bir doğrusal olmayan fonksiyon kullanılsa da sonuçta bunlar toplanır. Örneğin,

$$\text{wage} = \beta_0 + f_1(\text{year}) + f_2(\text{age}) + f_3(\text{education}) + \epsilon$$

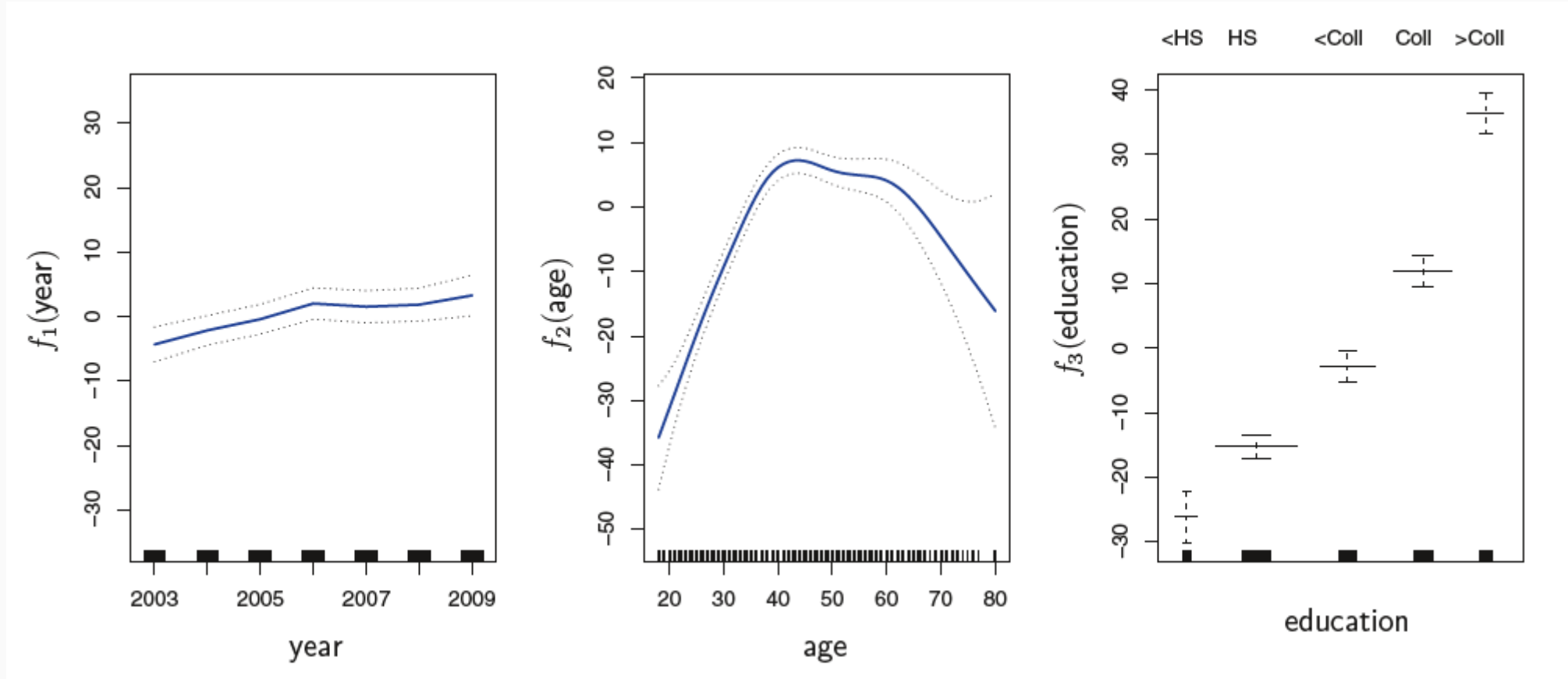
(izleyen grafiğe bkz.)

GAMs: Örnek



Year ve age: doğal spline (4 ve 5 serbestlik derecesi ile); education: adım fonksiyonu (Kaynak: ISLR Fig. 7.11, p.284)

GAMs: Örnek



Year ve age için düzleştirme spline'ları (4 ve 5 serbestlik dereceleri ile); education: adım fonksiyonu (Kaynak: ISLR Fig. 7.12, p.285)

GAMS ve Sınıflandırma problemleri

- GAM çıktı değişkeninin kategorik olduğu sınıflandırma problemleri için de kullanılabilir.
- Logit fonksiyonu parametrelerd doğrusaldır:

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$$

- GAM ile logit modeli aşağıdaki gibi yazılabilir

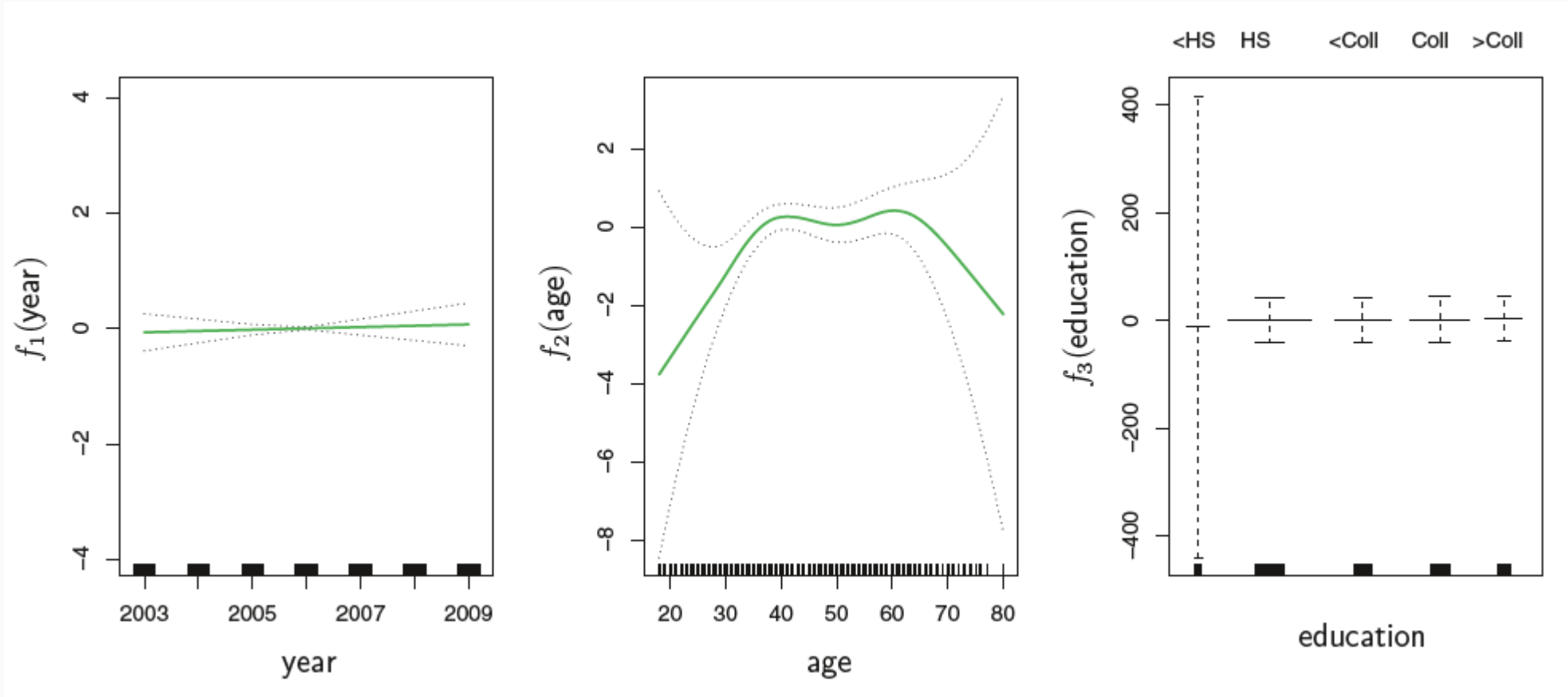
$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + f_1(X_1) + f_2(X_2) + \cdots + f_p(X_p)$$

- Örneğin

$$\log\left(\frac{p(X)}{1 - p(X)}\right) = \beta_0 + \beta_1 \times \text{year} + f_2(\text{age}) + f_3(\text{education})$$

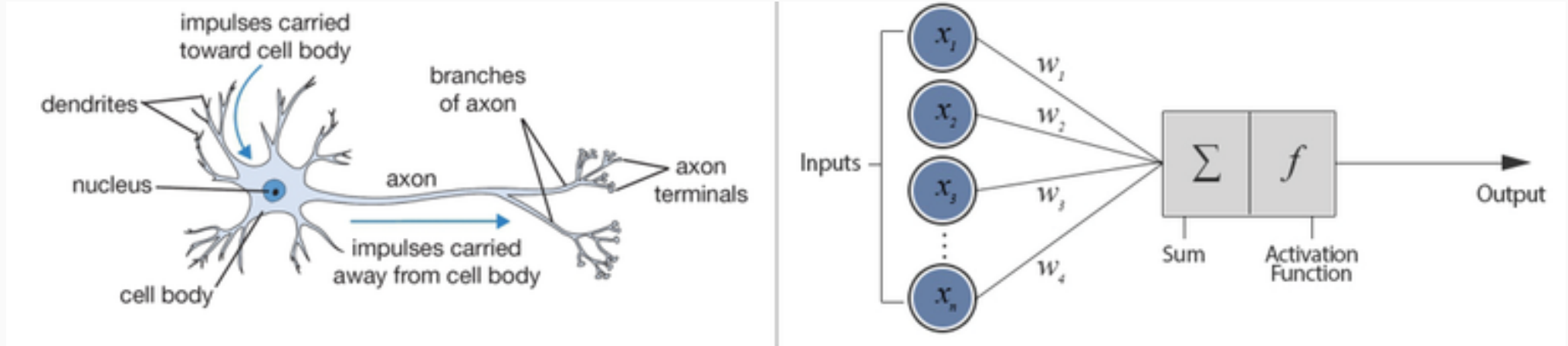
$$p(X) = \Pr(\text{wage} > 250 \mid \text{year}, \text{age}, \text{education})$$

GAM ve sınıflandırma: Örnek I(wage>250)



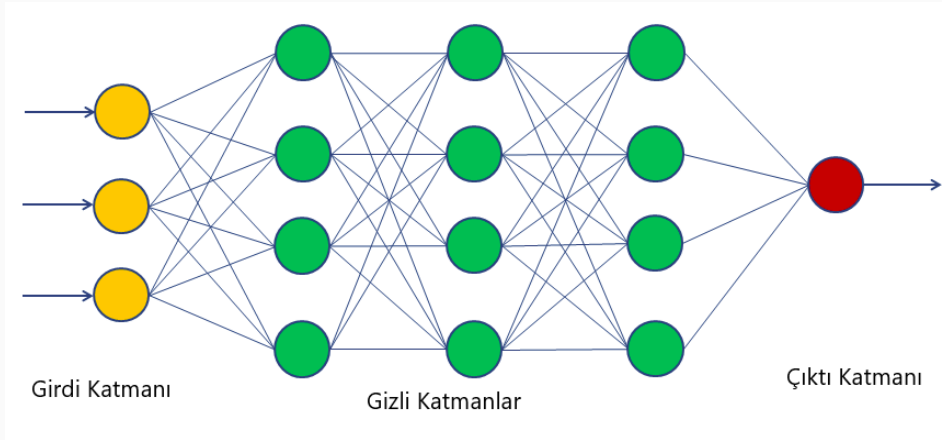
f_1 : doğrusal, f_2 : smoothing spline (df=5), f_3 : adım fonksiyonu. (Kaynak: ISLR Fig. 7.13, p.287)

Yapay Sinir Ağları (YSA)



- Yapay sinir ağları beyindeki sinir hücrelerinin (nöronların) çalışma prensiplerinden hareketle oluşturulan doğrusal olmayan ve oldukça esnek modellerdir.
- Biyolojik sinir ağları elektrik sinyallerini kullanarak bilgiyi iletebilen ve çıktı üreten nöronlardan oluşur.
- Benzer şekilde, bir YSA modeli, çok sayıda girdiyi bilinmeyen bir parametre vektörü ile ağırlıklandırarak çıktıya dönüştürür.
- Çıktı değişkeni birden fazla olabilir. Hem regresyon hem de sınıflandırma problemlerine uygulanabilir.

Yapay Sinir Ağları

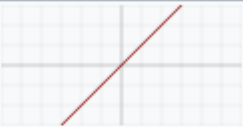


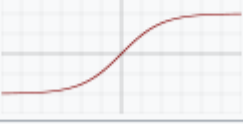



- Yapay sinir ağlarının yapısı bir şebeke grafiği ile stilize bir şekilde gösterilebilir.
- Tipik bir YSA üç ana katmandan oluşur: girdi katmanı (input), gizli katmanlar (hidden layers), ve çıktı katmanı (output)

- YSA çok katmanlı (multilayer) ya da tek katmanlı (single layer) olabilir.
- Bir YSA'nın bileşenleri şunlardır:
 - Girdiler: x_1, x_2, \dots, x_p
 - Ağırlıklar (bilinmeyen parametreler): w_1, w_2, \dots, w_p , ve w_0 (sabit)
 - Toplama fonksiyonu: $\sum_{j=1}^d w_j x_j + w_0$
 - Aktivasyon fonksiyonu: $h(\cdot)$
- Tek çıktılı basit bir YSA (perceptron - algılayıcı) aşağıdaki gibi yazılabilir:

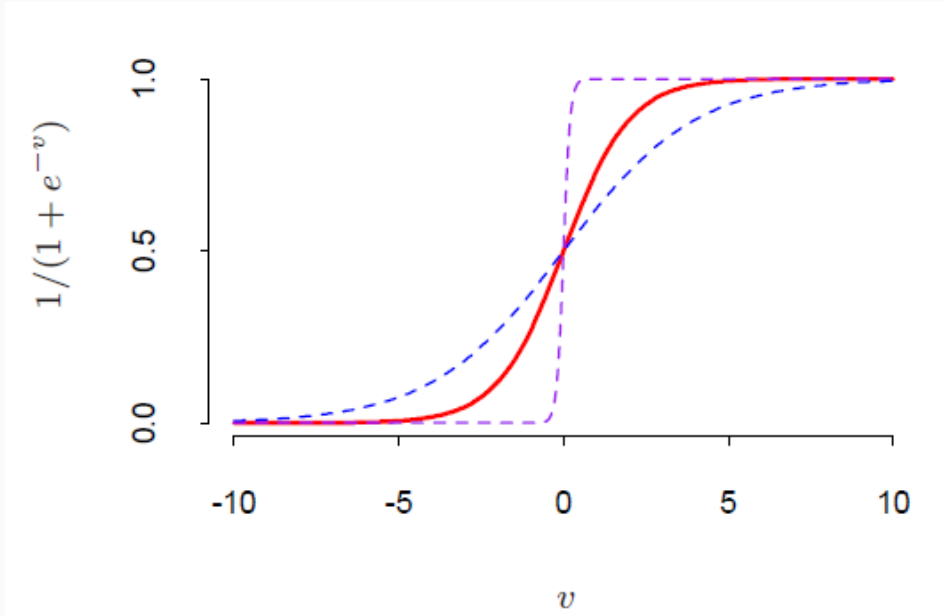
$$y = h \left(\sum_{j=1}^d w_j x_j + w_0 \right)$$

Aktivasyon Fonksiyonları

Name	Plot	Function, $f(x)$
Identity		x
Binary step		$\begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases}$
Logistic, sigmoid, or soft step		$\sigma(x) = \frac{1}{1 + e^{-x}}$ ^[1]
Hyperbolic tangent (tanh)		$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
Rectified linear unit (ReLU) ^[7]		$\begin{cases} 0 & \text{if } x \leq 0 \\ x & \text{if } x > 0 \end{cases}$ $= \max\{0, x\} = x \mathbf{1}_{x>0}$

- Aktivasyon fonksiyonları (Kaynak)

Sigmoid Aktivasyon fonksiyonu



- Sigmoid aktivasyon fonksiyonu (kırmızı):

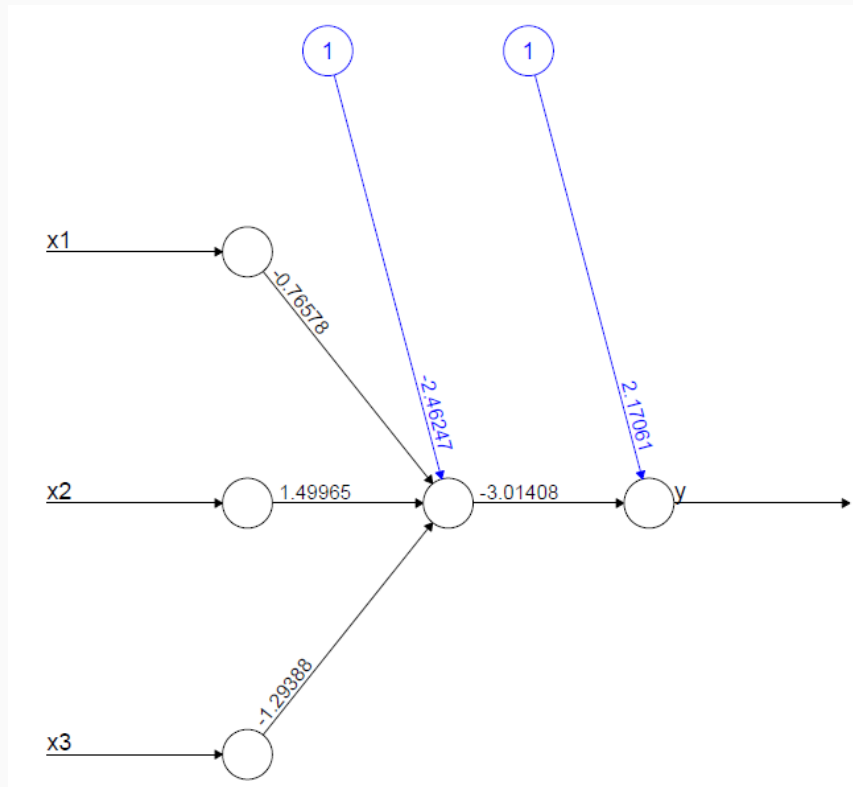
$$\sigma(v) = \frac{1}{1 + e^{-v}}$$

- Grafikte ayrıca $\sigma(sv)$ iki değer için gösterilmiştir: $s = 0.5$ (mavi), $s = 10$ (mor) (Kaynak: Elements of Statistical Learning, p.394)
- Aktivasyon fonksiyonu $\sigma = 1$ olduğunda (identity function) model girdilere göre doğrusal olur.
- Bir YSA aslında doğrusal sınıflandırma ya da regresyon problemlerinin doğrusal olmayan genelleştirmesi olarak düşünülebilir.

Örnek: Basit bir ileri beslemeli YSA

$$\hat{y} = -3,01408(-2.46247 - 0.76578x_1 + 1.49965x_2 - 1.29388x_3) + 2.17061$$

$$\hat{y} \approx 9.593 + 2.308x_1 - 4.52x_2 + 3.90x_3$$



- 3 girdi, tek katman, aktivasyon fonksiyonu: $h(z) = z$ (identity)
- Aslında bu OLS regresyonu ile aynıdır:

```
> regfit0 <- lm(y ~ x1 + x2 + x3, data=df)  
> regfit0
```

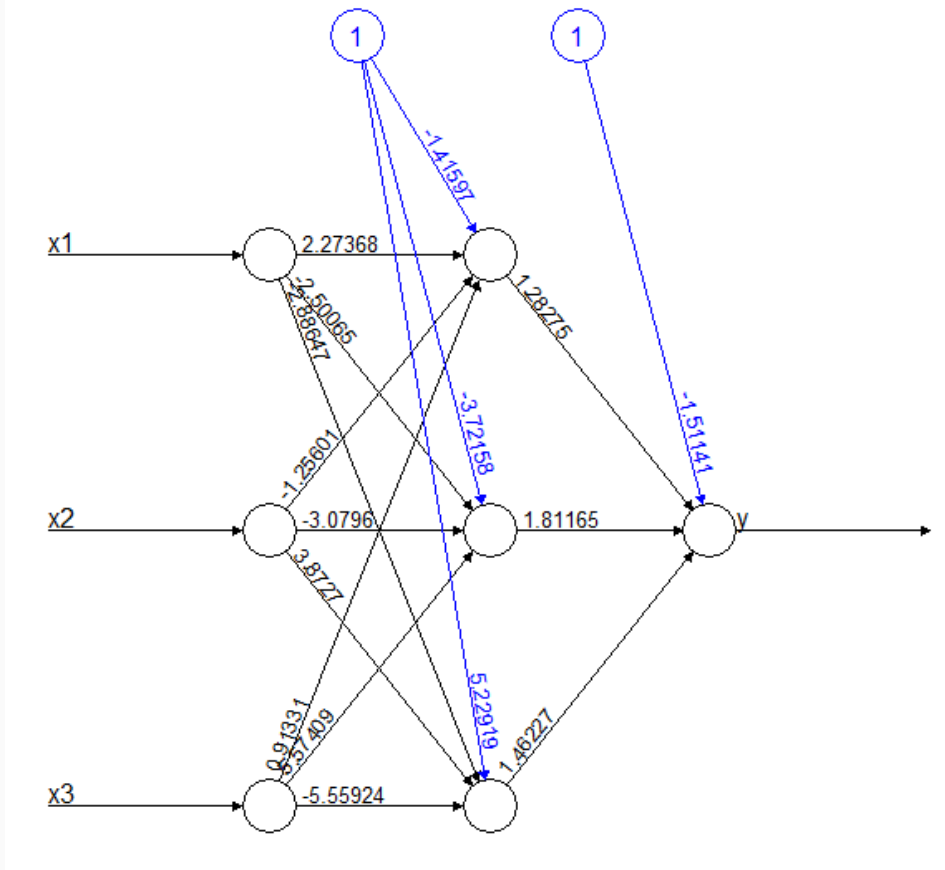
```
Call:  
lm(formula = y ~ x1 + x2 + x3, data = df)
```

Coefficients:

(Intercept)	x1	x2	x3
9.593	2.308	-4.520	3.900

- Bir yapay sinir ağacı x değişkenleri arasındaki etkileşimi açık olarak belirtmeden yakalayabilir.

Örnek



- Tek gizli katmanlı, üç hücreli bir ileri beslemeli yapay sinir ağı (mavi renkte gösterilenler sabitlerdir; YSA jargonunda "bias")
- Bu model değişken etkileşimlerine izin verir.

Yapay Sinir Ağlarının Eğitilmesi

- YSA modellerinde amaç fonksiyonu regresyon problemleri için kalıntı kareleri toplamı:

$$R(\theta) = \sum_{k=1}^K \sum_{i=1}^N (y_{ik} - f_k(x_i))^2$$

- Sınıflandırma problemleri için ise hata karesi ya da çapraz-entropi (deviance):

$$R(\theta) = - \sum_{i=1}^N \sum_{k=1}^K y_{ik} \log f_k(x_i)$$

θ : bilinmeyen ağırlık vektörü, y_{ik} : gözlem değerleri, f_k : k çıktısı için tahmin değeri, K

- Minimum $R(\theta)$ değerini veren ağırlıklar gradyan iniş algoritması ile bulunabilir (gradient descent). Bu algoritma geri-yayılım algoritması olarak da bilinir (back-propagation algorithm).
- Gradyan (birinci türev vektörü) zincir kuralı ile kolayca hesaplanabilir.
- Fazla uyumdan kaçınmak için genellikle gradyan vektörü küçük pozitif bir parametre ile çarpılır. Böylece öğrenme hızı kontrol edilebilir.
- Algoritmanın detayları için bkz. ss. 395-397, Hastie, Tibshirani, Friedman (2017),