**ML 06.07.2023**

**Recap:**

**Linear regresyonda--> feature’larin target uzerindeki etkisini olcuyoruz.**

**Simple linear reg.--> tek feature**

**Bir datayi gorsel olarak en iyi tanimanin yolu pairplottur.**

**Korelasyon yoksa regresyondan soz edemeyiz.**

**Correlation bize iliskiyi gosterir. Etkiyi ancak regresyon ile tespit edebiliriz.**

**R2-->feature’in bagimli degiskendeki degismin ne kadarini aciklayabildigni gosteriyor.**

**En az 30 adet gozlem olmali bir feature icin.**

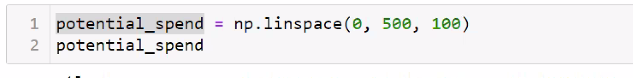
**Slope da intercept de any deger alir ama slope 0 olursa, regresyon cizgisi x eksenine paralel olur, bu da bir korelasyon yok demektir. Korelasyon yoksa regresyon da yok demektir. Dolayisiyla 0 haric denebilir.**

**Feature--> X, target-->y**

**LSE-->**

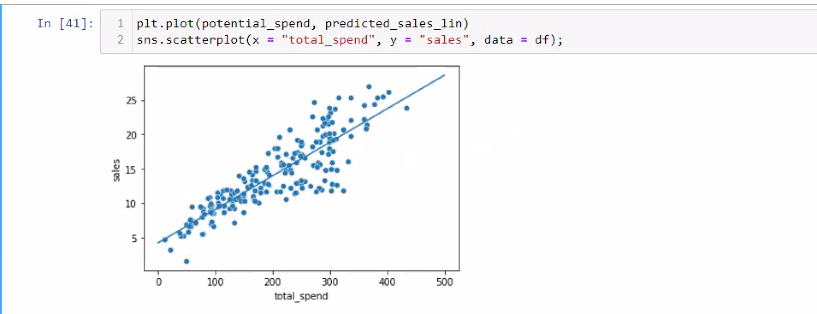
**Regressyon cizgisinin altinda ve ustunde kalan noktalar icin residual’lari toplarsan 0 cikar. O yuzden LSE’yi kullaniyoruz. Yani karelerini alarak hatalarin hesaba katilmasini sagliyoruz.**

**Lecture:**



**Potential\_spend olarak yeni bir df olusturduk.**

**Bunun icin de plot cizdirdik. Total\_spend scatterplot ile kiyasliyoruz.**



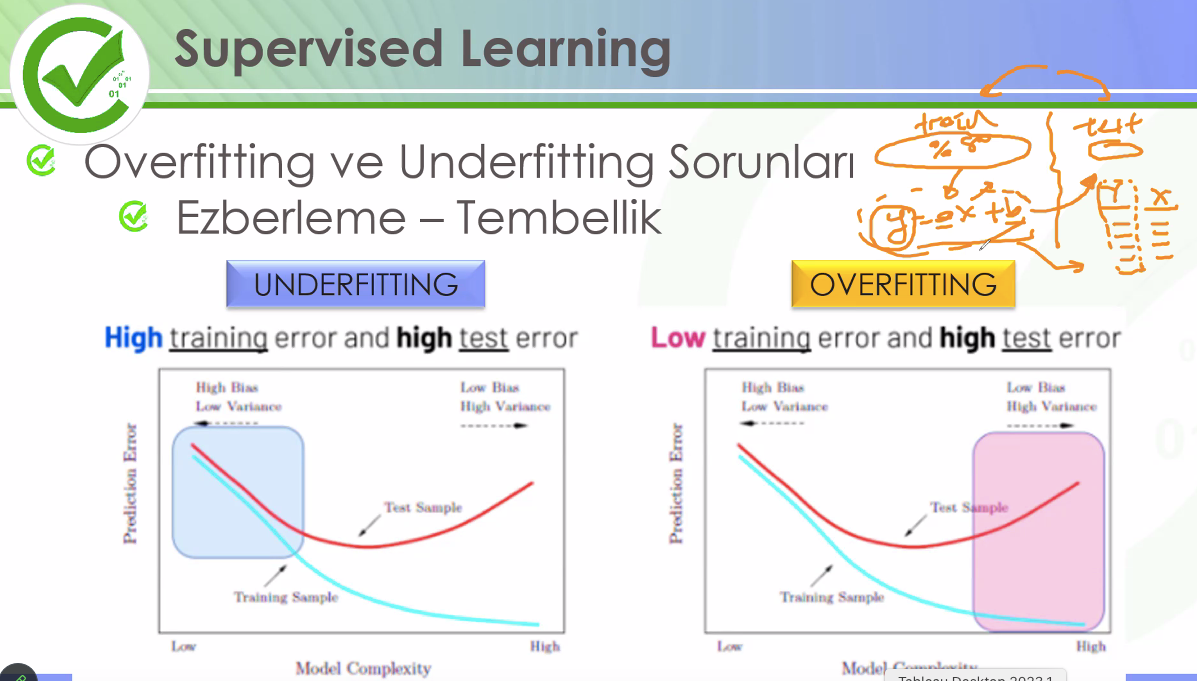
**Bu tahmini daha iyi nasil iyilestirebilirim?**

-degree’yi artirabiliriz.

-model complexity’i artirabiliriz.

-ancak cok artirirsak overfitting riski olusur.

**ML\_Theory devam**

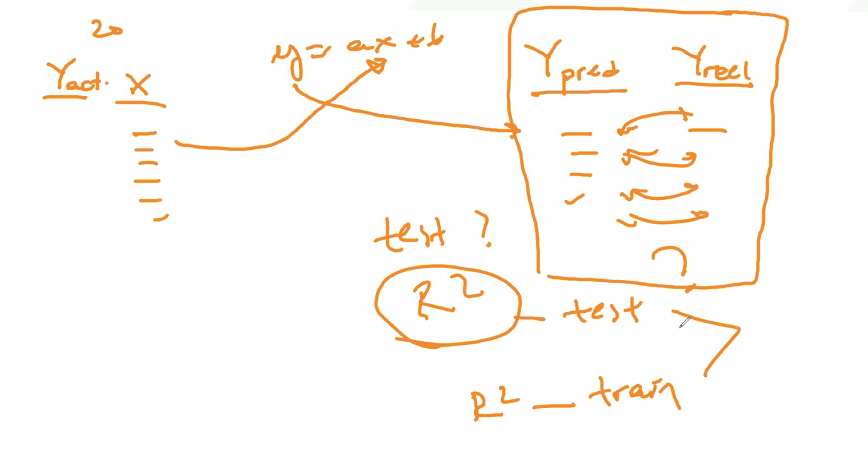


Datasetimin %80-%20 diye boluyorum, birini train dataseti, digerini test dataseti olarak ayiriyorum.

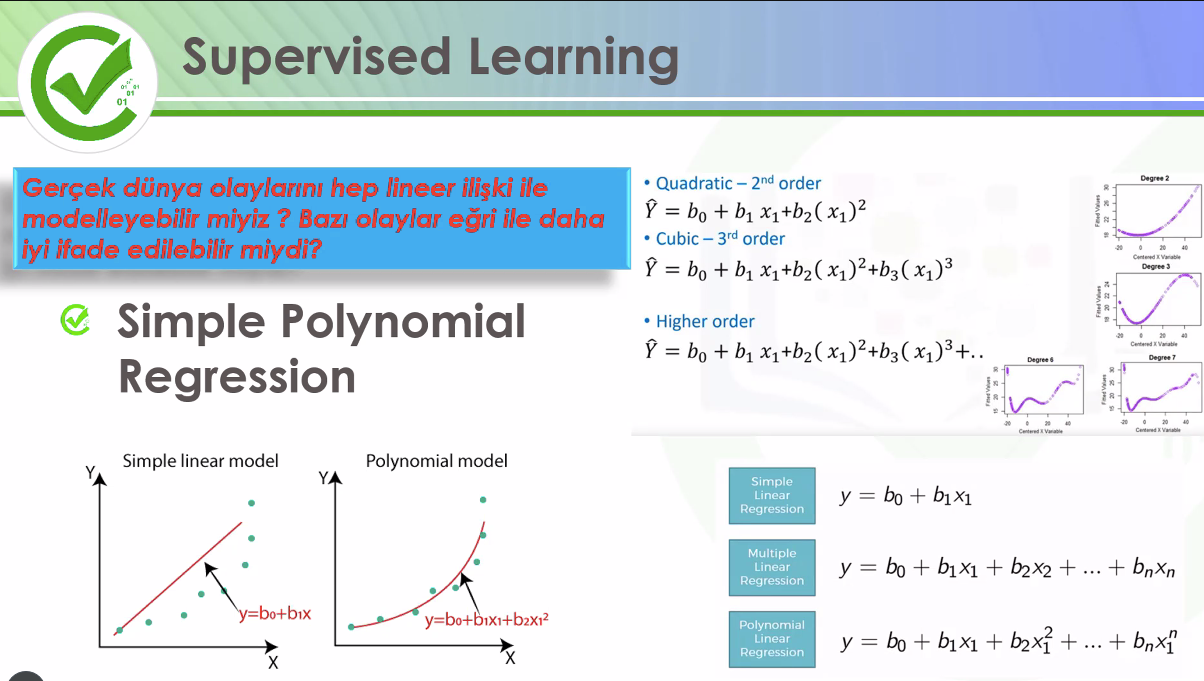
Elimde y (Yact) ve x degerleri var gercek data olarak.

Bir de y=ax+b formulune x datalarini yazarak elde ettigim predicted y var (Ypred).

Biz Ypred ile Yact’u kiyaslayarak aradaki farklara bakiyoruz.



**Simple Polynomial Regression**



Tahmin egrisi cizersem noktalara dogrudan daha yakin olabilir. Bu sayede bilgi kaybini azaltabilir, maliyeti dusurmus, hatayi minimize etmis olabiliriz. Polynom regresyonu ile bunu yapabiliriz.

Y=ax2+bx+c

Yani x’in hem 1. hem 2. kuvveti var icinde. Yani 2. dereceye cikardik.

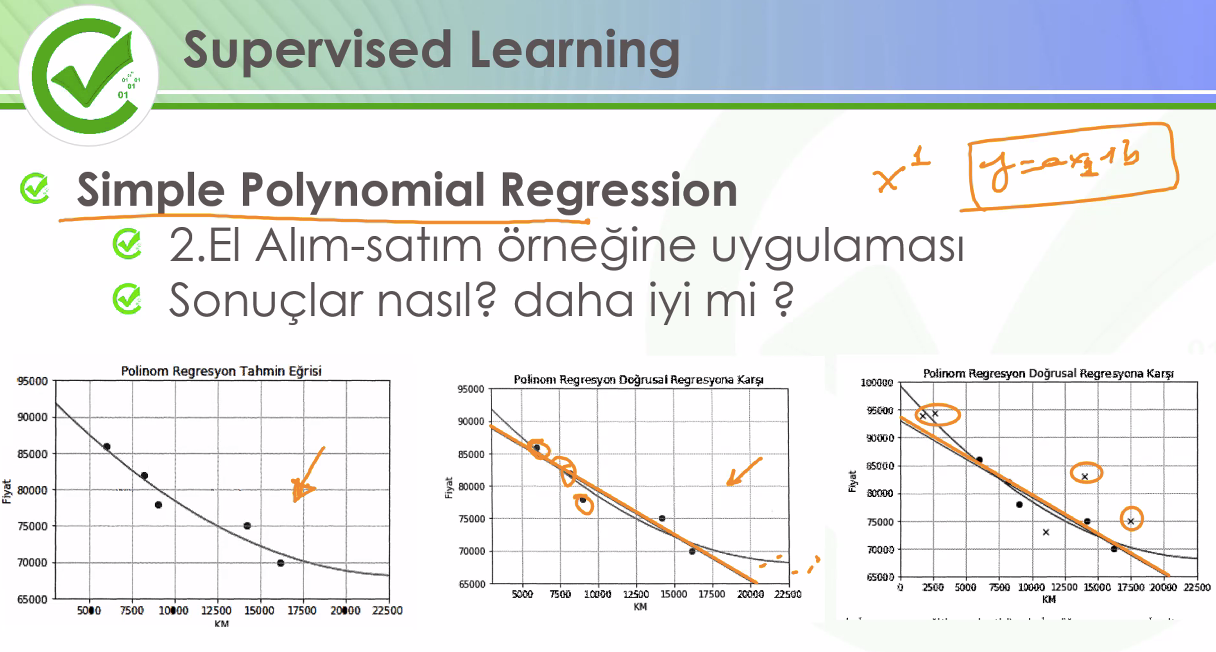
Bu sayede noktalara daha yakin ve noktalari daha cok temsil edecek bir egri elde etmeme yaradi dereceyi artirmak. Kubik olsaydi 3. dereceye cikmis olurdu.

Derece arttikca egri s’ye benziyor ve egriligi daha cok artiyor.

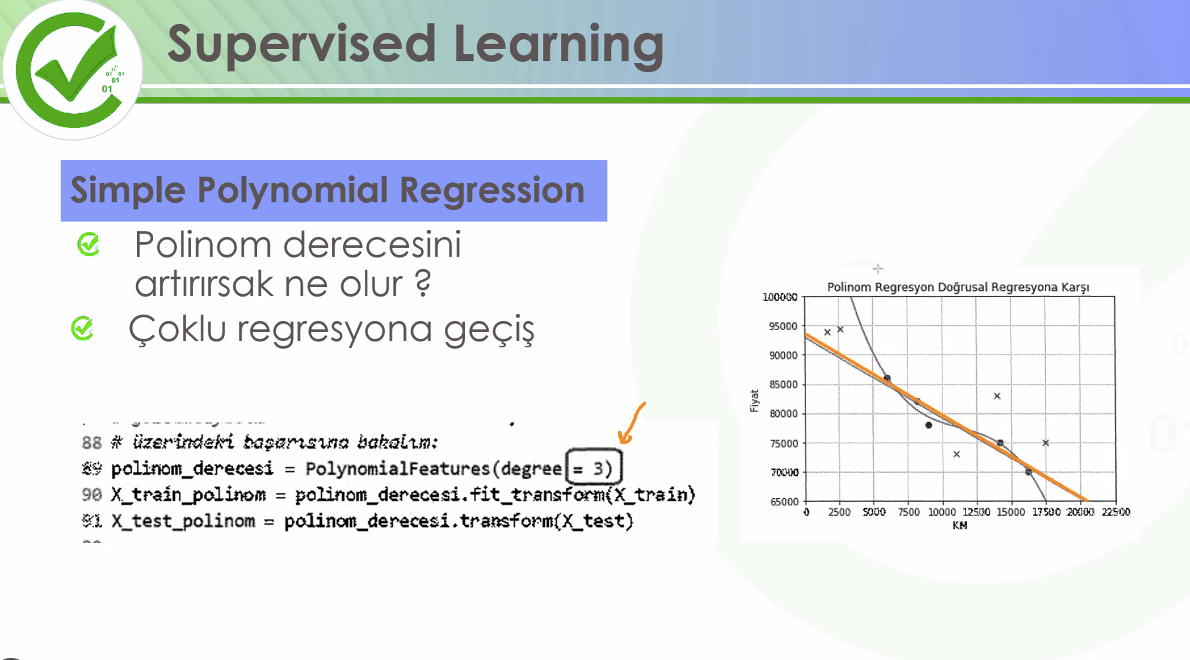
polynom ifadesi gectigi zaman x2, x3 demektir.

Multiple diyince birden fazla x var demektir.

Degree artinca R2 artiyorsa, datanin daha cogunu aciklayabilmis oluyoruz degree artinca deriz.



Train setindeki algoritma test setinde de ayni basariyi gosteriyor mu?

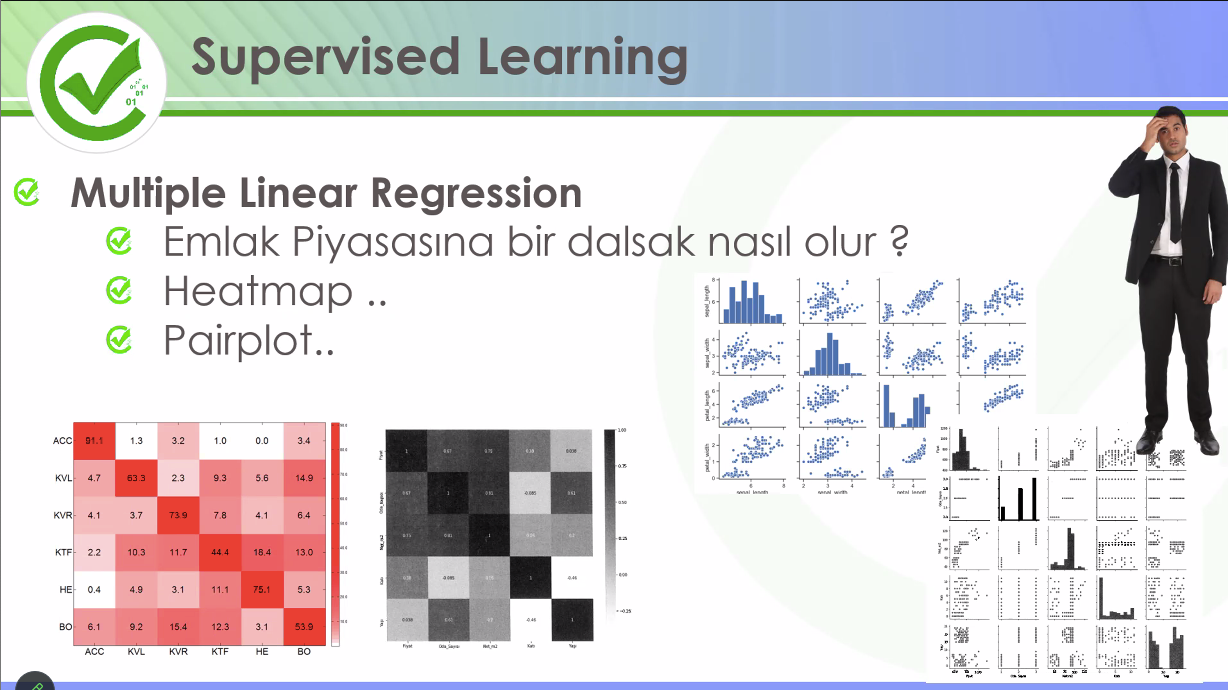


Olayi etkileyen asil feature’lari belirlememiz gerek.

Feature sayisi yeterli degilse, feature ile modeli beslemeliyiz.

Target’i etkileyebilecek feature’lari belirleyip onlari hesaba katmamiz gerek. Bu da multiple linear regression ile olacak.

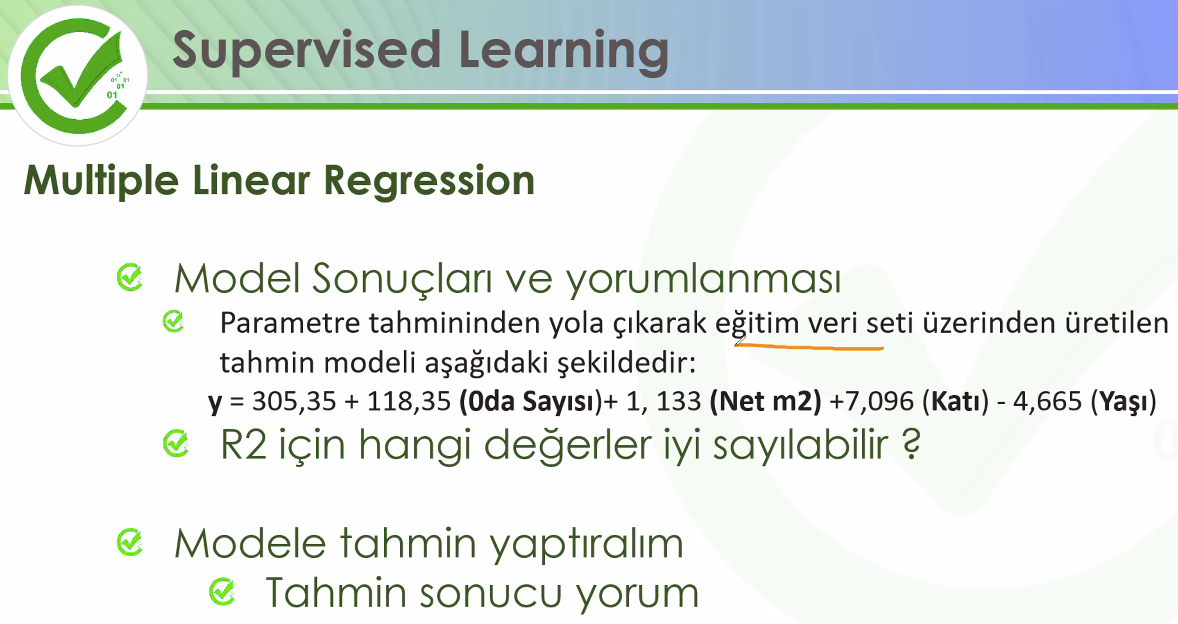
**Multiple Linear Regression**



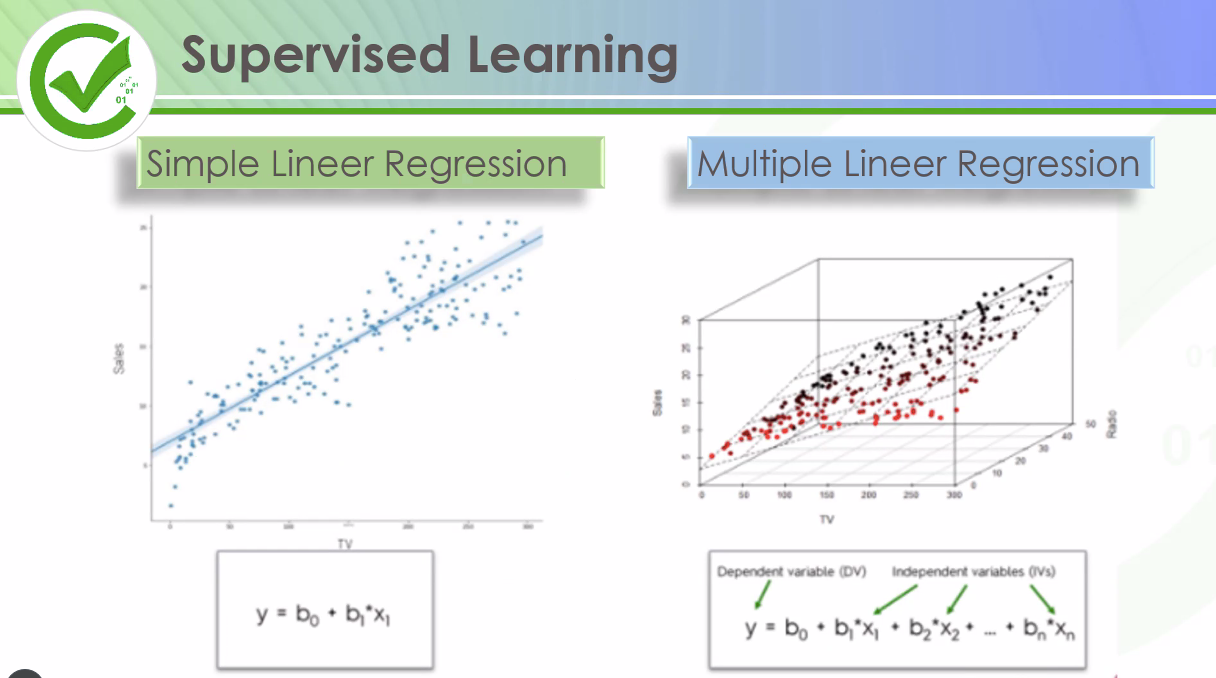
Evin fiyati en cok net m2 ile korele gorunuyor.

Elinizdeki datasetini %80-%20 olarak train dataseti ve test dataseti olarak ayirirsiniz.

Feature’lar arasinda bir celcius bir de fahreneit varsa, birini secerek devam etmem gerek. Cunku ikisini de dahil edersem, etkilerini bolerek modele sokmus oluyorum. **Multicolinearity** deniyor buna. Bagimsiz degiskenler arasinda ciddi korelasyon olmasi sorunu. O zaman modelde overfitting sorunu olur. Ezbere gider model.

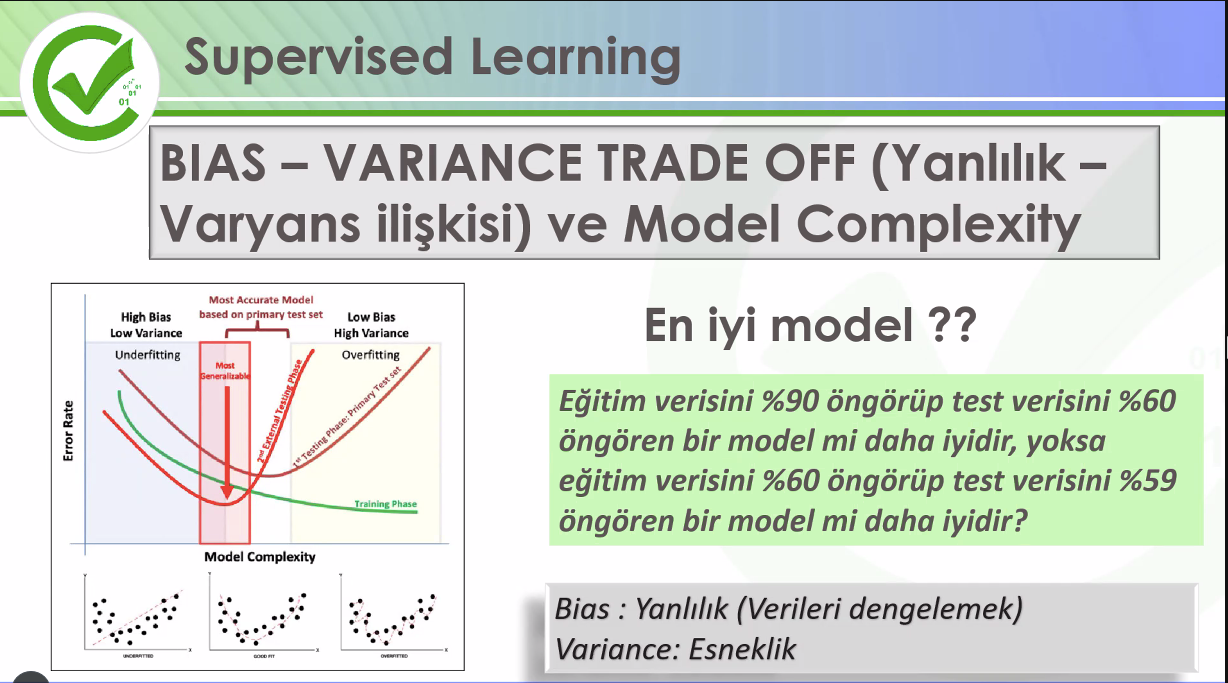


Amac feature’lardan yola cikarak target’i yuksek oranda tahmin etmeye calismak.



Hepsinin derecesi 1. Sadece birinde tek x var, digerinde birden fazla x

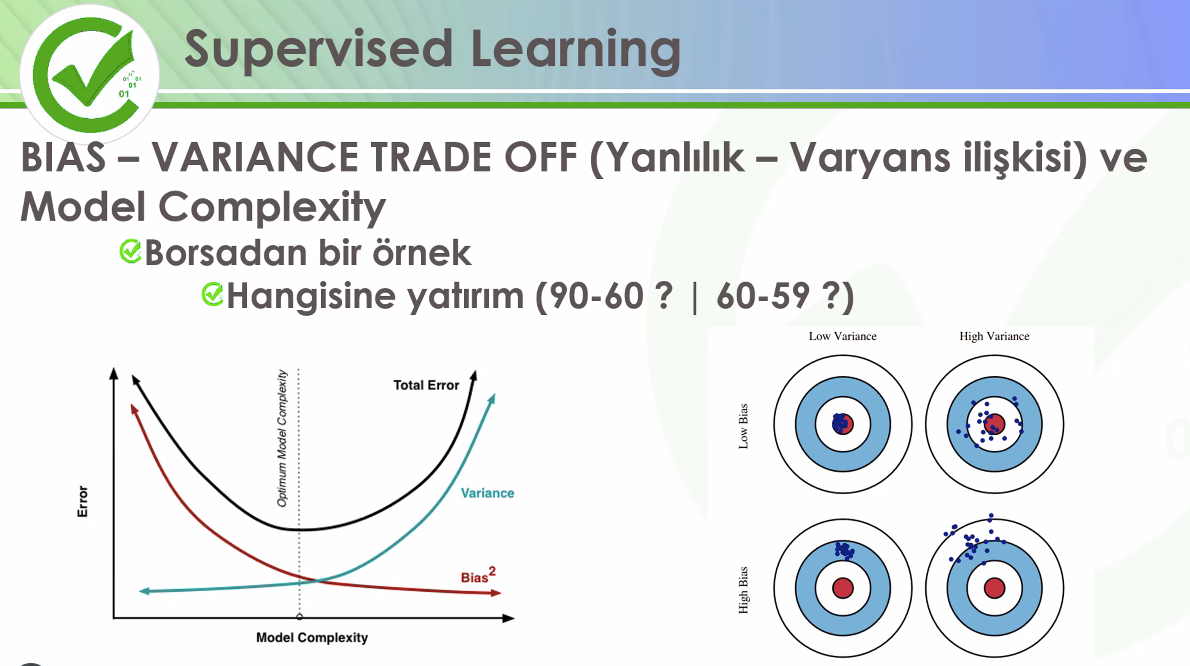
**Bias-Variance Trade Off**



En yuksek R2’ye sahip olan model en iyi model midir?

Bias-->yanlilik

Variance--> yayilma olcusu



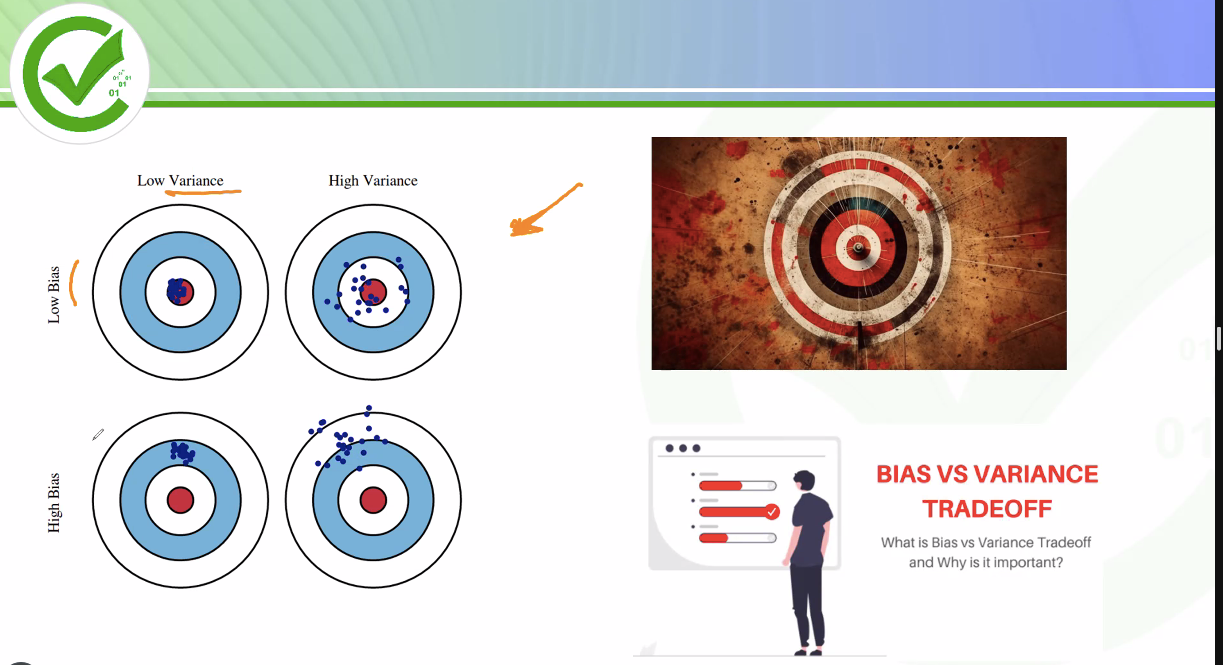
Ypred ile Yactual arasinda hata ne kadar az olursa, maliyet o kadar az olur.

Modeller herseye ragmen cogunlukla hata uretirler. Hatanin bilesenlerinden biri olan bias ve variance arasindaki bu alisveristen kaynaklaniyor.

**Modelin kompleskligini artirarak:**

**Yeni feature’lar ekleyerek + degreesini artirarak daha iyi sonuclar elde edebilirim.**

Az feature varken variance az oluyor. Ama feature arttikca variance artmis oluyor. Modelin kompleksligini arttirdikca hatayi dusuruyorsun belli yere kadar ama belli bir yerden sonra variance yeniden artiyor. O yuzden optimumunda kalmak gerek. Modelin kompleksligini cok artirirsan da overfitting’e goturuyorsun.



Dusuk bias+dusuk variance ideali.

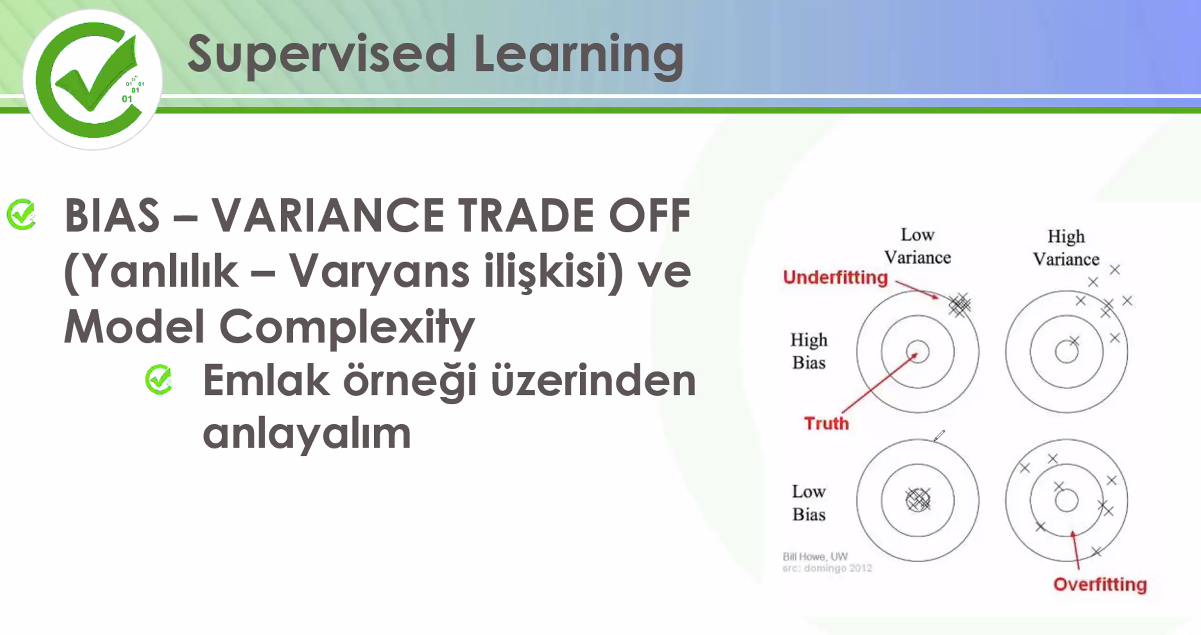
**Bias**--> Yanlis bir sekilde varsaymasindan kaynaklanir. Yuksek bias demek mesela atislarin hedefin epey ustunde kalmasi ya da epey altinda kalmasi.

**Variance**--> genellemeyi yapamamasi anlamina gelir yuksek variance. Yuksek variance--> dagilimi fazla

**Noise**--> verilerde gurultu,

|  |
| --- |
| **Bias-Variance Trade\_off (Mert Urper)**  **Yüksek Bias Düşük Varyans:** Modeller tutarlıdır, ancak ortalama hata oranı yüksektir.  **Yüksek Bias Yüksek Varyans** : Modeller hem hatalı hem de tutarsızdır .  **Düşük Bias Düşük Varyans**: Modeller ortalama olarak doğru ve tutarlıdır. Modellerimizde bu sonucu elde etmek için çabalamaktayız.  **Düşük Bias Yüksek Varyans:** Modeller bir dereceye kadar doğrudur ancak ortalamada tutarsızdır. Veri setinde ufak bir değişiklik yapıldığında büyük hata oranına neden olmaktadır. |

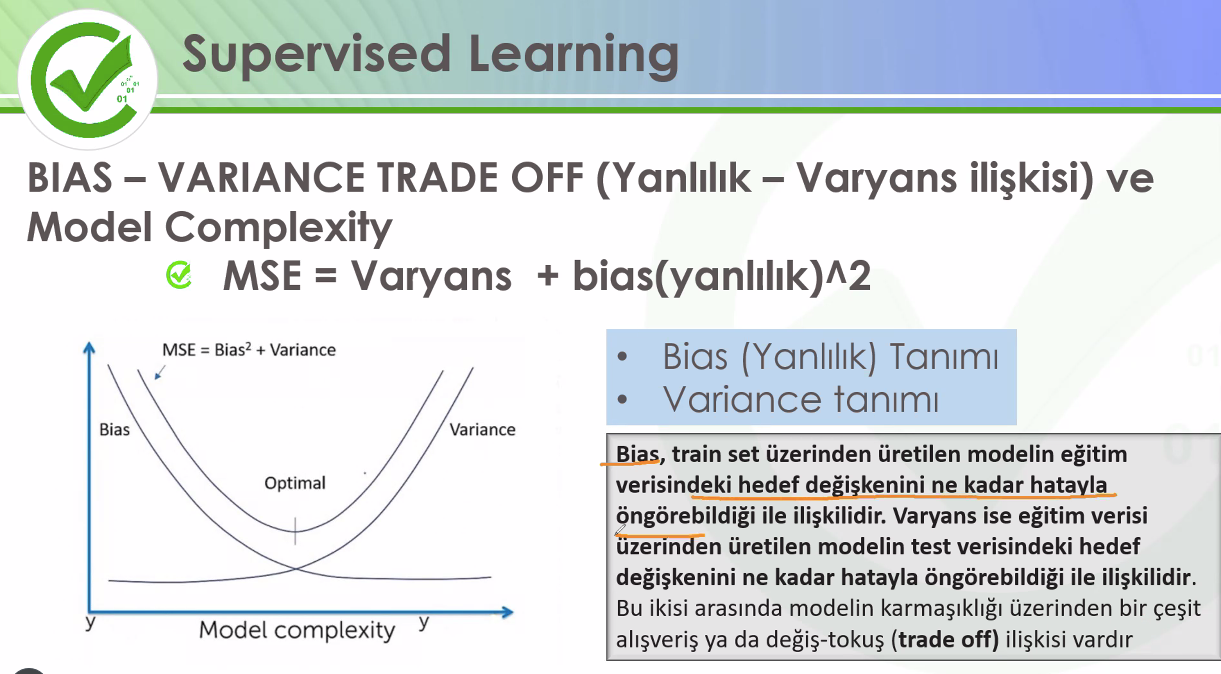
**modelin karmaşıklığı arttıkça variance artar, bias düşer--> bias-variance trade-off**



Olaya katilan her bir feature, olayi aciklayan yeni bir degisken demektir.

Train’de %90, testte %60 dogrulugun varsa, o zaman overfitting var demektir.

Train’de %60’larda dogrulugum varsa--> high bias var demektir. Zaten model trainden yeterince ogrenememis, underfitting durumu var. O zaman zaten teste bakmama gerek de yok.



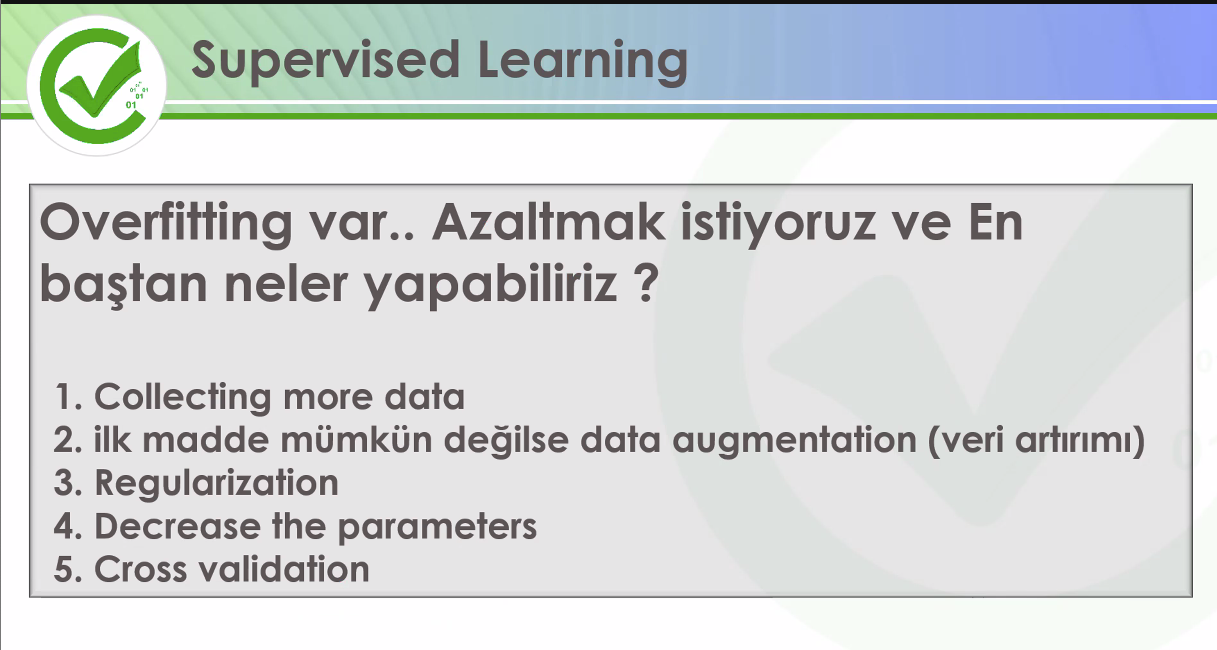
Karmasiklik arttikca, train dataseti ile test dataseti arasindaki fark acilmaya baslar

Bias asagi dogru azalmis oluyor ama variance yukari dogru artmis oluyor. This is trade-off.

Yuksek variance olursa, yeni durumlari aciklamak icin yetersiz kalir.

Optimum komplekslik noktasinda bir model olusturman gerek.

**Overfittingi nasil cozebilirim?**



ML’de pek **data augmentation** yapamiyoruz aslinda ama DL’de yapiyoruz.

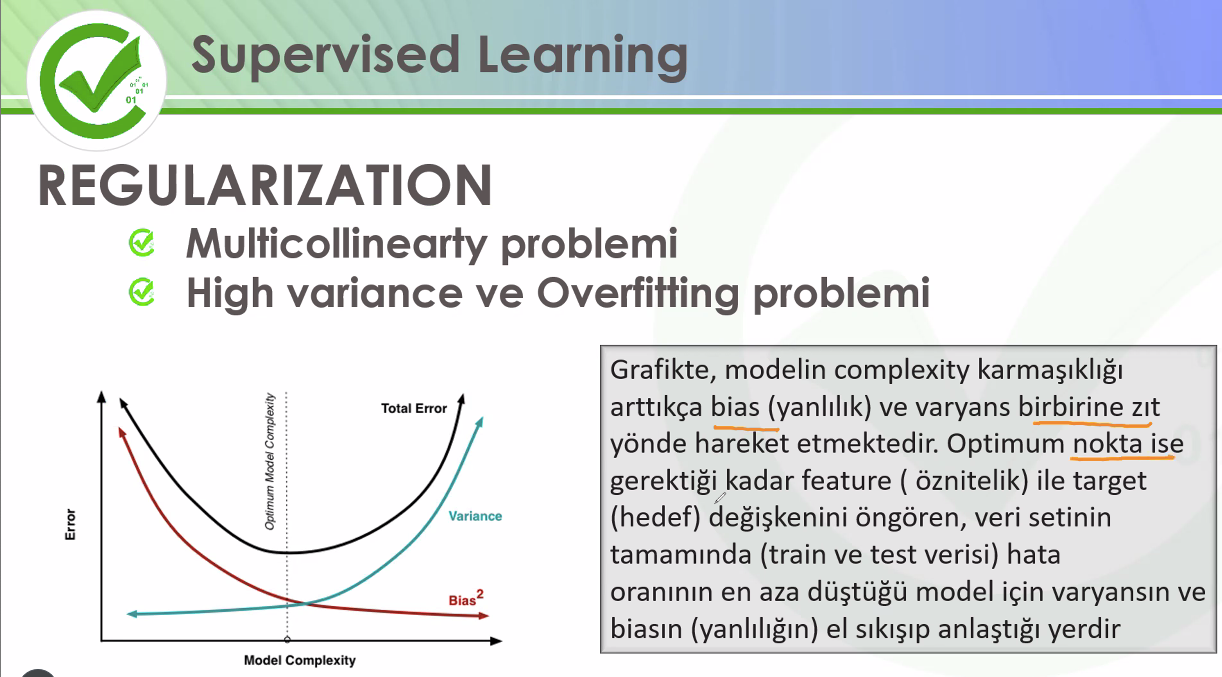
100 resmim var mesela, ters cevirdim 200 resim oldu. Zoom yaptim 300 oldu gibi. Data collect etmeden, ayni verisetinden data uretiyorum.

**Decrease parameters-**->Bir parametre var derinligni ayarliyorsun

Bir parametre var mesela dallanmasini ayarliyorsun gibi

O parametrelerle ayar yapiyoruz.

Mesela overfitting sorununa sebep olabilecek olanin parametresini azaltiyorum gibi.

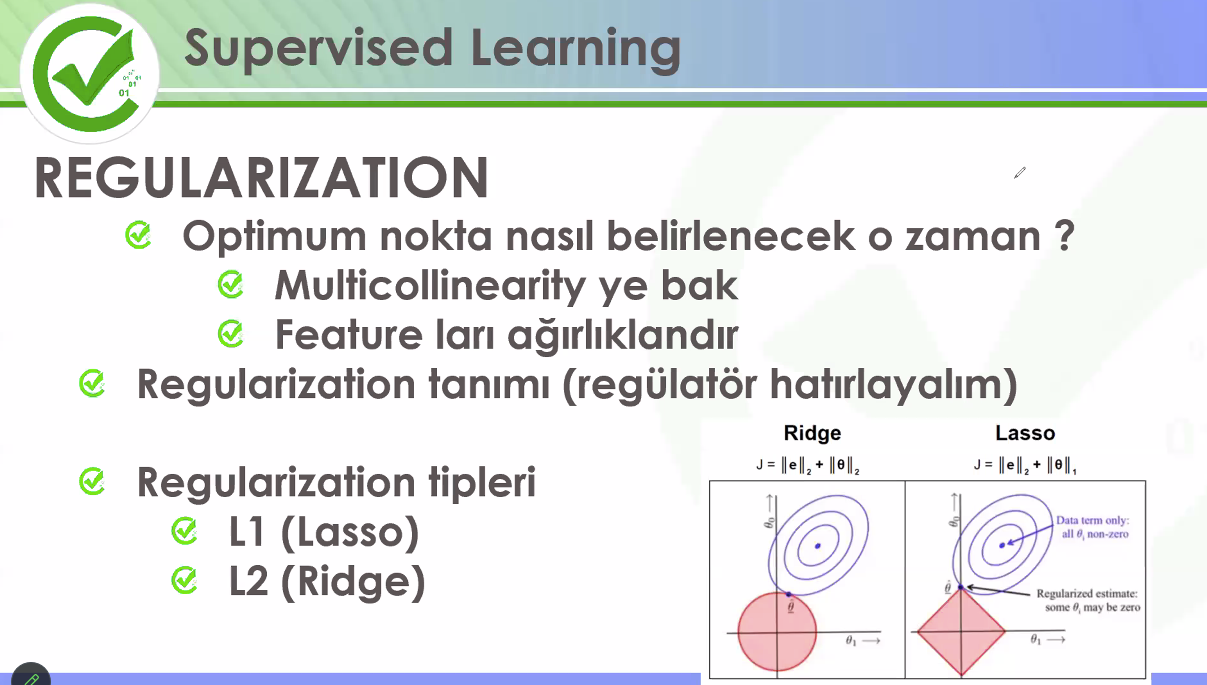


Optimum noktayi elde etmek icin regularization’da mucadele edecegimiz sorunlardan bir tanesi multicolinearity. Coklu dogrusal baglanti sorunu deniyor buna. Aralarinda yuksek korelasyon olan feature’larin modele eklenmesi mesela bu sorunu dogurur.

Cok feature eklersen, overfitting’e gider. Cunku gereksiz olan feature’lar olabilir.

Overfittingin 3 sebebi vardi:

1. Multicolinearity
2. Fazla feature
3. Bir de parametrelerin yanlis belirlenmesi.



1. **Feature’larin agirliklandirilmasi**

**L1-Lasso**--> Onemli olanlarin daha yuksek bir katsayi ile agirlandirilarak, onemsiz olanlarin agirliklarini azaltir. Katsayisi dusuk olanlari modelden cikariyor Lasso--> Lasso kelle alir.

Feature selection bu sekilde regularization ile yapiliyor.

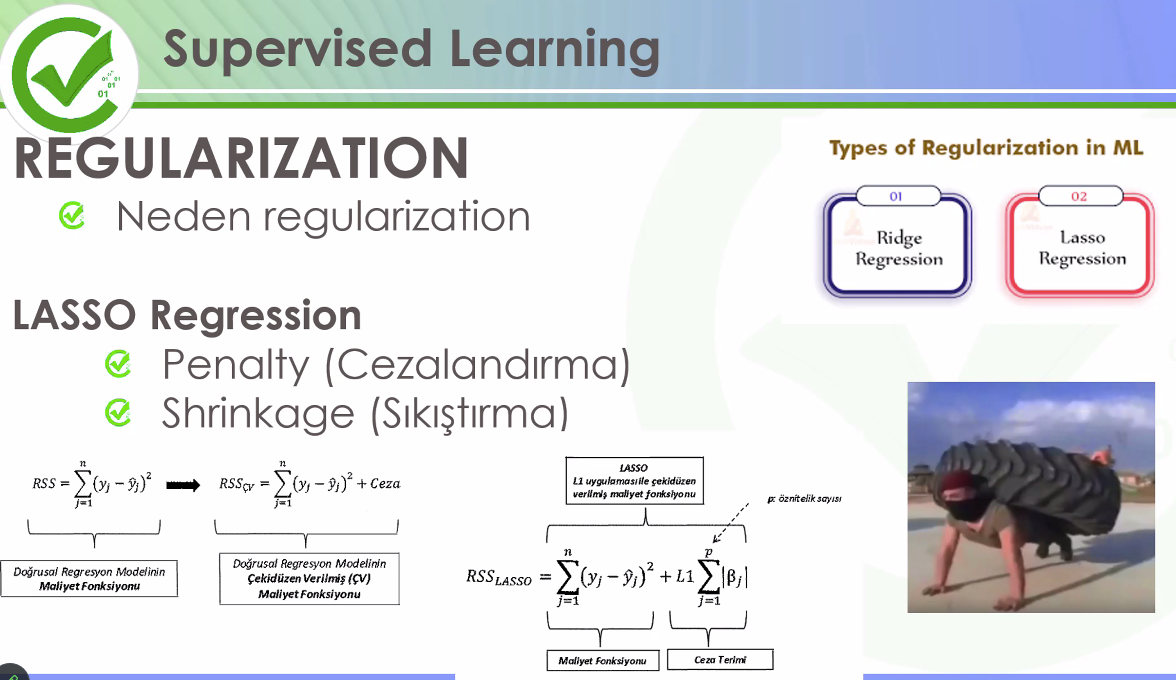
**L2-Ridge**’de de agirliklandirmayi yapiyor ama modelden cikarmiyor. Ama modelde etkisini azaltiyor. -->Butun feature’lar modelde kalir.

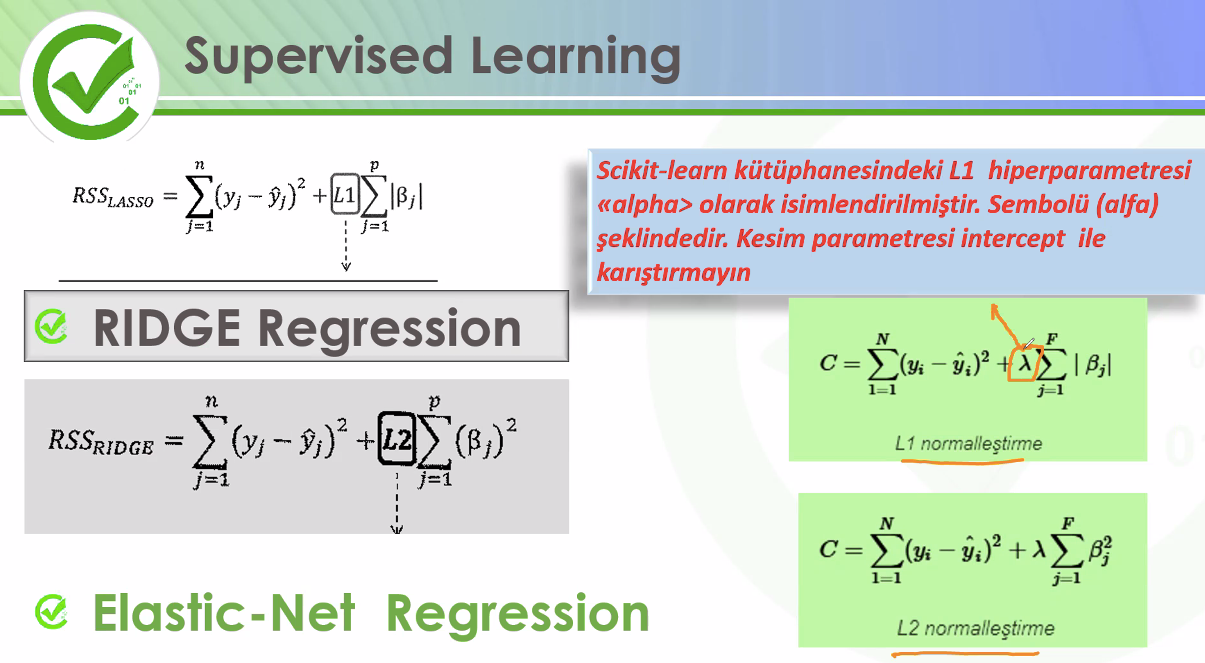
Hata orani yuksekse, onlara ceza uyguluyor. Ilgisiz degiskenleri modelden cikartmiyor ama onlara dusuk katsayi veriyor. Ilgisiz degiskenlerin katsayilarini 0’a yaklastirir.

Ridge ve Lasso’nun birbirlerine karsi ustunlukleri yok.

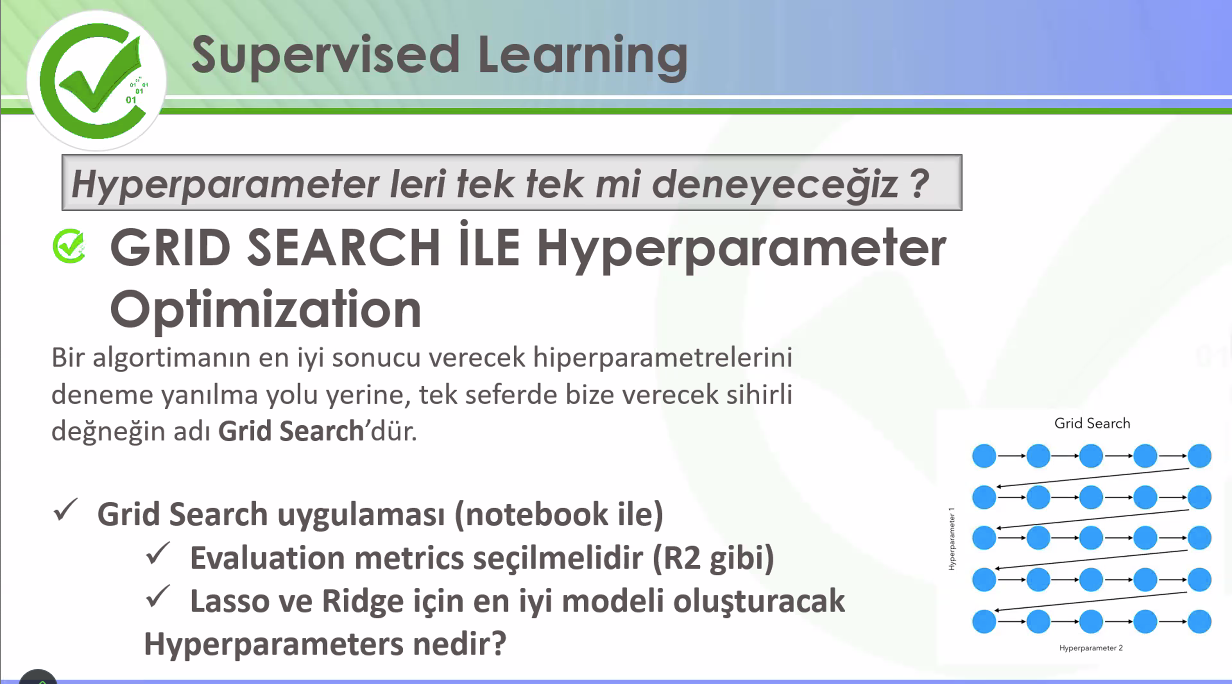
**Elastic net** ise ridge tarzi cezalandirma, lasso tarzi eliminasyon yapiyor. En modern ve etkili yontemdir.

**Yani regularization--> Train-test arasindaki variance’i azaltmaya calisiyor.**





**Elastic net** ise ridge tarzi cezalandirma, lasso tarzi eliminasyon yapiyor. En modern ve etkili yontemdir.



Optimumu grid search ile buluyoruz.

**Grid search-**-> modelin icindeki parametrelerin en iyisini bulmaya calisiyoruz. Bunun bir calisma mantigi var. A parametresine bir liste atiyorum mesela, B ve C icin de mesela farkli listeler atiyorum, kullanmak istedigim parametrelerin araligini belirleyerek

Herbir parametre ayari icin modeli calistiriyor ve tum parametreleri deneyerek bana en iyi olani veriyor. A parametresi icin mesela 300 en iyi degerdir diyor. Ben bu en iyileri alip modeli bu degerlerle calistiriyorum. Hyper parametre tuning yaptim der. HPT

Best parameters yapip calistirirsam bunlari bana vermis oluyor.

Cross validation ile modelin dogrulugunu kanitlamaya calisiyordum, train dataseti icinden

