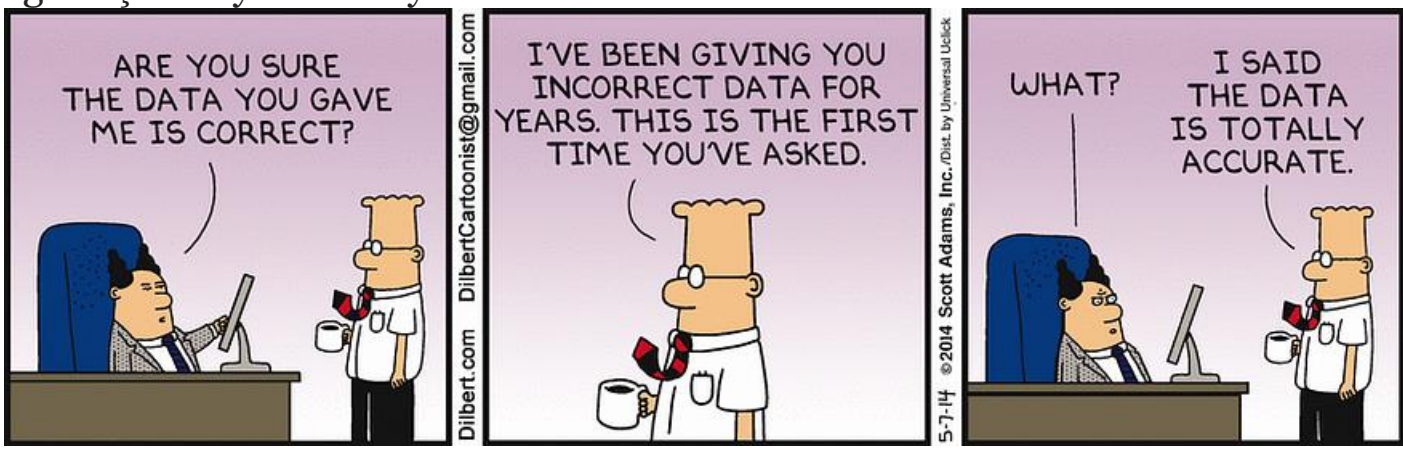
Model Eğitmek (Training)

Makine Öğrenmesi modellerinin geliştirilmesinde eğitilen modelin yeni veya daha önce karşılaşmadığı veriler üzerinde iyi performans göstermesi arzu edilir. Yeni/karşılaşılmayan verileri simüle etmek için mevcut verilerimizi eğitim ve test veri seti olarak 2'ye ayırıyoruz. Özellikle, ilk bölüm eğitim seti olarak kullanılan veri daha büyük veri alt kümesidir (orijinal verilerin% 80'ini hesaba katmak gibi), ikincisi ise daha küçük bir alt kümedir ve test seti olarak kullanılır (geri kalan% 20 veri).

Örneğin, algoritmaya biz bir veri veriyoruz ve bu veri üzerinden algoritma bir yapı öğreniyor (ev fiyat tahmin modelleri gibi). Bir diğer deyişle faktörlerin etkilerini, bu etkilerin yönlerini öğreniyor. Öğreterek oluşturmuş olduğumuz bu algoritmaların başarısını test etmek için de veri setimizi 2'ye ayırıyoruz. 1000 gözlemlik veri seti var diyelim, 800'ü eğitmek için ayırıyoruz ve kalan 200'ü model eğitmiş mi diye test ediyoruz.



Değişken Seçimi

Modelleme çalışmaları sırasında veri setimizin büyüklüğüne göre elmizde 5,10 hatta 100 tane bağımsız değişkenimiz olabilir. Bu

bağımsız değişkenlerle Y bağımlı değişkenini tahmin etmeye çalışacağız. Modelleme çalışmalarında değişkenlerin hepsini modelde tutmaya çalışmayız, amaç en az değişkenle, en fazla açıklanabilirliği yakalamaya çalışmaktır.

Model Seçimi

Temelde iki yöntem ön plana çıkmaktadır.

- Oluşabilecek değişken kombinasyonları ile oluşturulan modeller arasında en iyi modelin seçilmesini sağlamaya çalışıyoruz.
- Kurulan birbirinden farklı modeller arasından model seçmeye çalışıyoruz.

Model Neye Göre Seçilir?

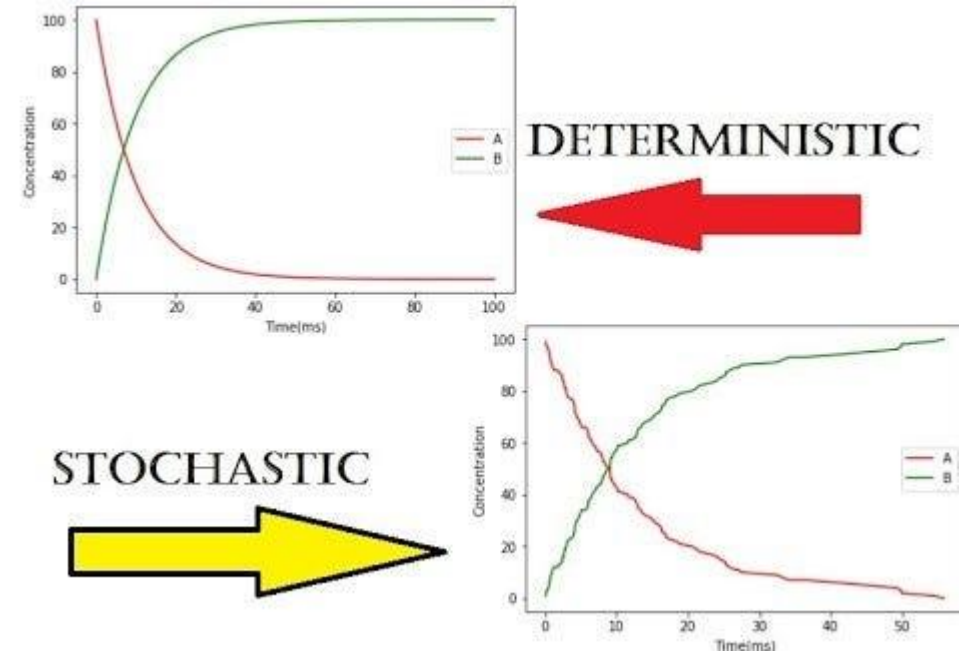
- Regresyon problemleri için açıklanabilirlik oranı ve RMSE (hata ölçüm metriği) türevi bir değer kullanılır.
- Sınıflandırma problemleri için doğru sınıflandırma oranı (model başarısını değerlendirme metriği) türevi bir değer kullanılır.

Aşırı Öğrenme (Overfitting)

Veri setinin algoritmanın kendisine verilen veri setinin yapısını çok iyi düzeyde öğrenip hatta ezberleyip görmediği yeni veri setleri üzerinde tahmin yapmak istendiğinde başarısız olma durumudur.

Bir diğer deyişle, veriyi test ve train olarak 2'ye ayırıyoruz. Algoritma eğitim setini çok iyi öğreniyor. Ancak görmediği veri setiyle modeli tahmin etmeye çalıştığımızda tahmin performansı düşmeye başlıyor. Bu duruma aşırı öğrenme (overfitting) denmektedir.

Deterministik Modeller vs Stokastik Modeller

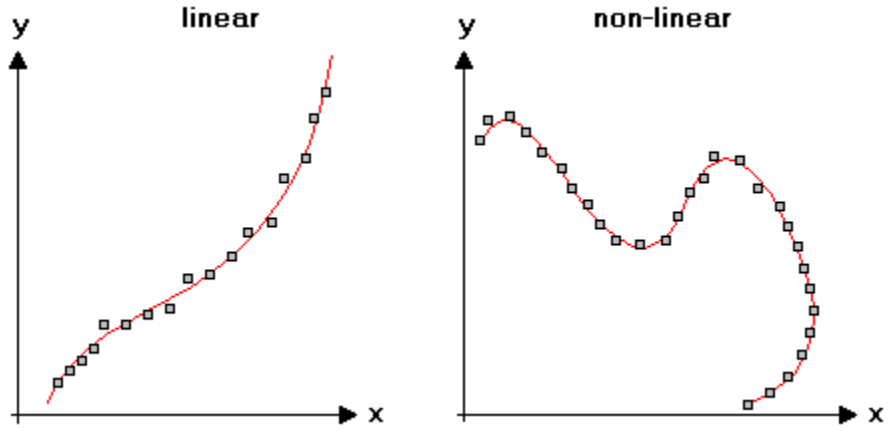


Deterministik modeller, değişkenler arasında kesin bir ilişki olduğunu varsayan modellerdir. İki değişken arasındaki ilişki bir doğru ile ifade edilir.

Stokastik modeller ise olasılıksal modellerdir. Burada tesadüfi hata mevcuttur.

Yukarıdaki grafikte de gördüğümüz gibi, stokastik modellerde, X ile Y arasındaki ilişkiyi tahmin etmeye çalıştığımızda doğrusal olarak ifade edemiyoruz. Bir hata payı mevcut oluyor.

Doğrusal vs Doğrusal Olmayan Modeller



X ile Y arasındaki ilişki bir doğru ile ifade edilirse doğrusaldır. Bir doğru yerine eğri aracılığıyla veya ağaca dayalı yöntemlerle veya diğer farklı yöntemlerle değişkenler arasındaki ilişkiler modellenmeye çalışılırsa doğrusal olmayan yöntemler denilmektedir.

Makine öğrenmesi basitçe anlatmak gerekirse matematikten, istatistiğe geçiş sürecidir.

Matematik kesinlik, istatistik olasılık içerir. İstatistikte kesinlik yoktur, hep bir hata ve tahmin etme işlemi vardır.

Model Doğrulama (Model Validation) Yöntemleri

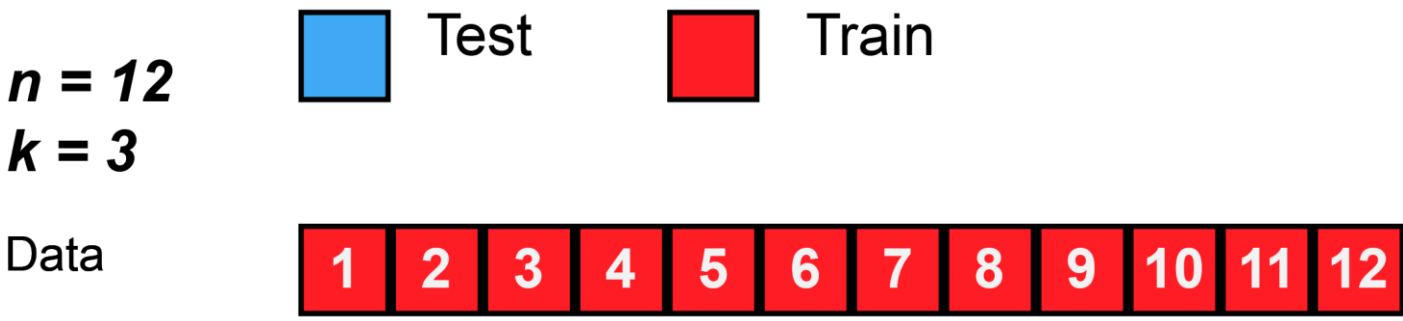
Bağımlı ve bağımsız değişkenler arasındaki ilişki bulmak için model kurarız. Örneğin, tahmin etmek istediğimiz bağımlı değişkenimiz evlerin fiyatı ve onu oluşturan bağımsız değişkenlerimiz, evin büyüklüğü, konumu, bulunduğu kat vs. Modeli kurduktan sonra model ürettiği sonuçlarını değerlendirmemiz gerekir. Bu çalışmalara model doğrulama yöntemleri denir. Regresyon modellerinde farklı, sınıflandırma modellerinde farklı yöntemler kullanılır.

Holdout Yöntemi (Sınama Seti)

Orijinal bir veri seti var diyelim. 1000 gözlemlik veri setini %80 — %20 olarak eğitim ve test seti gibi bölüyoruz. 800 gözlemle eğitiyoruz ve 200 gözlemle test ediyoruz. Örneğin ev fiyat tahmin modelindeki katsayıları eğitim seti ile öğreniyoruz sonra acaba bu tahmin ne kadar iyi olduğunu 200 gözlemle test ediyoruz.

Holdout yönteminde gözlem sayımız az olursa veri setini eğer eğitim ve test seti olarak ayıramayabiliriz. Örneğin gözlem sayımız 50 olduğunda veriyi eğitmek ve test etmek için bölmeyebiliriz.

K-Katlı Çapraz Doğrulama (K Fold Cross Validation) Yöntemi



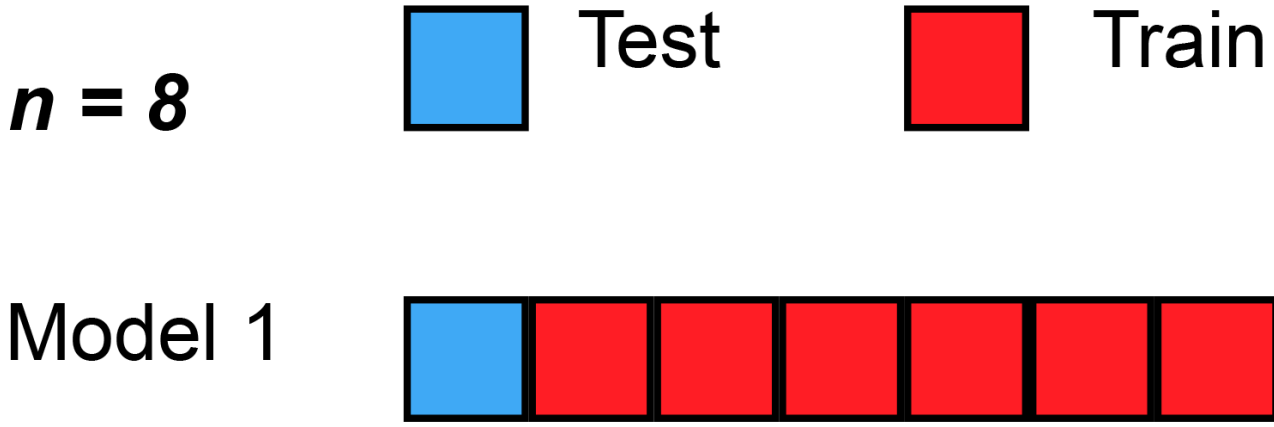
Veri seti k adet parçaya ayrılır. Daha sonra belirlenen alt kümlerden birisi dışarıda bırakılır. Elde kalan diğer kümeler ile model oluşturulur ve model dışarıda bırakılan küme ile test edilir. Bu çalışma bütün parçalar için tekrar edilir.

Elde edilen hataların ortalaması alındığında bu bizim validasyon/doğrulama (eğitim) hatası olur. Daha sonra çalışmanın en başında böldüğümüz test seti ile modelimizi test ederiz.

Elimizdeki veri setini her zaman ikiye ayırmamız gerekiyor. Test ve Train seti olarak. Bu ayırma işleminden sonra K fold yöntemini her zaman eğitim seti üzerinden yapmamız gerekir. Eğitim seti üzerinden doğru bir eğitim hatası hesaplayıp bunu kenarda tutup, oluşturmuş olduğumuz modeli en son bir de test seti üzerinde test etmemiz gerekiyor.

Özetle, elimizde orijinal bir veri seti olur. Bu veri setini %80-%20 test-train olarak 2'ye böleriz. Bu %80lik train üzerinden validasyon işlemi yaparız, yani doğrulama işlemi yaparız. Bunu 5 veya 10 katlı olacak şekilde gerçekleştirip, buradan bir model kurup, modele ilişkin bir eğitim hatası elde edip bunun üzerinden test seti üzerinden yeni bir test etme işlemi gerçekleştirilir.

Leave One Out Yöntemi

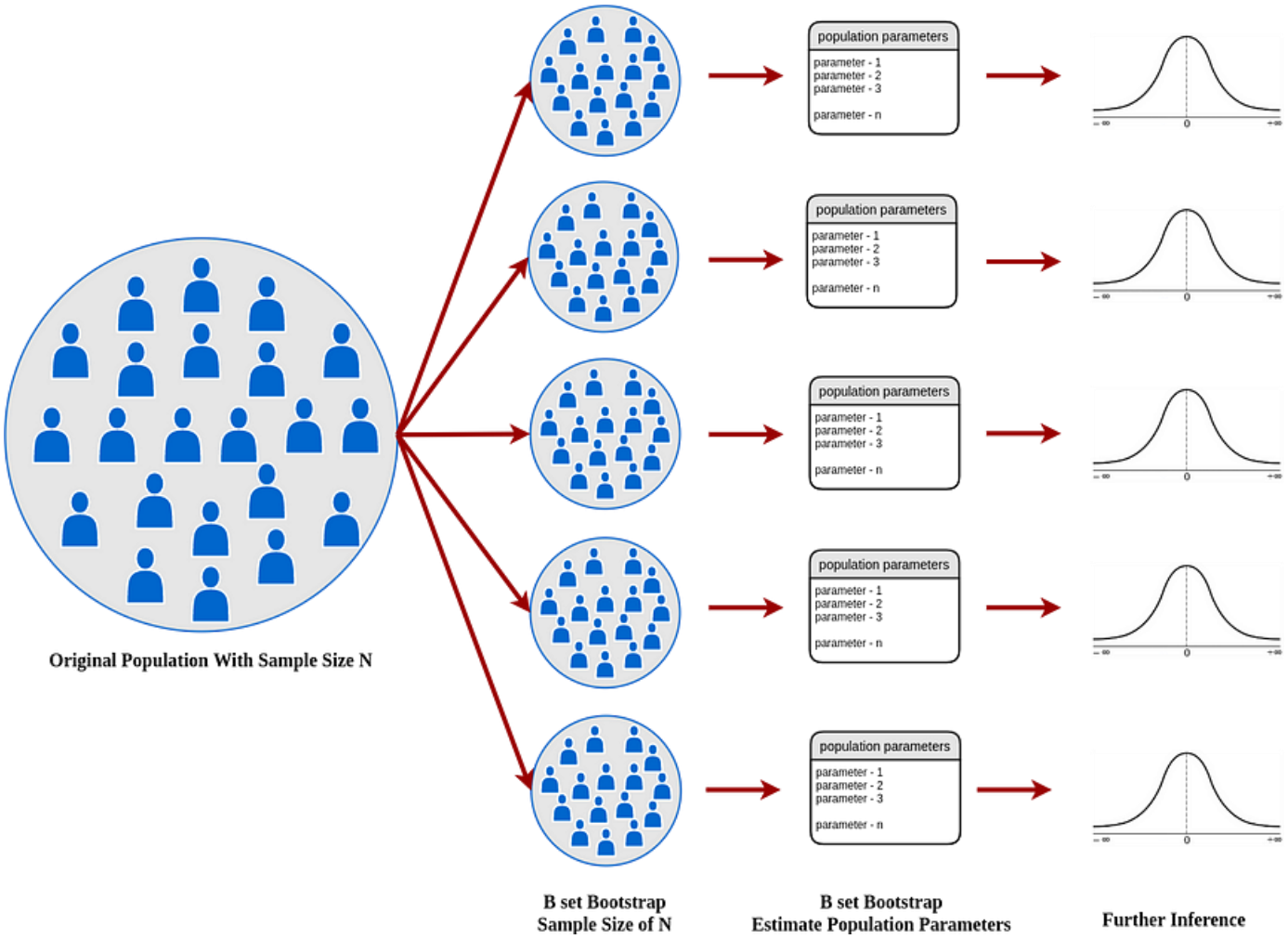


K katlı yöntemin özel bir halidir. K'da veri setini 5–10 parçaya bölüp, her iterasyonda bir parçayı dışarıda bırakıp diğer parçalarla model kurup test etmek için dışarıda bıraktığımız parçayı kullanmıştık. Burada ise K küme sayısı kadar veri setindeki örnek sayısı n'e eşittir. Yani n tane kadar küme varsayılır.K gibi sırasıyla hepsi test edilir.

Örneğin 1000 tane gözlemimiz olsun. Her seferinde 999 tane gözlem birimi ile model kurulup bir gözlem birimi test edilir. İkinci iterasyonda bir başka gözlem birimi dışarıda bırakılıp diğer tüm gözlemlerle model kurulur ve dışarıda bırakılan test edilir. Böylelikle tüm veri seti gezilmiş olur.

Bu yöntem teorik olarak güzel ancak veri seti büyüdükçe uygulanması zordur.

Bootstrap Yöntemi



Diğer yöntemlere benzer olarak; veri setini bir şekilde 2'ye bölelim, bir kısmıyla modeli eğitelim, diğer kısmı ile test edelim gibi yaklaşımlara dayanıyor.

Bootstrap bu yöntemlere ek olarak yeniden örnek oluşturacak şekilde çalışıyor.

Örneğin, elimizde orijinal veri seti var. Bu veri seti içerisinden, veri seti gözlem sayısından daha az olacak şekilde bootstrap örnekleri oluşturulur. Bootstrap1, Bootstrap2, Bootstrap3.. gibi örneğin 10 adet olsun. Bu 10 adet veri üzerinden model kurulur. Kurulmuş olan

modeller test seti yaklaşımı ile test edilip, train ve testlerin ortalaması alınarak sonuçlar değerlendirilmiş olru.

Özetle; yerine koymalı bir şekilde veri stei içerisinden veri türetmek olarak kullanılır ve oluşan yeni verilerin her birisi üzerinden model kurulur, bu modeller test edilir ve buna göre sonuçlar değerlendirilir.