Model Değerlendirme ve Geçerleme

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

Hüseyin Taştan
Yıldız Teknik Üniversitesi

Plan

- Test Hatasının Tahmini
- Geçerleme (Validation)
- Çapraz Geçerleme (Cross Validation)
- Biri-Hariç Çapraz Geçerleme
- k-Katlı Çapraz Geçerleme
- Bootstrap

Eğitim ve Test Hatası

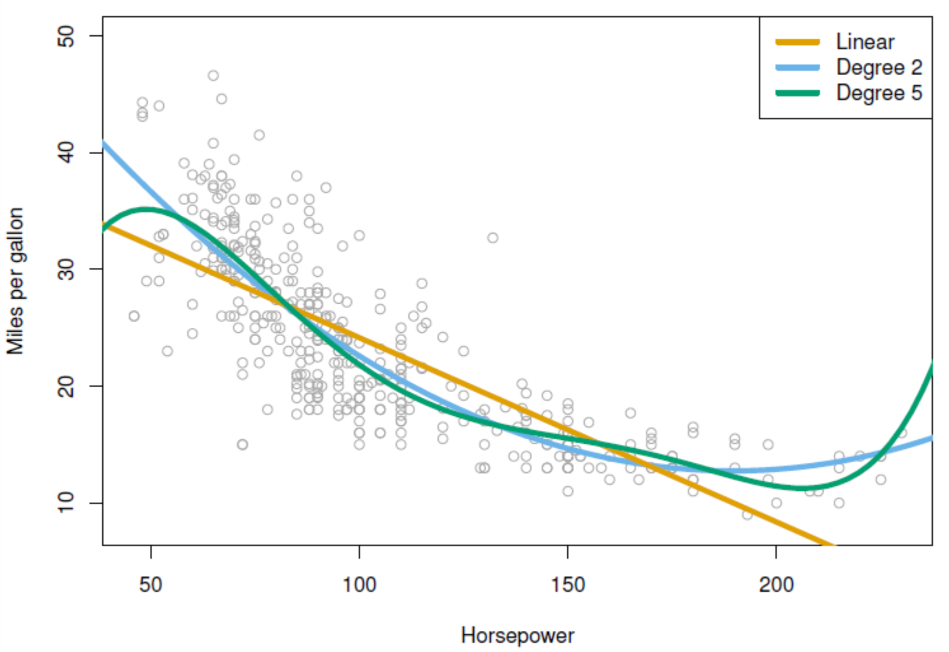
Örnek: Polinom regresyonu

Örnek olarak aşağıdaki polinom regresyonu düşünelim:

$$y = \beta_0 + \beta_1 x + \beta_2 x^2 + \ldots + \beta_p x^p + \epsilon$$

(Dataset = AUTO, y=mpg (miles per gallon), x=horsepower)

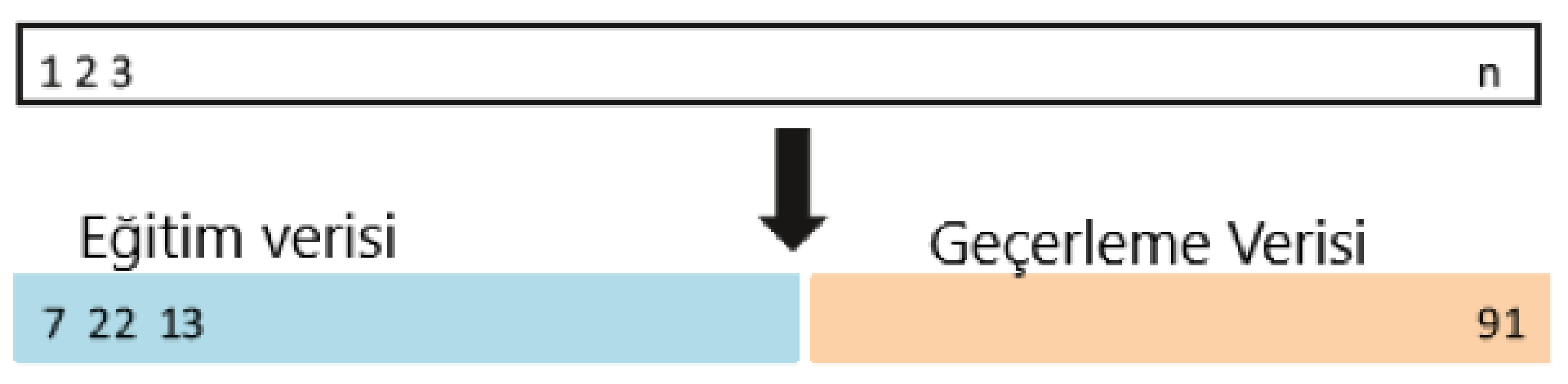
- En iyi kestirimleri veren (en küçük MSE değerine sahip) p (polinom derecesi) kaçtır?



Test Hatası Nasıl Hesaplanır? Geçerleme Yaklaşımı

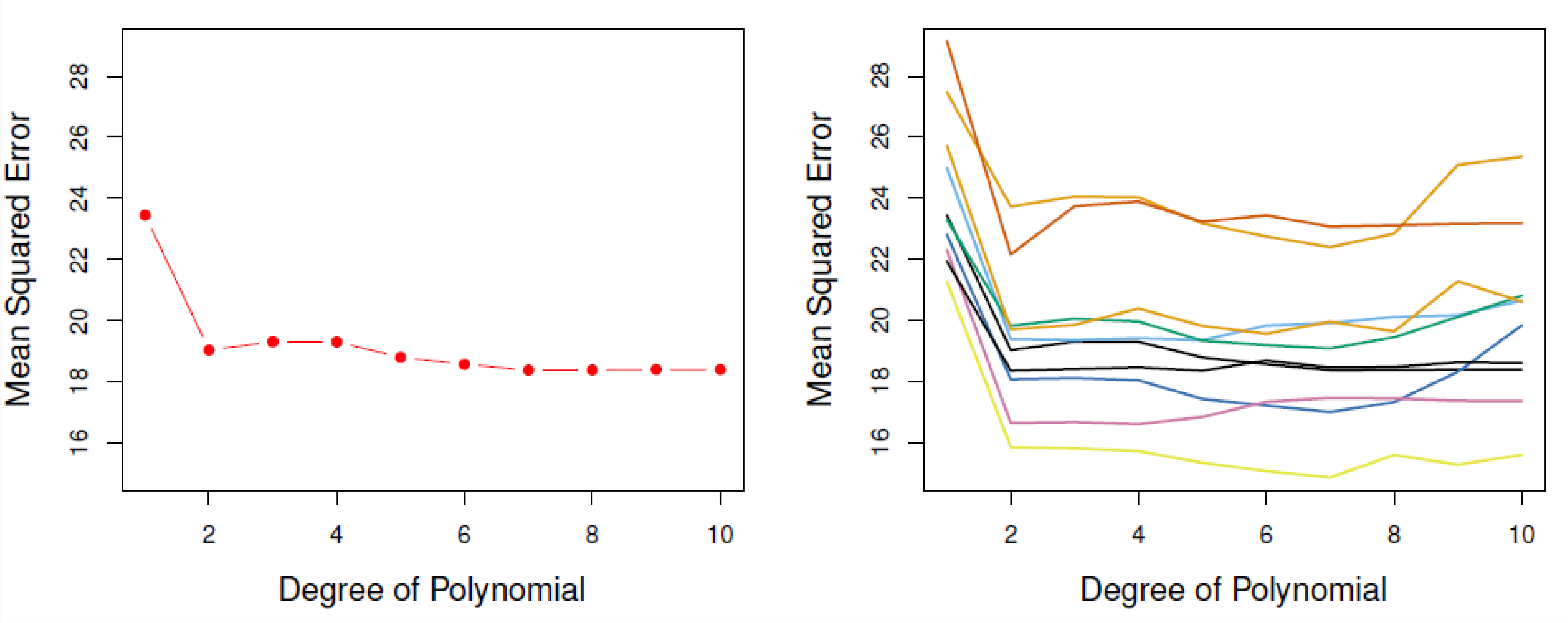
- Pratikte genellikle elimizde sadece bir veri seti vardır. Modelin kestirim başarısını ölçebileceğimiz ayrı bir test veri seti genellikle yoktur.

- Bu durumda verileri rassal olarak iki gruba ayırabiliriz: eğitim verisi ve geçerleme (validation, hold-out) verisi



- Model sadece eğitim verileriyle tahmin edilir. Geçerleme verileri kullanılarak kestirimler oluşturulur ve test hatası hesaplanır.

Örnek: Polinom regresyonu



- Veri seti rassal olarak ikiye bölündü ve her seferinde geçerleme verilerinden hareketle her polinom derecesi için MSE hesaplandı. Solda: tek geçerleme seti
- Sağda: 10 kere tekrarlanmış geçerleme MSE değerleri. Fazla değişkenlik olduğuna dikkat ediniz.

Geçerleme yaklaşımı: Zayıf tarafları

- Veriler rastgele ikiye ayrıldığı için buradan kaynaklı belirsizliği dikkate almamız gerekir. Bunun için süreci tekrar ederek çok sayıda rastgele geçerleme tahminleri yaptığımızda da, önceki grafikte görüldüğü gibi, yüksek değişkenlik gözlemlenmektedir.

- Geçerleme yaklaşımında sadece eğitim verilerinin model tahmininde (eğitiminde) kullanıldığı unutulmamalıdır. Geçerleme veri seti her bir model için (örneğimizde her bir p için) sadece kestirim hatalarının hesaplanmasında kullanılır. Eğitim setinde daha az veri kullanıldığı için modellerin performansı azalır. Sonuçta geçerleme hatası test hatasını olduğundan yüksek tahmin edebilir.

- Alternatifler

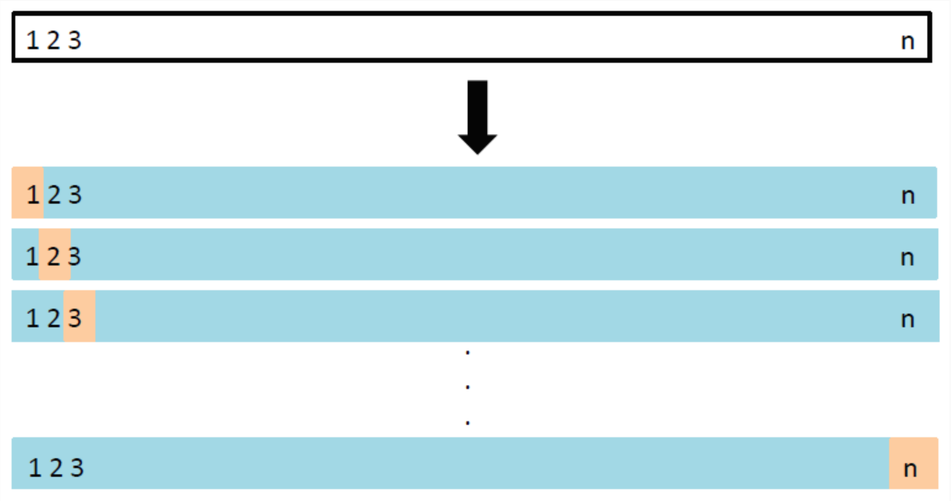
  - Biri-hariç Çapraz Geçerleme (Leave-one-out Cross Validation)
  - k-katlı Çapraz Geçerleme (k-fold Cross Validation)

Biri-hariç Çapraz Geçerleme

LOOCV (Leave-one-out Cross Validation)

- Gözlemlerden sadece biri geçerlemede kullanılır; geriye kalan (n-1) gözlem modelin eğitiminde kullanılır.
- Bu süreç her seferinde bir gözlem eğitimden dışlanacak şekilde n kere tekrarlanır ve her biri için MSE_i elde edilir.
- Bu n MSE değerinin ortalaması test hata tahminidir:

$$\text{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n}\text{MSE}_i$$

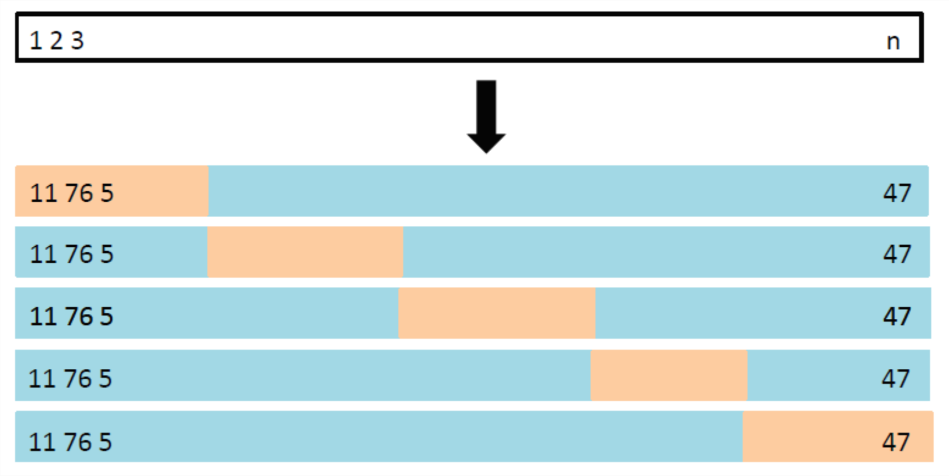


k-Katlı Çapraz Geçerleme

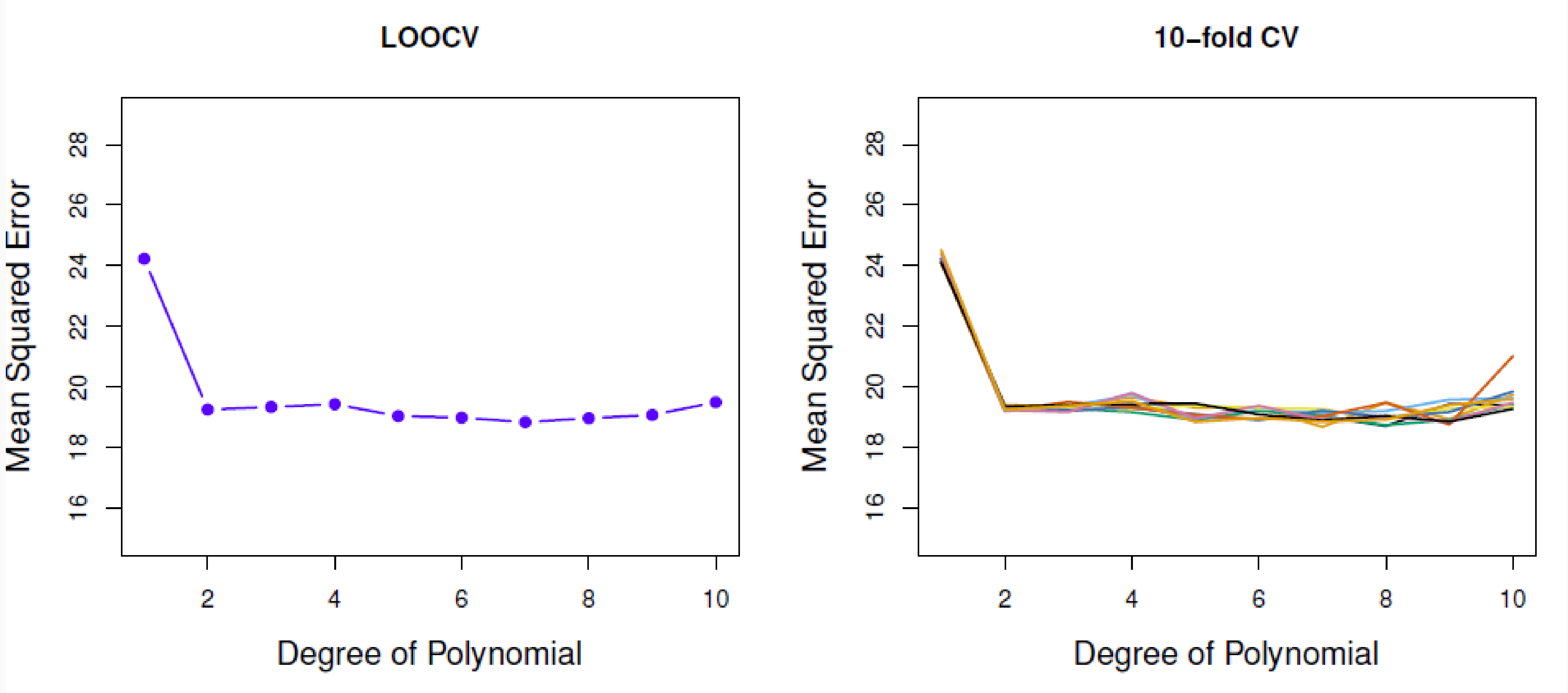
k-Fold Cross Validation

- Biri-hariç çapraz geçerleme n büyük olduğunda hesaplamada zorluk çıkarabilir.
- Alternatif olarak gözlemler rassal şekilde k gruba (kat) ayrılabilir.
- Sırasıyla her kat geçerleme seti olarak kullanılır; geriye kalan gözlemlerle model eğitilir.
- Sonuçta elimizde k tane MSE değeri vardır. Test hata tahmini bunların ortalamasıdır:

$$\mathrm{CV}_{(k)} = \frac{1}{k}\sum_{i=1}^{k} \mathrm{MSE}_i$$



Örnek



Solda: Auto veri seti için (bkz önceki örnek) Biri-hariç Çapraz Geçerleme MSE değerleri, **Sağda**: $k = 10$ katlı Çapraz Geçerleme (Kaynak: James et al., ISLR Fig-5.4, p.180)

k-Katlı Çapraz Geçerlemede Sapma-Varyans Ödünümü

- k-Katlı çapraz geçerleme (ÇG) biri-hariç çapraz geçerlemeye göre hesaplama açısından avantaj sağlar.
- Ancak asıl önemli avantaj test hatasının biri-hariç çapraz geçerlemeye (LOOCV) göre daha doğru tahmin edilmesidir.
- Çapraz geçerleme test hatasını fazla tahmin etme eğilimlidir. LOOCV ise test hatasını sapmasız tahmin eder. k-katlı ÇG bu ikisinin arasında bir yerdedir.
- Bu açıdan bakıldığında her seferinde LOOCV'yi tercih etmemiz gerekir.
- Ancak k-katlı çapraz geçerlemenin varyansı biri-hariç ÇG'ye göre daha düşüktür.
- Bunun sebebi LOOCV'de test tahminlerinin birbiriyle çok yüksek ilişkili olmasıdır.

Sınıflandırma Problemlerinde Çapraz Geçerleme

- Çapraz Geçerleme yaklaşımı, çıktı değişkeni Y'nin nitel olduğu sınıflandırma problemlerinde de kullanılabilir.
- Bu durumda, daha önce gördüğümüz gibi, sınıflama hatasını

$$Err_i = I(y_i \neq \hat{y}_i)$$

olarak tanımlarsak Biri-Hariç Çapraz Geçerleme test hatası aşağıdaki gibi tanımlanabilir:

$$\mathrm{CV}_{(n)} = \frac{1}{n}\sum_{i=1}^{n} Err_i$$

- Benzer şekilde k-Katlı ÇG:

$$\mathrm{CV}_{(k)} = \frac{1}{k}\sum_{i=1}^{n} Err_i$$

Zaman Serisi Verileri

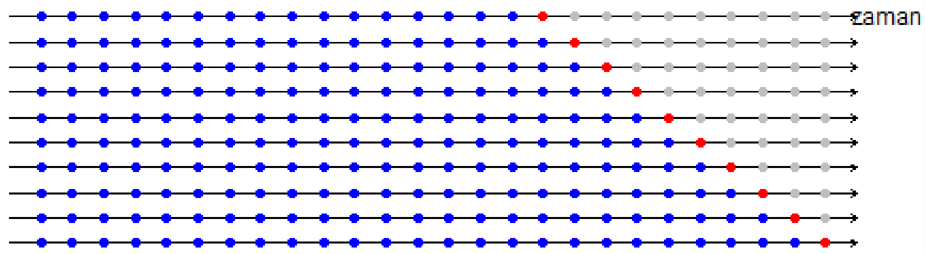
- Zaman serisi verileriyle öngörü modellerin örneklem-dışı (out-of-sample) öngörü başarısı değerlendirilirken iki yaklaşım benimsenebilir.
- Geleneksel Yaklaşım ve Çapraz Geçerleme Yaklaşımı
- Zaman serileri genellikle türdeş ve bağımsız (iid) olmaz. Ayrıca verilerdeki kronolojik yapının bozulmaması gerekir. Bu nedenle rassal örneklemeyle çapraz geçerleme yapamayız.

Geleneksel Yaklaşım

- Veriler eğitim ve test kısımlarına ayrılır. Test verileriyle öngörü hataları hesaplanır.

Çapraz Geçerleme

Zaman Serisi Çapraz Geçerleme



- Öngörüler Biri-Hariç çapraz geçerlemede olduğu gibi bir test gözleminden hareketle hesaplanır.
- İzleyen adımda bir önceki test gözlemi eğitim setine eklenir ve model tekrar tahmin edilir. Bu modelden hareketle yeni bir öngörü oluşturulur.
- Tüm test verileri için aynı işlem yapılır. En sonunda öngörü hatalarının ortalaması hesaplanır.

Detaylar için bkz. Hyndman, R.J., & Athanasopoulos, G. (2019) Forecasting: principles and practice, 3rd edition, OTexts: Melbourne, Australia. https://otexts.com/fpp3/

Bootstrap

- Bootstrap özellikle istatistiksel tahmincilerin özelliklerini değerlendirmede sıklıkla kullanılan bir yeniden örnekleme yöntemidir.
- Bootstrap örneklemesinin özünde mevcut veri setinden yerine koyarak yönelemeli örnekleme yapılması bulunmaktadır.
- Elimizdeki veri setinden yerine koyma usulüyle tesadüfi örnekleme yapıldığı için bazı değerlerin tekrar etme şansı vardır.
- Özelliklerini incelemek istediğimiz istatistik her bir bootstrap örneklemi için tahmin edilir (örneğin B=1000 kez).
- Sonunda ilgili tahmincinin örnekleme dağılımı yaklaşık olarak oluşturulabilir.

Bootstrap

- En kolay bootstrap yöntemi iyi tanımlı bir anakütleden birbirinden bağımsız ve türdeş dağılmış (identically and independently distributed - iid) bir veri seti üzerinden tanımlanabilir.
- Pratikte elimizde ilgili anakütleden çekilmiş sadece bir örneklem bulunur.
- İstatistiksel çıkarsamada standart yaklaşım ilgili tahmicinin örnekleme dağılımının oluşturulmasına dayanır.
- Teorik olarak bir örnekleme dağılımı ilgili anakütleden çekilebilecek tüm örneklemlerin bilgisiyle oluşturulan soyut/teorik bir kavramdır.
- Çoğu tahminci için belli varsayımlar altında en azından büyük örneklemlerde normal dağılıma uyar.

Bootstrap

- Küçük örneklemlerde ise örnekleme dağılımı normalden sapabilir.
- İşte bu durumda Bootstrap yöntemi örnekleme dağılımının yaklaştırılmasında çok faydalı olabilir. Standart analize göre daha az varsayıma dayanır ve ayrıca normallik varsayımı gerekmez.
- Daha önce belirttiğimiz gibi Bootstrap yöntemi aslında mevcut örneklem sanki anakütleymiş gibi yeniden örnekleme yapar. Sonuçta amaç örnekleme dağılımının yaklaşık olarak tahmin edilmesidir.
- Bootstrap yöntemi özellikle standart hataların ve güven aralıklarının tahmininde yaygın olarak kullanılır.

Örnek: Örneklem ortalamasının örnekleme dağılımı

```r
set.seed(12345)
n <- 10
x <- rnorm(n, mean=0, sd=1) # anakütle standart normal
xbar <- mean(x)
se_xbar <- 1/sqrt(n)  # teorik standart hata
se_xbar_est <- sqrt(var(x)/n) # örneklem standart hatası
# Bootstrap standart hatasının bulunması
# Tek bootstrap örneklemi için:
x_boot1 <- sample(x, n, replace = TRUE) # yerine koyarak örnekleme
x_boot1
```

```
##  [1] -0.2761841  0.6300986 -1.8179560  0.5855288 -0.4534972 -0.2761841
##  [7] -0.9193220 -0.1093033 -0.2841597 -0.4534972
```

```r
# boot1 için örneklem ortalaması
xbar_boot1 <- mean(x_boot1)
xbar_boot1
```

```
## [1] -0.3374476
```

Örnek

```r
B <- 500 # Bootstrap yineleme sayısı
xbar_boot <- numeric(B)
for (i in 1:B) {
  xbar_boot[i] <- mean(sample(x, n, replace = TRU
}
sd(xbar_boot) # bootstrap std hatası
```

```
## [1] 0.2321545
```

```r
1/sqrt(n) # theoretical std error (sigma/sqrt(n))
```

```
## [1] 0.3162278
```

```r
sqrt(var(x)/n) # sample std error
```

```
## [1] 0.2573004
```

```r
hist(xbar_boot)
```