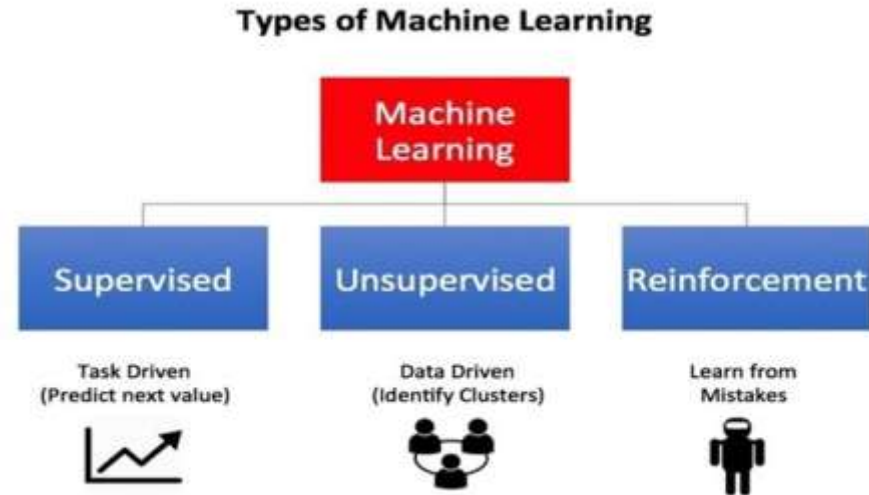


# Q LEARNING

EMİR ÇUBUKÇU-19360859052

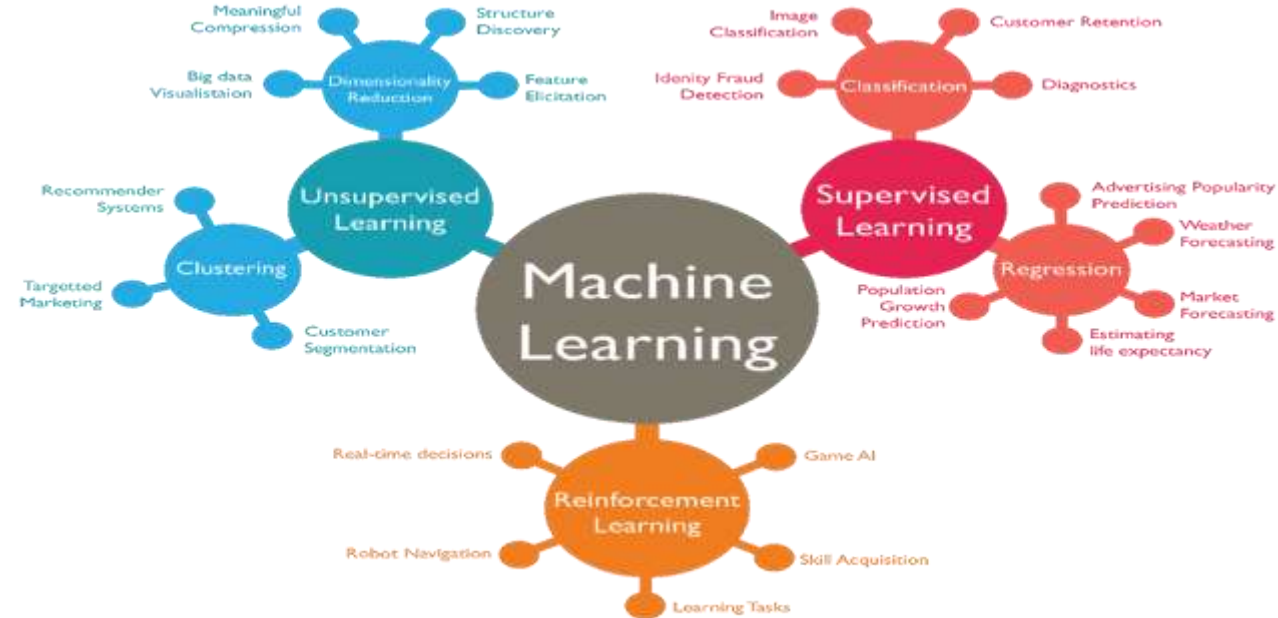
# MAKİNE ÖĞRENMESİ NEDİR?

- Makine öğrenmesi, insanların öğrenme şekillerini taklit etmek için veri ve algoritmaların kullanımına odaklanıp doğruluğunu kademeli olarak artıran bir yapay zeka (AI) ve bilgisayar bilimi dalıdır.



# PEKİŞTİRMELİ ÖĞRENME

- Makine öğrenmesi algoritmaları genel olarak üç ana başlık altında toplanır. Bu başlıklardan ilki Eđiticisiz Öğrenme (Unsupervised Learning); K-Means, Temel Bileşen Analizi gibi yöntemleri kapsamaktadır.
- İkinci ana başlık Eđiticili Öğrenme (Supervised Learning) ise Karar Ağaçları, Lojistik Regresyon Analizi ve Destek Vektör Makineleri gibi yöntemleri kapsamaktadır.

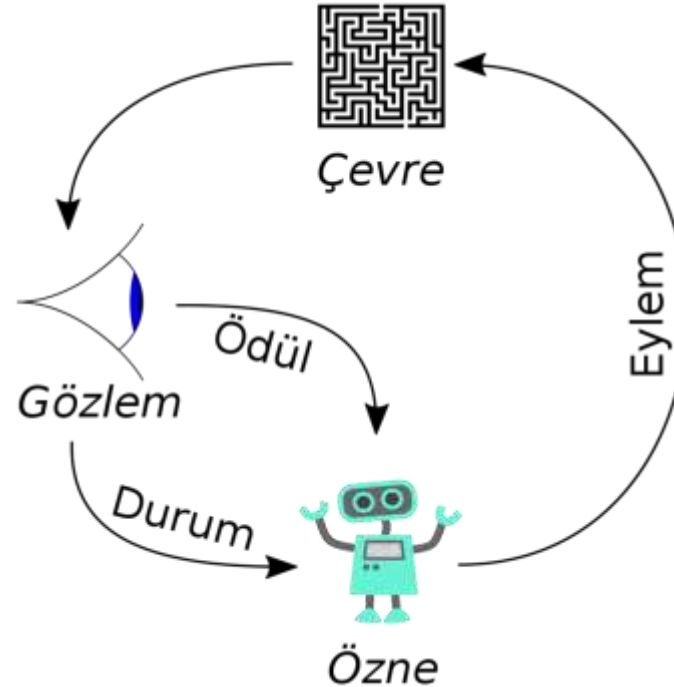


# PEKİŞTİRMELİ ÖĞRENME

- Makine öğrenmesi algoritmalarının üçüncü ana başlığı ise pekiştirmeli öğrenme algoritmaları oluşturmaktadır.
- Pekiştirmeli öğrenme, davranışçılıktan esinlenen, öznelerin bir ortamda en yüksek ödül miktarına ulaşabilmesi için hangi eylemleri yapması gerektiğiyle ilgilenen bir makine öğrenmesi yaklaşımıdır.

# PEKİŞTİRMELİ ÖĞRENME

- Diğer yöntemlerin aksine Pekiştirmeli Öğrenme ne öğrenilecek veriye ne de verinin etiket bilgisine ihtiyaç duymaktadır.
- Özne bir eylemde bulunarak çevre ile etkileşir. Çevreyi gözlemleyerek yeni durumu ve elde ettiği ödülleri değerlendirir. Eylemler ve gözlemler bir döngü içinde birbirini takip eder.



# Q LEARNING

- İlk olarak Christopher Watkins and Peter Dayan tarafından 1992 yılında literatüre kazandırılan Q Öğrenme (Q Learning) yöntemi, 2013 yılında DeepMind yapay zeka şirketinin kurucuları tarafından yayınlanan “Playing atari with deep reinforcement learning” makalesi ile oldukça popüler hale gelen bir pekiştirmeli öğrenme yöntemidir.



# Q LEARNING

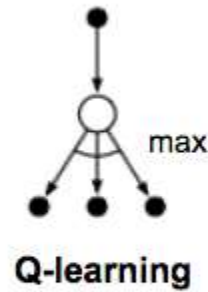
- Bu makalede Atari 2600 oyun konsolunda bulunan Pong, Breakout, Space Invaders, Seaquest ve Beam rider oyunlarında Deep Q learning ile eğitilen model sayesinde uzman bir oyuncu seviyesinde puanlara ulaşabilmiştir.



	Beam Rider	Breakout	Enduro	Pong	Q*bert	Seaquest	Space invaders
İnsan	7456	32	368	-3	18900	28010	3690
Model	4092	168	470	20	1952	1705	581

# Q LEARNING AMACI

- Q-Learning algoritmasındaki temel amaç bir sonraki hareketlerini inceleyip yapacağı hareketlere göre kazanacağı ödülü görmek ve ödülünü maksimum yapmasını sağlayacak hareketleri belirlemektir.





# Q LEARNING TANIMLAR

- Ajan(Agent):Modeli eğitirken kullandığımız hareketleri gerçekleştirdiğimiz ve hareketlerin sonucunda öğrenen etken
- Ödül(Reward):Ajanın yaptığı hareketlere göre geri dönüş yaptığımız sayısal değer
- Durum Değeri (State value):Ajanın bulunduğu konumu ifade eden sayı
- Q-Table: Ajanın her tecrübesinin kaydedildiği tablo
- Reward table: Ödülün bulunduğu ve bütün yolların puanlandırıldığı ve kaydedildiği tablo

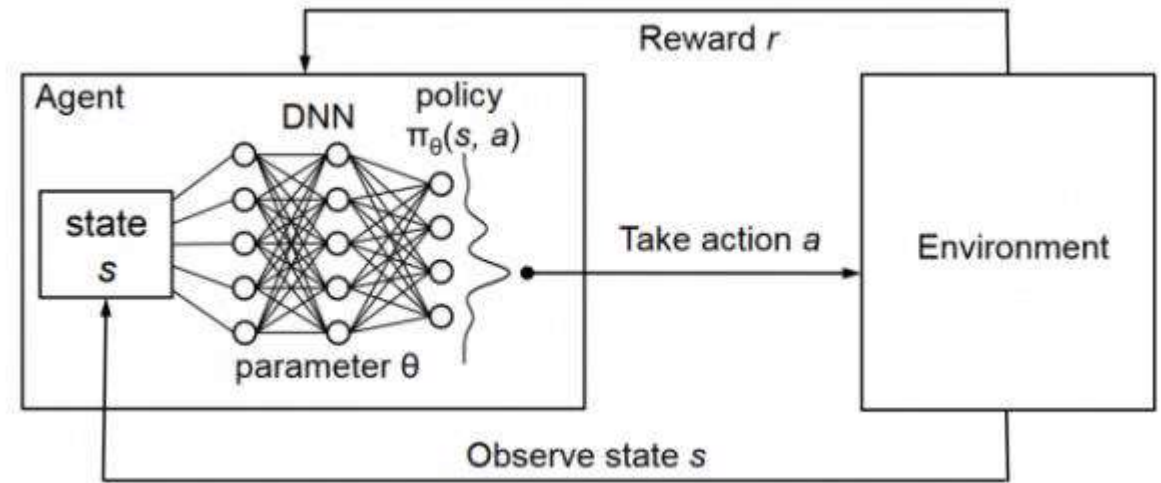
# Q LEARNING İŞLEYİŞİ

- Ajan iterasyonlar şeklinde çalışırlar.
- Ajan başlangıçta her zaman rastgele hareket eder çünkü Q-table boş yani bulunduğu çevreyi bilmez. Ödülü bulana kadar devam eder.
- Ajan yaptığı tüm hareketleri ödül tablosunu da kullanarak Q-table a kaydeder. Kaydetme işlemi aşağıdaki formül ile yapılır.

$$Q(s,a) = Q(s,a) + lr*(r(s,a) + \gamma*\max_{a'}(Q(s',a')) - Q(s,a))$$

# DEEP Q LEARNING

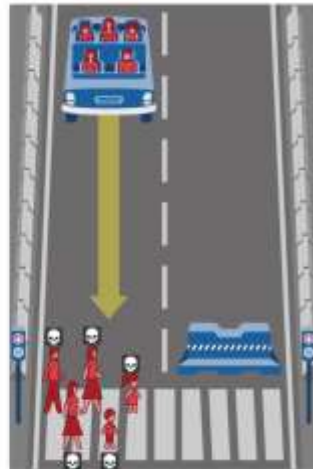
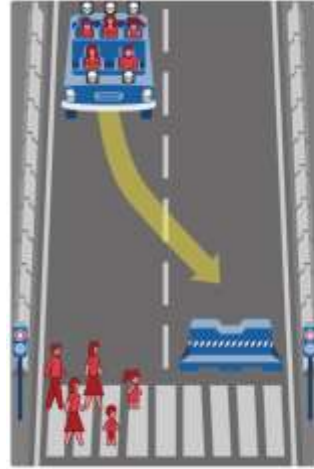
- Deep Q Learning de aynı Q Learning gibi bir Reinforcement Learning algoritmasıdır. En önemli farkı ise Q-Table yerine yapay sinir ağlarının kullanılmasıdır.



# DOUBLE Q LEARNING

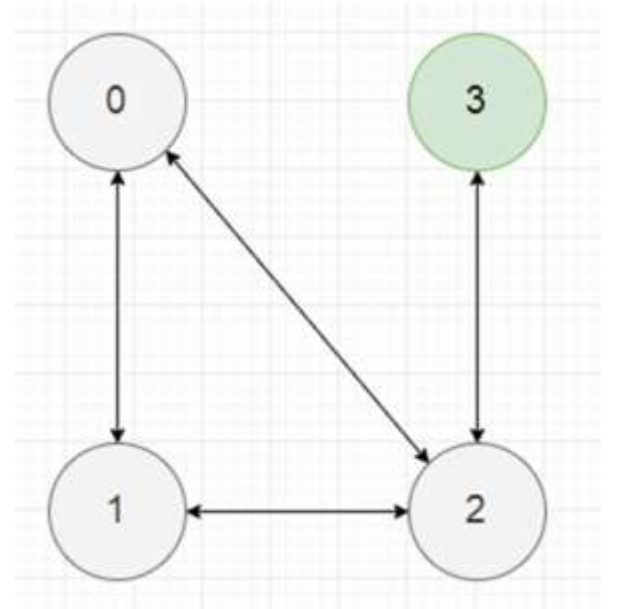
- Q learning kullanılırken her iterasyonda aynı Q-table kullanıldığı için gürültülü ortamlarda öğrenme hızı yavaşlamaktadır.
- Bu sorunu atlatabilmek için simetrik iki farklı değer ile birlikte iki farklı Q table oluşturularak öğrenme hızı artırılmıştır.

# PEKİŞTİRMELİ ÖĞRENME KULLANIM ALANLARI



# Q LEARNING ÖRNEK

- 0 numaralı düğümde bulunan bir ajanı 3 numaralı düğüme götürmek için bir sistem eğitmek için öncelikle bir ödül tablosu oluşturmalıyız.



## Q LEARNING ÖRNEK

- Ödül tablosunu oluştururken hedef düğüme büyük bir değer girilir.
- Gidilemeyen ya da gidilmesi istenmeyen düğümler için ise negatif değerler atanır.
- Kalan düğümler ise herhangi bir negatif veya pozitif etkisi olmadığı için 0 değeri verilir.

	0	1	2	3
0	-1	0	0	-1
1	0	-1	0	-1
2	0	0	-1	100
3	-1	-1	0	-1

# Q LEARNING ÖRNEK

- Başlangıçta Q-table sıfırlardan oluşmaktadır.

	0	1	2	3
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0



# Q LEARNING ÖRNEK

- Ajanın gidebileceği iki düğüm vardır ve bu düğümlerden birisini seçmelidir. Q-Tablosunda gideceği iki düğümün de değeri sıfırdır bu yüzden rastgele bir seçim yapar.(Gamma değerini 0.8, öğrenme değerini de 0.7 alalım)

$$Q(0,2) = Q(0,2) + 0.7 * (0 + 0.8 * \max(Q(2,1), Q(2,3)) - Q(0,2))$$

$$Q(0,2) = 0 + 0.7 * (0 + 0.8 * 0) - 0$$

$$Q(0,2) \leftarrow 0$$

# Q LEARNING ÖRNEK

- Ajan herhangi bir ödüle ulaşmadığından Q-table daki değeri değişmemiştir.
- Ajanın 2. düğümden devam edebileceği 3 farklı düğüm bulunmaktadır.

	0	1	2	3
0	0	0	0	0
1	0	0	0	0
2	0	0	0	0
3	0	0	0	0

	0	1	2	3
0	-1	0	0	-1
1	0	-1	0	-1
2	0	0	-1	100
3	-1	-1	0	-1

# Q LEARNING ÖRNEK

- 3 numaralı düğüme gittiğini varsayarak Q learning algoritmasını uygulayalım.

$$Q(2,3) = Q(2,3) + 0.7 * (100 + 0.8 * \max(Q(3,2)) - Q(2,3))$$

$$Q(2,3) = 0 + 0.7 * (100 + 0.8 * 0) - 0$$

$$Q(2,3) \leftarrow 70$$

# Q LEARNING ÖRNEK

- Son durumda Q table yandaki gibi oluşmaktadır.

	0	1	2	3
0	0	0	0	0
1	0	0	0	0
2	0	0	0	70
3	0	0	0	0