biomechatronics
LABORATORY

Mekatronik Mühendisliği Uygulamalarında Yapay Zekâ

Makine Öğrenmesi

Erhan AKDOĞAN, Ph.D.

Bu ders notunun hazırlanmasında Dr. U.Orhan ve Banu Diri ’nin ders notlarından yararlanılmıştır.

Makine öğrenmesi nedir?

- Makine Öğrenmesi, verilen bir problemi probleme ait ortamdan edinilen veriye göre modelleyen bilgisayar algoritmalarının genel adıdır.

Makine Öğrenmesi nedir?

- Çok büyük miktarlardaki verinin elle işlenmesi ve analizinin yapılması mümkün değildir.
- Amaç geçmişteki verileri kullanarak gelecek için tahminlerde bulunmaktır.
- Bu problemleri çözmek için Makine Öğrenmesi (machine learning) yöntemleri geliştirilmiştir.
- Makine öğrenmesi yöntemleri, geçmişteki veriyi kullanarak yeni veri için en uygun modeli bulmaya çalışır.
- Verinin incelenip, içerisinden ise yarayan bilginin çıkarılmasına da Veri Madenciliği (data mining) adı verilir.

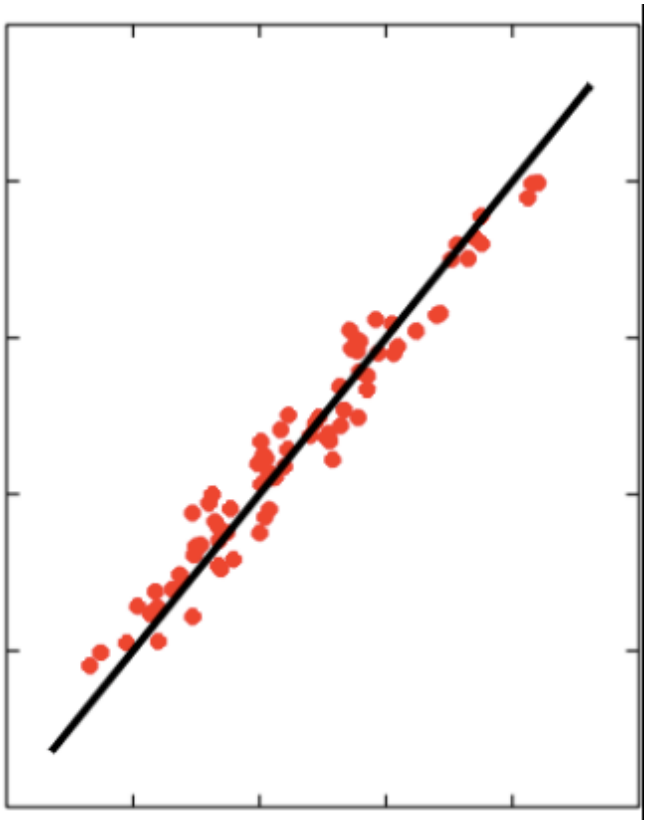
Makine Öğrenmesi Terimleri:

- Tahmin (prediction): Veriden öğrenen modellerde sistem çıkısının nicel olması durumunda kullanılan yöntemlerin ürettiği değerlerdir.

- Sınıflandırma (classification): Giris verisine ait çıkısların nitel olduğu durumlarda kullanılan yöntemlerin her veri örneğinin hangi sınıfa ait olduğunu belirlemesidir.

- İstatistikte rasgele bir değişkenin bilinmeyen bir değerinin belirlenmesi için tahmin (prediction), bilinmeyen bir sabitin belirlenmesi içinse kestirim (estimation) kavramından bahsedilir. Çok yakın anlamları dolayısıyla bu iki terim literatürde çoğunlukla karıştırılarak aynı şeyi ifade etmede kullanılır.

Örneğin sağda veri noktalar için bir doğrusal eğri fonksiyonu tespit edilmiştir. Herhangi bir x değerine karşılık gelen y değerini belirlemekte “tahmin”, tahmini yapmamıza yardımcı olan eğrinin (eğim ve sapma değeri gibi) parametrelerinin belirlenmesinde ise “kestirim” terimleri tercih edilmelidir.



Tahmin örneği:

Bir probleme ilişkin olarak sağdaki veri kümesi veriliyor. Amaç bir x girdisine karşı bir y değerini üretmektir. Buna göre veri üzerinde doğru ilişkiyi bulan öyle bir $f : X \rightarrow Y$ fonksiyonu tanımlanmalıdır ki en küçük hatayı üretsin.

Bu basit problemde $y=3*x$ çözümü kolayca görünmektedir.

X	Y
3	9
5	15
8	24
10	30
19	57
24	72
27	81
31	93
38	114
43	129

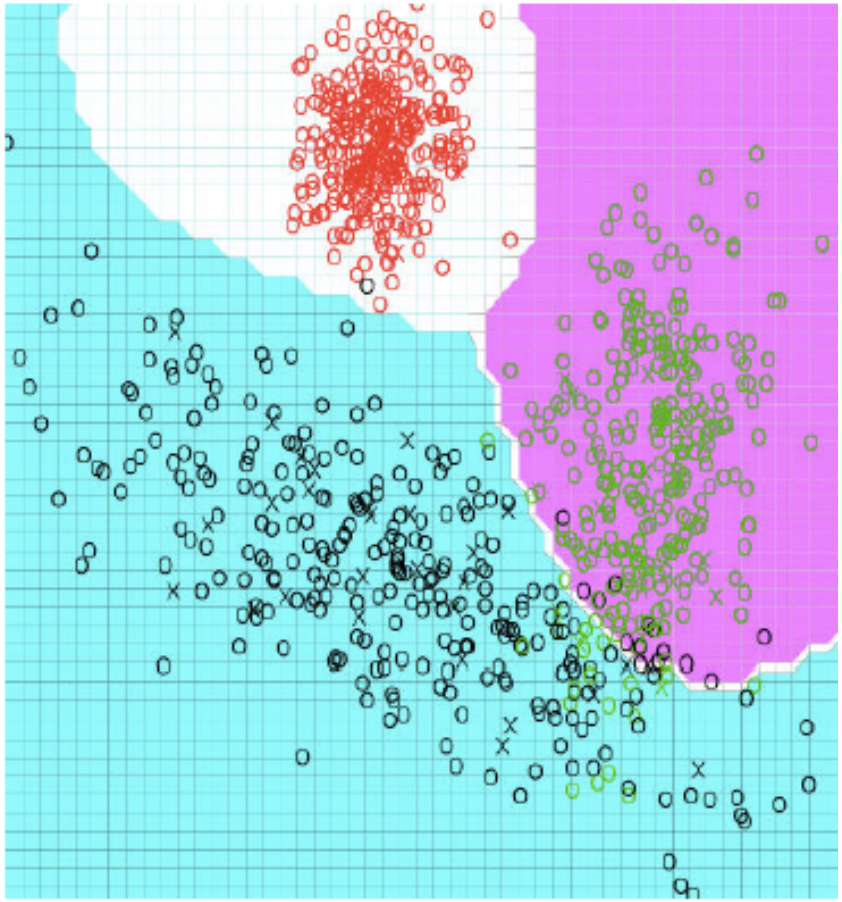
Tahmin örneği:

Bulunan y = 3 * x çözümüne iliskin olarak eğer x=50 değeri için y değeri istenirse kestirilen denklem yardımı ile y=150 tahmin değerini kolayca hesaplamak mümkün olur.

X	Y
3	9
5	15
8	24
10	30
19	57
24	72
27	81
31	93
38	114
43	129

Sınıflandırma (Classification)

- Amaç probleme ait tüm uzayın belirli sayıda sınıfa bölünmesidir. Sağdaki resimde her renk bir sınıfı göstermektedir.
- Sınıflandırma teknikleri sayesinde hiçbir veri örneğinin olmadığı bölgeler de renklendirilebilir.



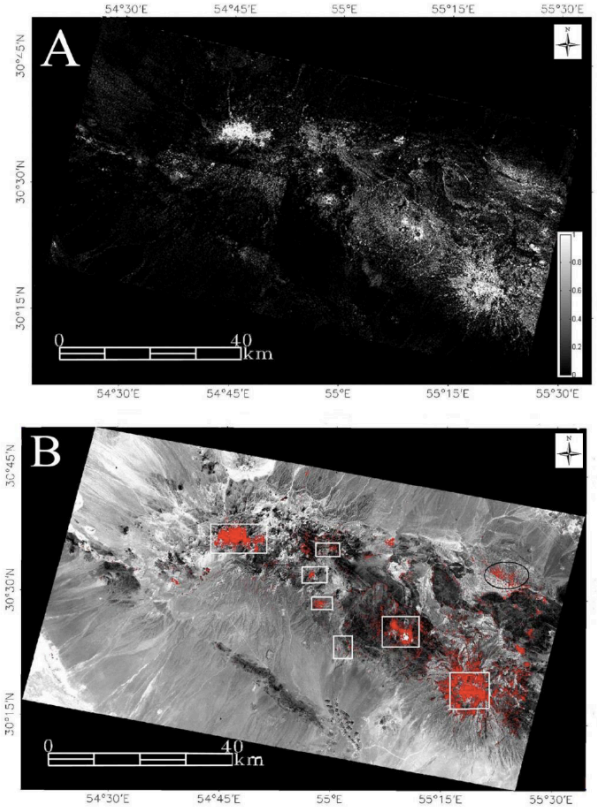
Sınıflandırma örneği:

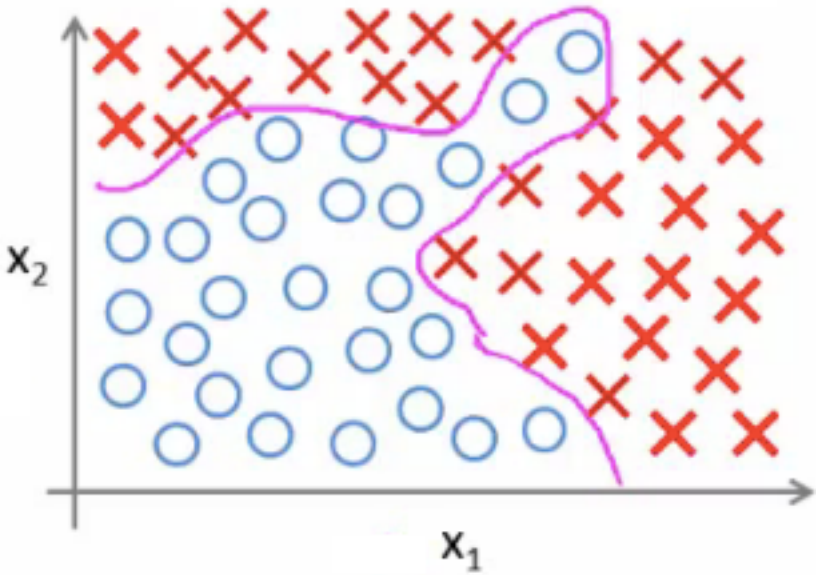
Sağdaki veri kümesinde amaç X girdisine karşı Y etiket (label) değerini üretmektir. Öyle bir $f: X \rightarrow Y$ sınıflandırma fonksiyonu tanımlanmalıdır ki en az hata ile sınıflandırma yapılabilsin.

$$Y = \begin{cases} 0, & X < 20 \\ 1, & X \geq 20 \end{cases}$$

X	Y
2	0
5	0
9	0
13	0
19	0
20	1
27	1
33	1
39	1
47	1

Sınıflandırma örnekleri

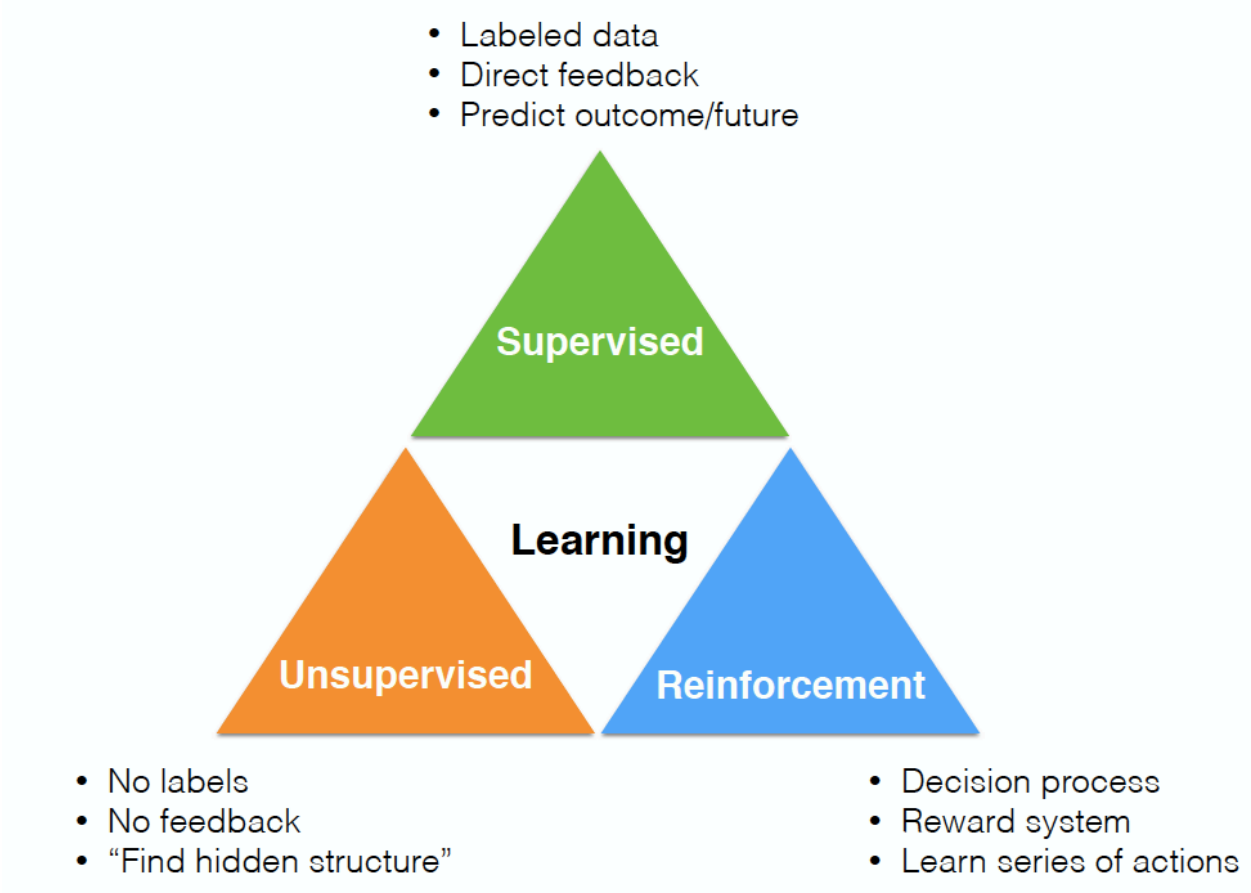




Makine öğrenmesi terimleri:

- Makine Öğrenmesi yöntemleri verinin yapısına göre ikiye ayrılır:
  - **Danışmanlı (supervised) Öğrenme:** Veri, etkiye tepki prensibiyle çalışan sistemlerden alınır ve giris-çıkış düzeninde organize edilir.
  - **Danışmansız (unsupervised) Öğrenme:** Sınıf bilgisi olmayan veya verilmeyen veri içerisindeki grupları keşfetmeyi hedefler.

Öğrenme Yöntemleri:



Reference: Dr. Sebastian Raschka, An Introduction to Supervised Machine Learning and Pattern Classification: The Big Picture

Danışmanlı ve Danışmansız Öğrenme:

Bir örnek için aşağıdaki linki inceleyiniz:
http://dataaspirant.com/2014/09/19/supervised-and-unsupervised-learning/

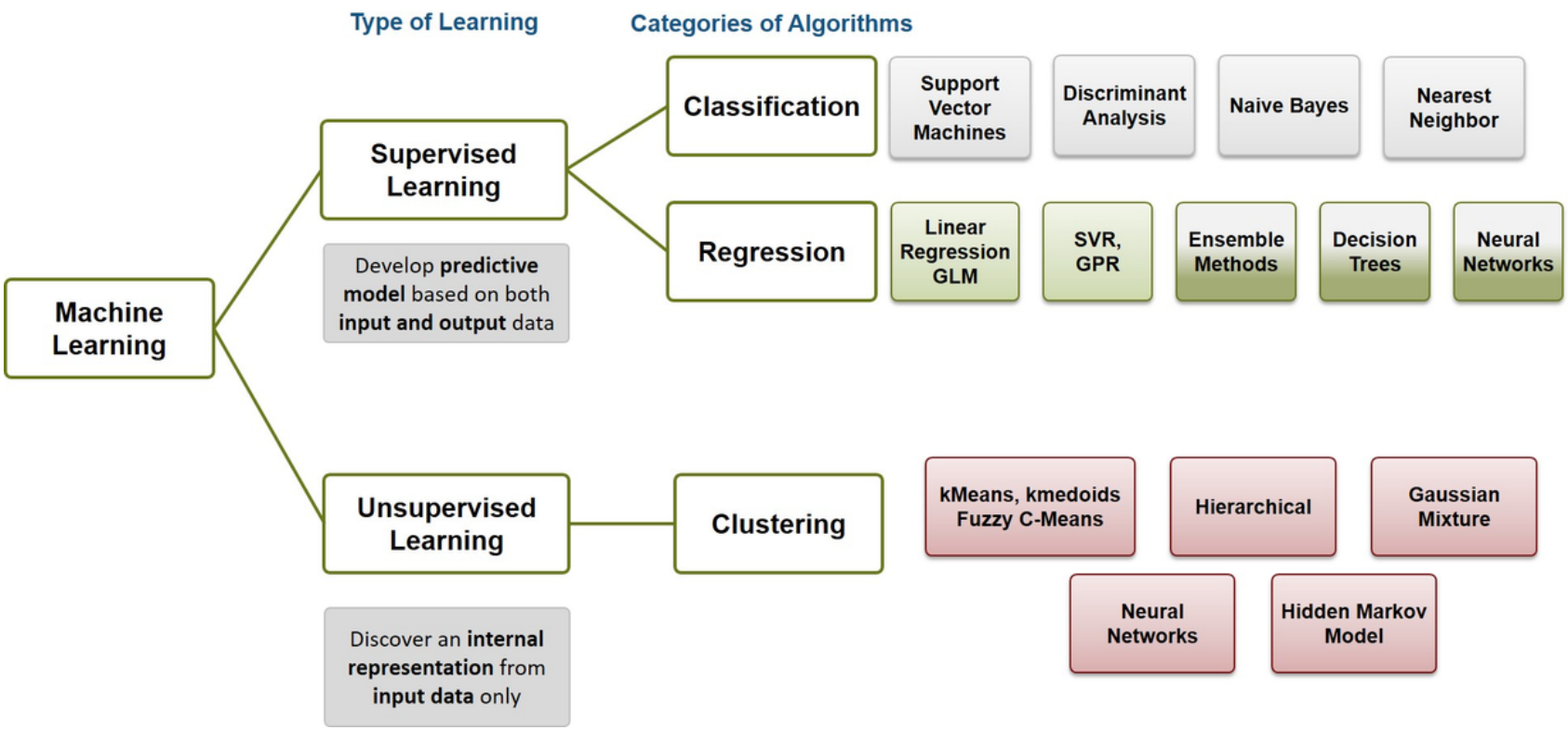
Danışmanlı öğrenme algoritmaları:

- Tüm sınıflandırma ve regresyon metodları danışmanlı öğrenme altında değerlendirilebilir:
  - Logistic Regression
  - Decision trees
  - Support vector machine (SVM)
  - k-Nearest Neighbors
  - Naive Bayes
  - Random forest
  - Linear regression
  - polynomial regression
  - SVM for regression

Danışmansız öğrenme algoritmaları:

- Kümeleme algoritmaları danışmansız öğrenme altında değerlendirilebilir:
  - K – means clustering
  - Hierarchical clustering
  - Hidden Markov models

İlişkili Algoritmalar:

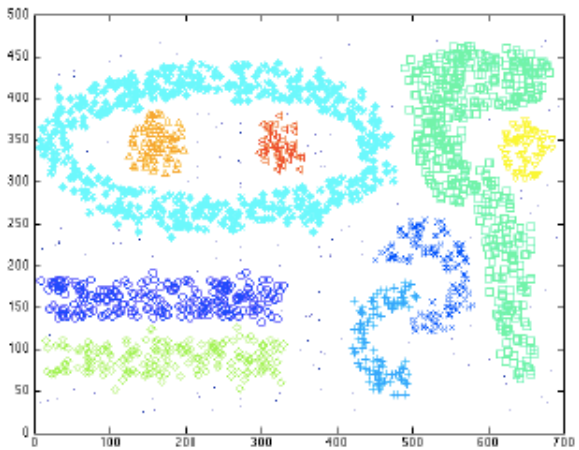


Reference: http://embedded-computing.com/articles/analytics-driven-embedded-systems-part-2-developing-analytics-and-prescriptive-controls/

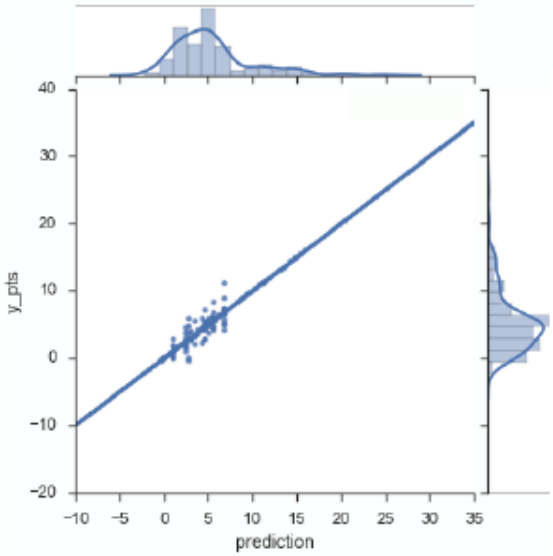
Karşılaştırma:

Unsupervised Learning

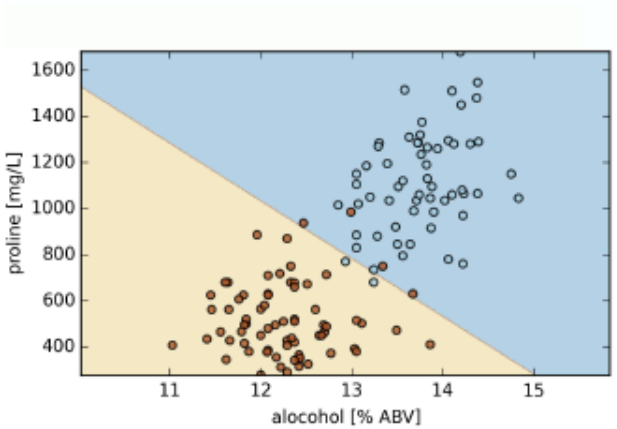
Supervised Learning



Clustering:
[DBSCAN on a toy dataset]



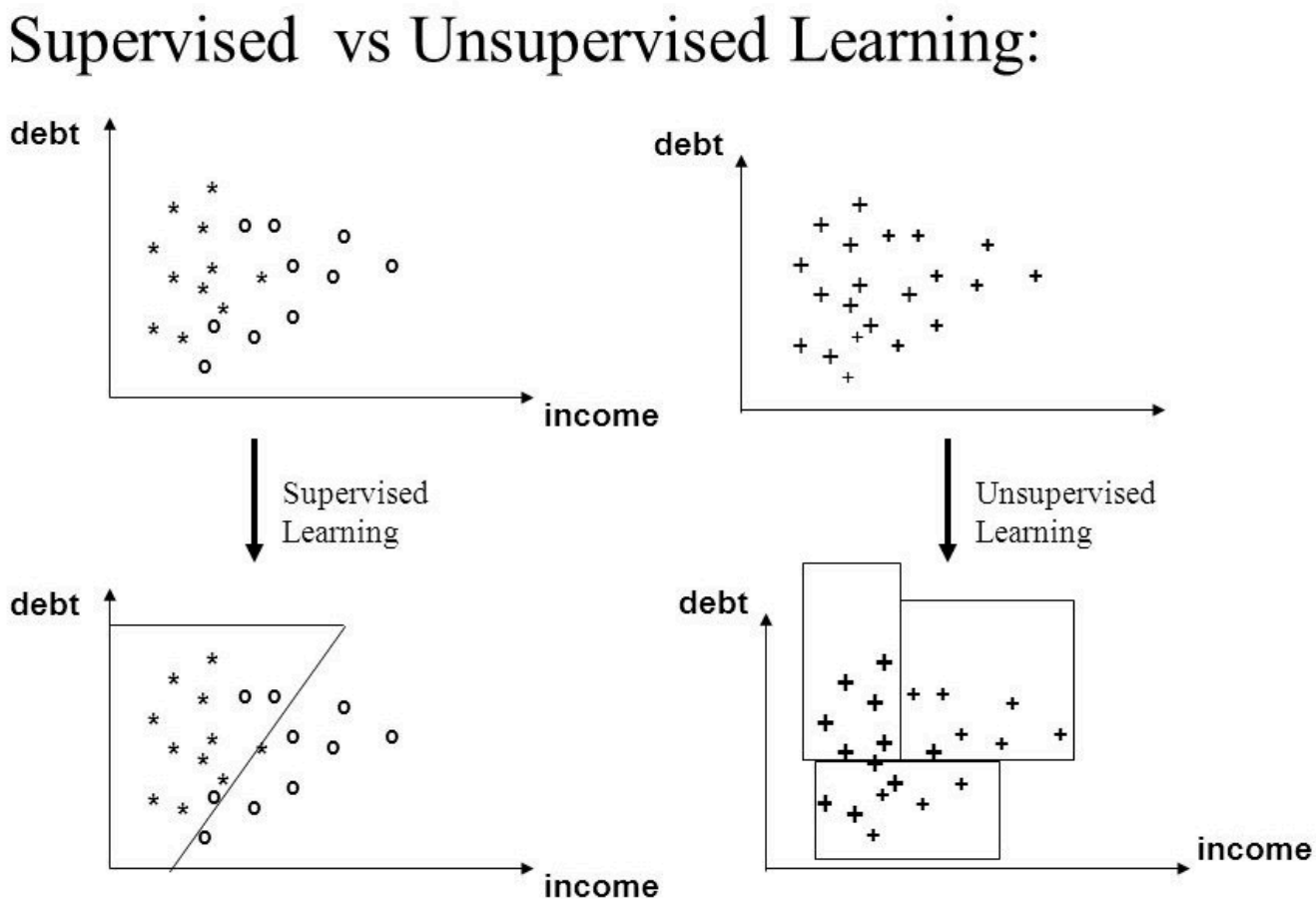
Regression:
[Soccer Fantasy Score prediction]



Classification:
[SVM on 2 classes of the Wine dataset]

Reference: Dr. Sebastian Raschka, An Introduction to Supervised Machine Learning and Pattern Classification: The Big Picture

Karışılaştırma:

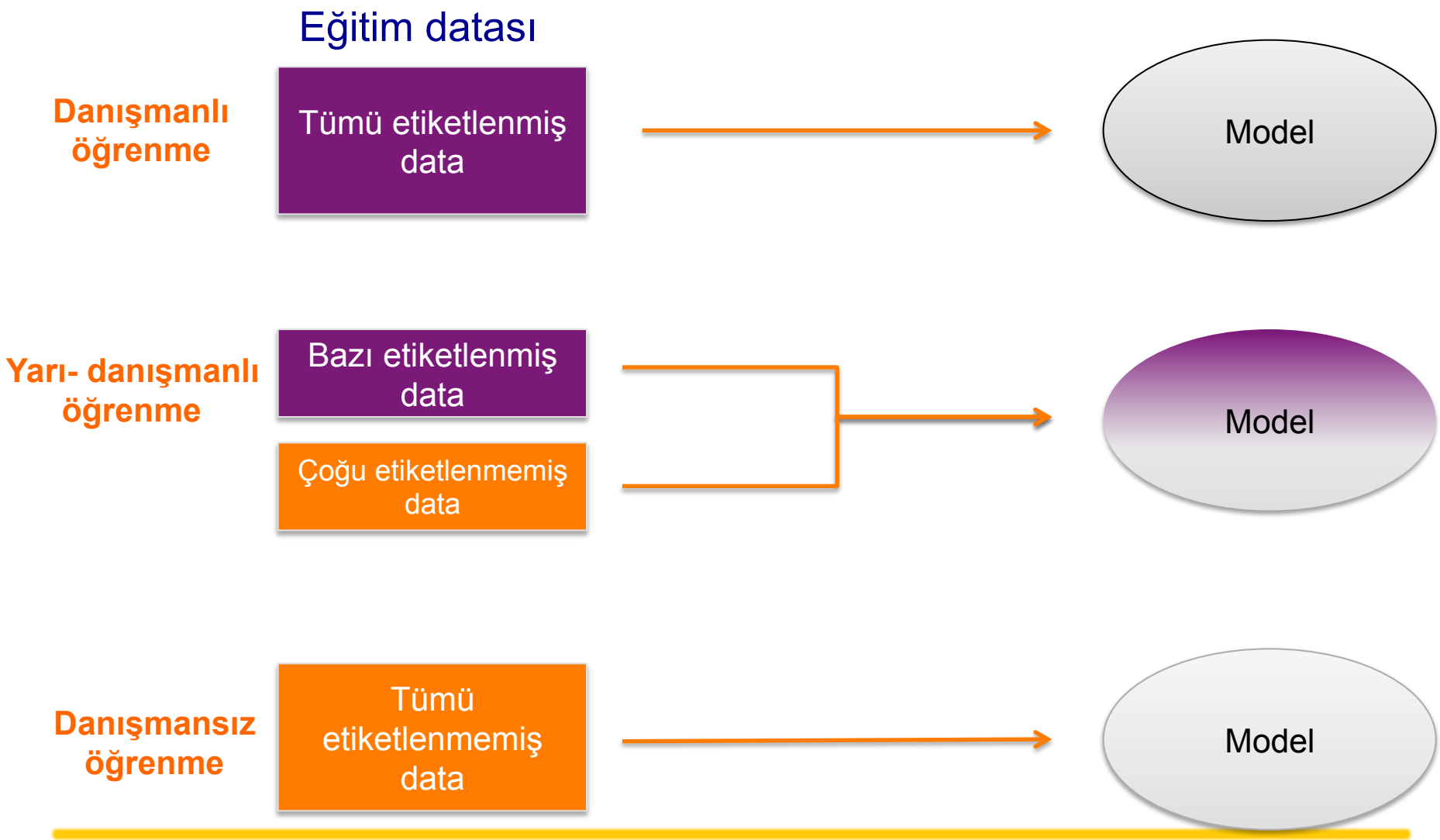


Reference: http://datacafeblog.com/supervised-vs-unsupervised-learning-know-the-difference/

Destekli (Takviyeli) Öğrenme:

- Bazen öğretici, sisteme beklenen sonucu tam söyleyemez. Ama sistemin ürettiği sonuç için “doğru/ yanlıs” seklinde fikir belirtir.
- Bu öğrenme sekli de takviyeli (reinforcement) öğrenme adıyla anılır.
- Boltzmann makinesi, LVQ ve genetik algoritma örnek olarak sayılabilir.

Yarı danışmanlı öğrenme:



Kümeleme nedir?

- Geçmisteki verilerin sınıfları/etiketleri verilmediği/bilinmediği durumlarda verilerin birbirlerine yakın benzerliklerinin yer aldığı kümelerin bulunmasıdır.

Kümeleme:

- Kümeleme algoritmaları danışmansız(eğiticisiz) öğrenme metotlarıdır.
- Örneklere ait sınıf bilgisini kullanmazlar.
- Temelde verileri en iyi temsil edecek vektörleri bulmaya çalışırlar.
- Verileri temsil eden vektörler bulunduktan sonra artık tüm veriler bu yeni vektörlerle kodlanabilirler ve farklı bilgi sayısı azalır.

Bir başka deyişle kümeleme:

- Danışmansız öğrenme kullanılararak yapılan kümelemede amaç:
  - veri kümesi içerisindeki veri örneklerini sadece özellik (feature) vektörlerine göre gruplamaktır.
- Bunun için örneklerin birbirine benzerliği gözetilir.
- Kümeleme literatüründe “benzerlik” terimi “uzaklık” terimiyle zıt anlamda kullanılmaktadır.
- Birbirine benzer olan örnekler aynı kümeye ve birbirinden uzak olan örnekler farklı kümelere yerleştirilmeye çalışılır.
- Küme sayısı genellikle dışarıdan verilir.

Örnek:

Elimizde tek boyutlu 10 örnek içeren bir verimiz olsun.

12 15 13 87 4 5 9 67 1 2

Bu 10 farklı veriyi 3 farklı veri ile temsil etmek istersek

12 12 12 77 3 3 3 77 3 3 şeklinde ifade ederiz.

Gerçek değerler ile temsil edilen değerler arasındaki farkı minimum yapmaya çalışır.

Yukarıdaki örnek için 3 küme oluşmuştur.

12-15-13 örnekleri 1. kümede

87-67 örnekleri 2. kümede

4-5-1-2-9 örnekleri 3. kümede yer almaktadır.

Resim kümeleme

10*10 luk bloklar kmelenmiş

Kümeleme metodları

- K-Means
- Self Organizing Map (SOM)

K-Means:

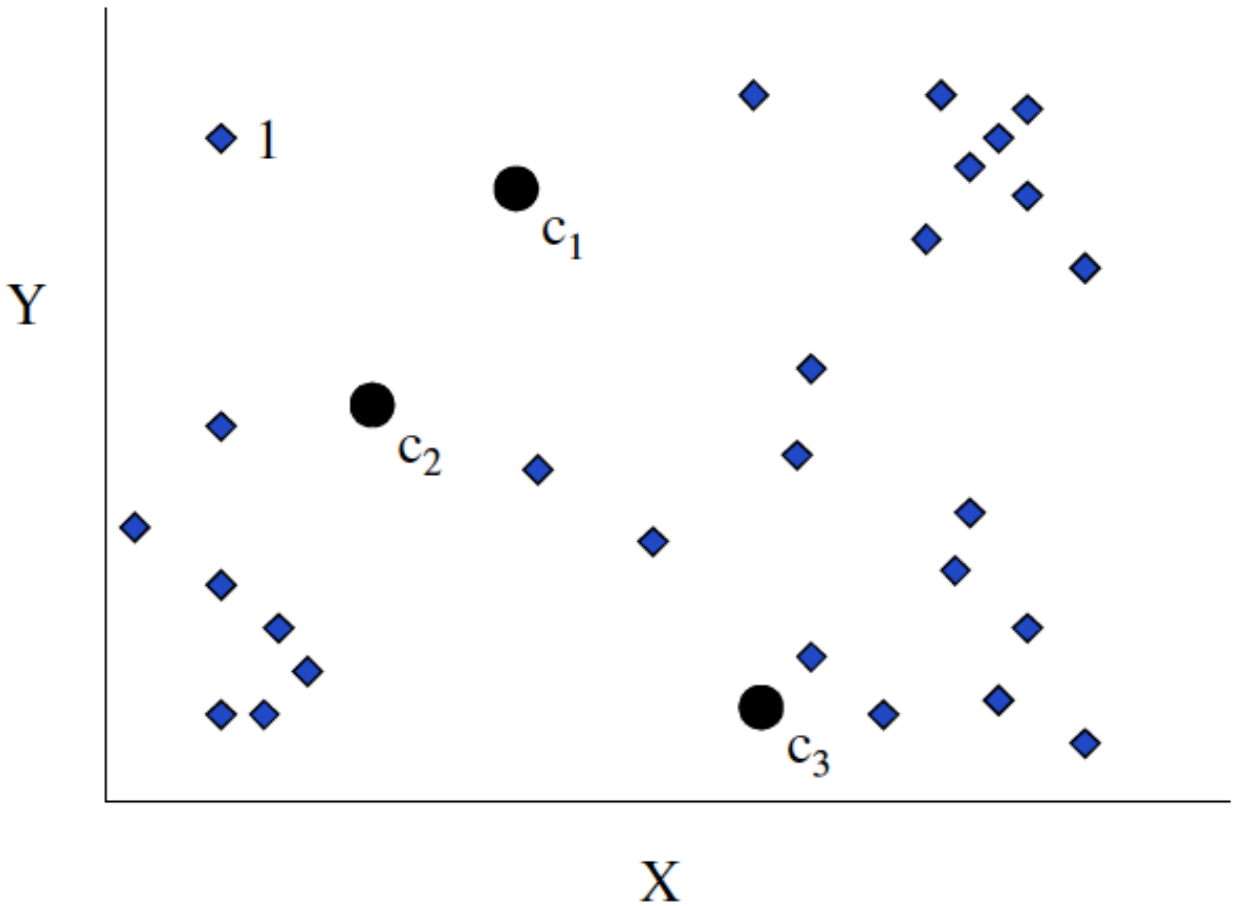
- Kümeleme algoritmalarının en basitidir.
- Veriyi en iyi ifade edecek K adet vektör bulmaya çalışır.
- K sayısı kullanıcı tarafından verilir.
- Nümerik değerler için çalışır.
- i adet merkez belirlemek için :
  - Rasgele K adet küme merkezi atanır (C1, C2, ...,Ck)
  - Her örnek en yakınındaki merkezin kümesine atanır.
  - Ci'ler tekrar hesaplanır (her kümedeki örneğin ortalaması alınır)
  - Ci' lerde değişiklik olmuş ise 2. ve 3. adımlar tekrar edilir. Bu isleme küme değiştiren örnek kalmayıncaya kadar devam edilir, aksi taktirde işlem sonlandırılır.

K-Means

Solda256 renkle temsil edilen resim, sağda ise K-Means kullanılarak 16 renge indirilmiş resim görülmektedir.

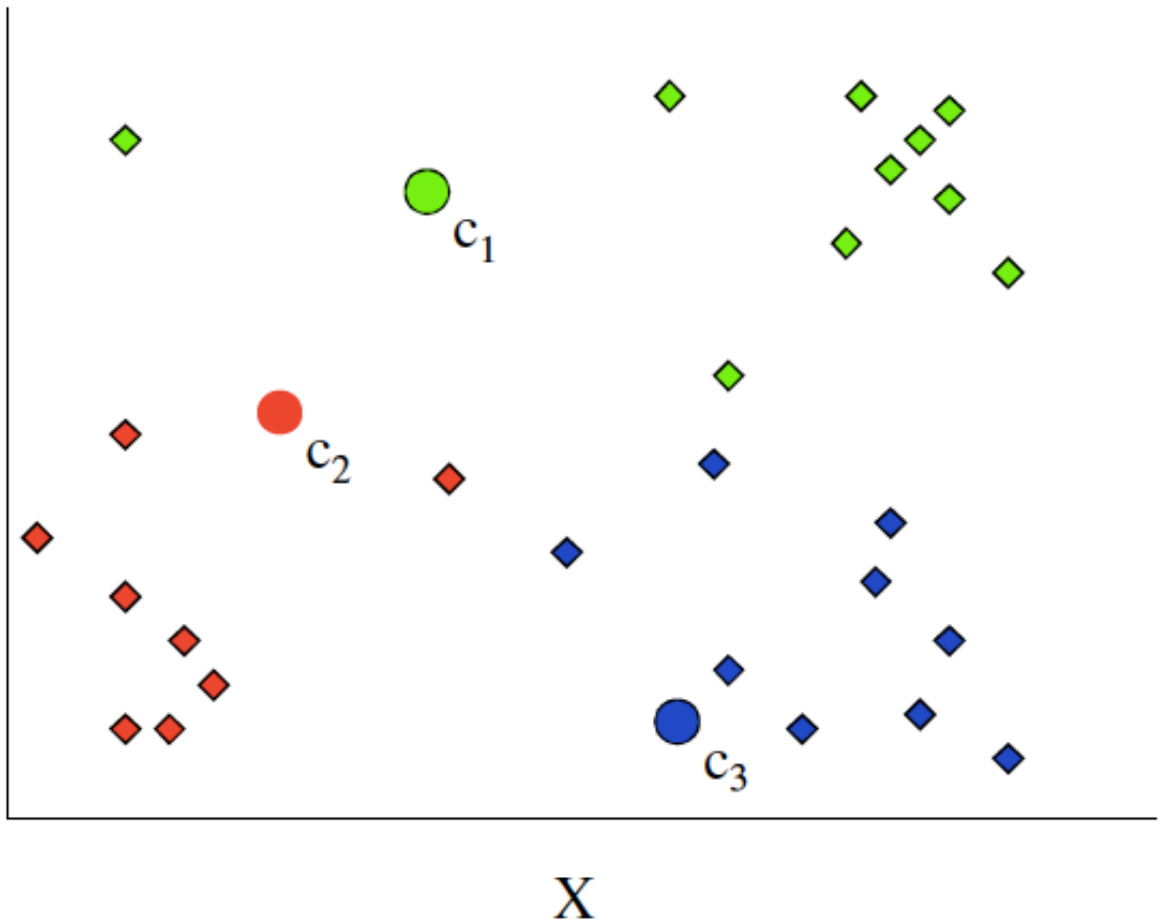
Adım 1

Rasgele 3 küme merkezi ata.



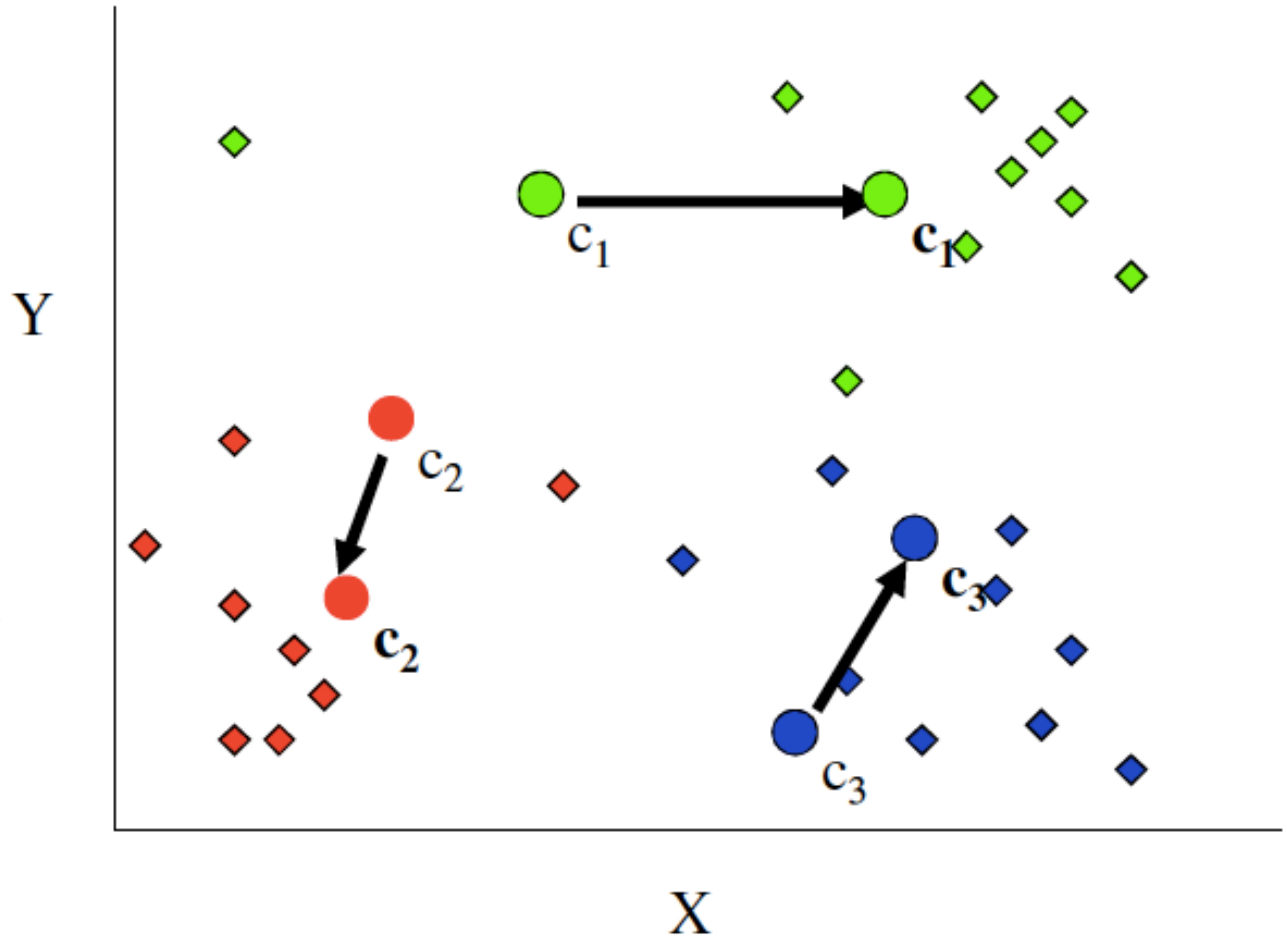
Adım 2

Her örnek
en yakınındaki
merkezin
kümesine atanır.



Adım 3

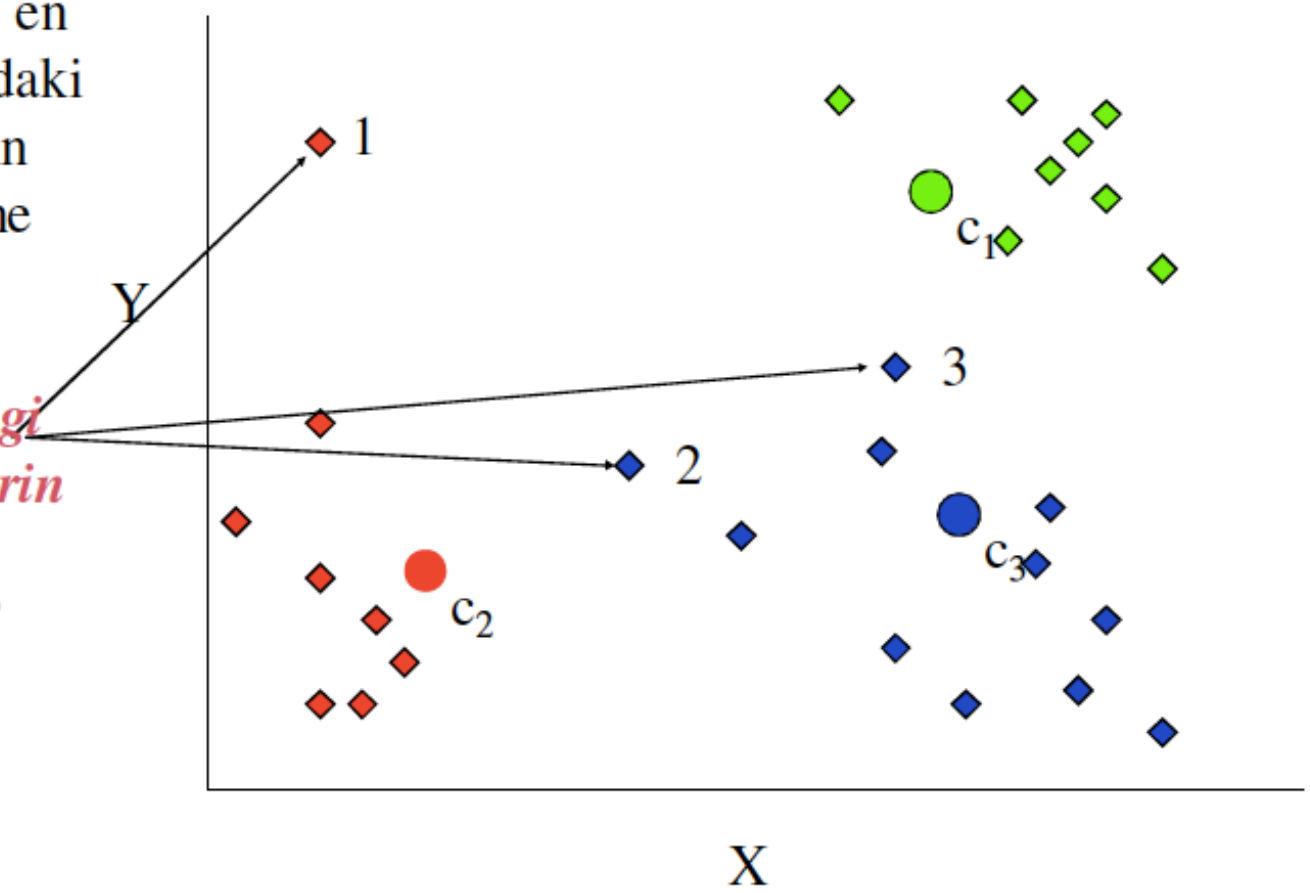
Merkezleri kendi kümelerinin merkezine götür.



Adım 4

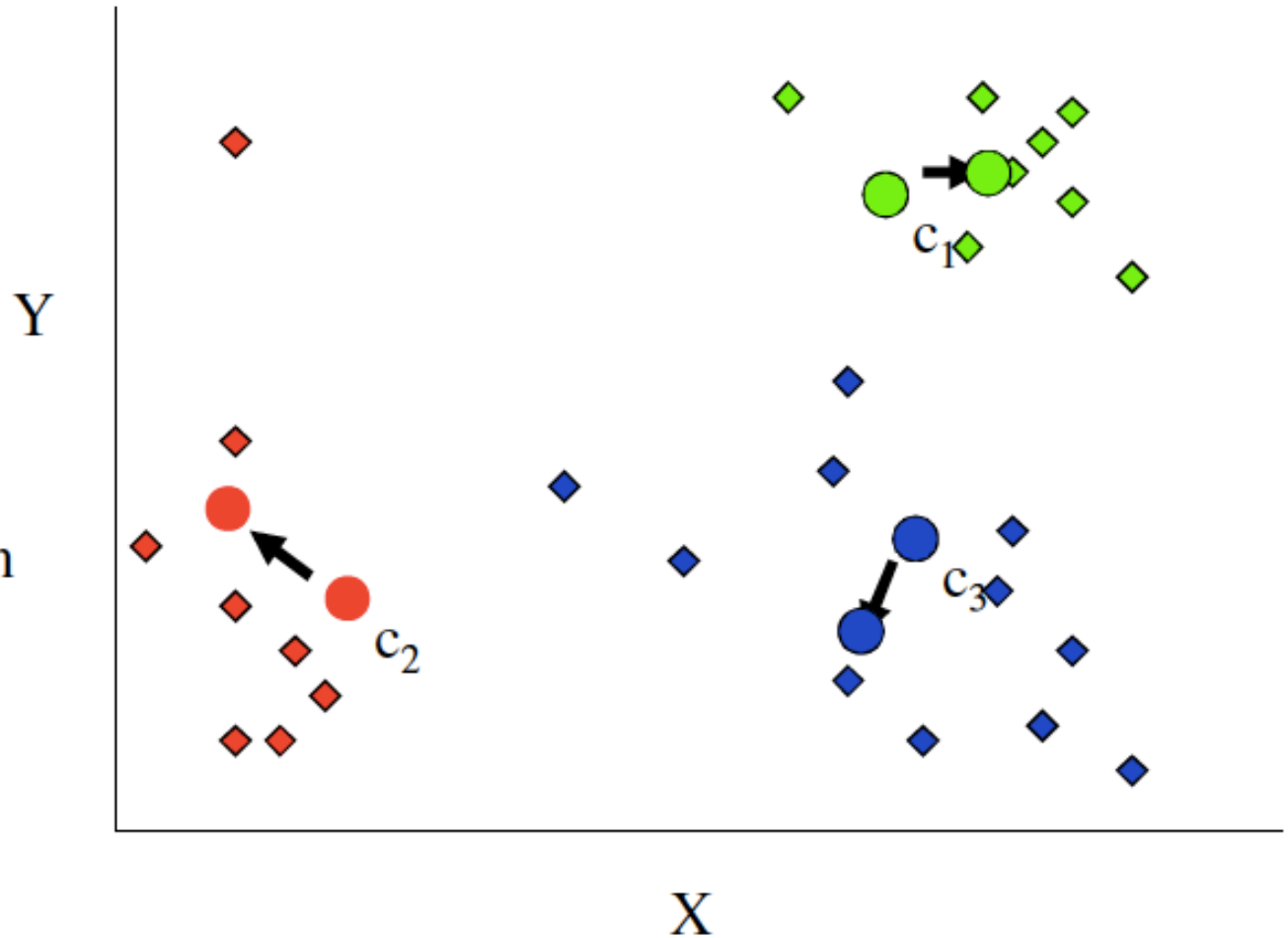
Her örneği yeniden en yakınındaki merkezin kümesine ata.

Q: Hangi örneklerin kümesi değişti?



Adım 5

Merkezleri kendi kümelerinin merkezine götür.



Self Organizing Map

- K-Means algoritmasında merkez noktalar arasında herhangi bir ilişki yok iken SOM'da merkez noktalar 1 veya 2 boyutlu dizi içerisinde yer alırlar.
- K-Means algoritmasında sadece kazanan merkez güncellenirken, SOM 'da bütün merkezler kazanan nöronun komşuluklarına göre güncellenirler.
- Yakın komşular uzak komşulara göre daha fazla hareket eder.

Self Organizing Map

- SOM merkezleri 1 boyutlu bir dizide birbirlerine komşudur, başlangıçta rasgele atandıkları için yığılma mevcuttur ancak eğitim tamamlandıktan sonra SOM merkezleri düzgün dağılmıştır.

- Makale: ”GEZGİN ROBOT TASARIMI VE KENDİ KENDİNE ORGANİZE OLAN HARİTALAR İLE ROBOTUN ENGELLERDEN UZAKLAŞMASI”

Özellik çıkarma:

- Veriye ait olan birçok özellikten bazıları ilgili verinin kümesini/sınıfını belirlemede önemli rol oynar. Bu gibi durumlarda özellik kümesinin bir alt kümesi seçilir (özellik seçimi) veya bu özelliklerin birleşiminden yeni özellikler elde edilebilir (özellik çıkarımı).

Online ve Off-line öğrenme:

- Çevrimiçi (online) öğrenme: Gerçek zamanlı olarak çalışan sistemlerde öğrenme aşamasının çalışma anında da sürdürülmesi gerekiyorsa kullanılır.

- Çevrimdışı (offline) öğrenme: Uygulama sistemi henüz çalışmaz iken eğitim gerçekleştirilir, eğitilmiş yöntem sisteme yüklenir ve sistem çalıştırılır.

Öğrenme Kuralları:

- Hebb
- Delta
- Hopfield
- Kohonen

Hebb Öğrenme Kuralı:

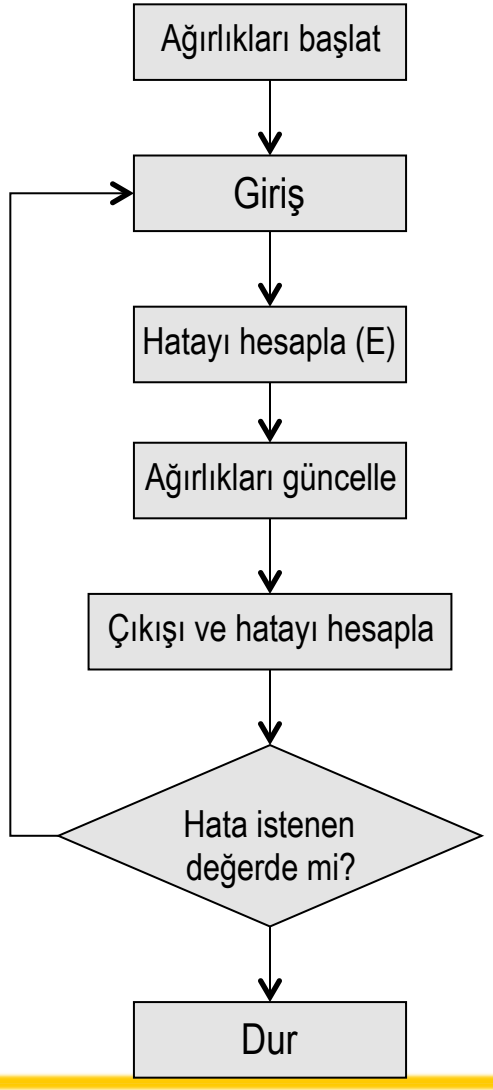
- 1949'da geliştirilen ilk öğrenme kuralıdır.
- "Bir hücre, bağlı olduğu diğer hücreleri etkiler" prensibine dayalıdır.
- Bu öğrenme kuralı geliştirilerek farklı öğrenme kuralları geliştirilmiştir.

Delta Kuralı:

- Beklenen sonuç ve hesaplanan sonuç arasındaki karesel fark, sistemin hatasıdır.
- Bu hatayı indirgemek için hücreler arasındaki bağlantılar sürekli değiştirilir.
- Çok katmanlı algılayıcı ağlar bu kurala göre eğitilirler.

$$E = \frac{1}{2}\sum_{i=1}^{n}(t_i - y_i)^2$$

Delta Kuralı



Hopfield Öğrenme Kuralı:

- Beklenen sonuç ile hesaplanan sonuç aynı ise hücreler arası bağlar belirli bir oranda güçlendirilir, aksi durumda bağlar zayıflatılır.
- Recurrent ve Elman ağları bu kural ile eğitilirler.

Kohonen Öğrenme Kuralı:

- Bu danısmansız modelde hücreler yarış halindedir. En büyük sonucu üreten hücre yarısı kazanır.
- Kazanan hücrenin ve komsularının bağları güçlendirilir.
- ART (Adaptive Resonance Theory) ve Kohonen tarafından gelistirilen SOM (Self Organizing Map) ağları örnek olarak verilebilir.

Aşırı eğitim problemi (ezberleme):

- Veri ile iterasyonel eğitim yapan modellerin hepsinde öğrenme sürecinin zamanı gelince durdurulması gerekir.
- Eğitim durdurulmazsa öğrenilmesi gereken veri içerisindeki tüm örnekler sistem tarafından ezberlenir ve bilinmeyen örnekleri tahmin yeteneği azalır.
- Aşırı eğitim (overfitting) denilen bu istenmeyen durum makine öğrenmesinin temel amacı olan genelleştirme kavramına zarar verir.

Referanslar:

- Dr. Umut Orhan, Makine Öğrenmesi Ders Notu.
- http://datacafeblog.com/supervised-vs-unsupervised-learning-know-the-difference/
- Dr. Sebastian Raschka, An Introduction to Supervised Machine Learning and Pattern Classification: The Big Picture
- http://dataaspirant.com/2014/09/19/supervised-and-unsupervised-learning/
- Dr. Banu Diri, “Makine Öğrenmesine giriş” ders notları.