Pekiştirmeli Öğrenmeye Giriş

Dr. Öğr. Üyesi Barış Akgün

Koç Üniversitesi

Yapay Öğrenme Yaklaşımları

1. Paranın değeri?
   - Gözetimli Öğrenme
2. Birbirine benzeyen paralar?
   - Gözetimsiz Öğrenme
3. Parayı çoğaltmak?
   - Pekiştirmeli öğrenme

Gözetimli Öğrenme

Veri: Ölçüm-İşaret ikilileri


1 kuruş


25 kuruş


5 kuruş


50 kuruş


10 kuruş


1 TL

Yeni Ölçüm: İşareti nedir?


?

Gözetimsiz Öğrenme

Birbirine benzeyen sikkeler?

Pekiştirmeli Öğrenme

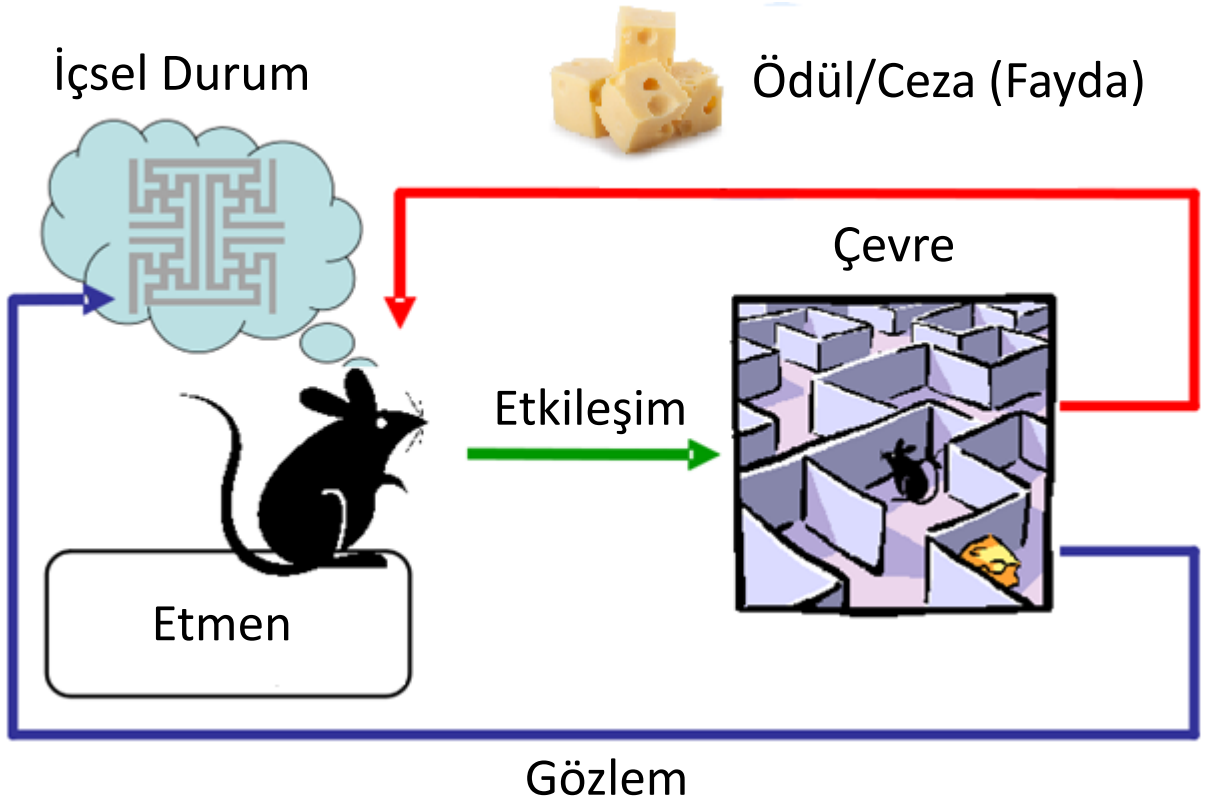
En yüksek puanı yapmak

Kısa Beyin Fırtınası

- Son zamanlarda "yapay zekada" alanındaki gelişmeler nedir?
- Dezavantajlar?
- Yapay Öğrenme Yaklaşımları:
  - Gözetimli – Gözetimsiz – Ödül/Maliyet Tabanlı (Pekiştirmeli)
- Biz nasıl öğreniyoruz?

- Çevre ile etkileşime girerek bir amaca ulaşmak/fayda sağlamak (mutluluk, hayatta kalmak, yemek vb.)
  - Kim yapıyor bunu? – Etmen (ing. a*gent*)

Örnek:

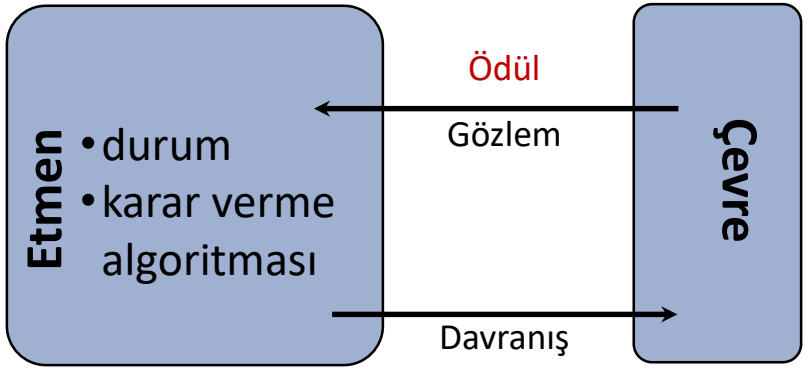


Peynir olmak zorunda değil ☺

Neler Gerek?

- Bir etmenin çevre ile etkileşime girerek bir amaca ulaşması/fayda sağlaması

- Gözlemler: Kendisini, diğer etmenleri ve çevreyi algılamak
- Durum: Kendisini, diğer etmenleri ve çevreyi anlatmak
- Davranışlar: Etkileşime girmek
- Ödül: Neyin iyi neyin kötü olduğunu bilmek
- Politika: Durum-davranış bağlantısı



- **Amaç**: Beklenen faydayı en-iyileyecek politikayı öğrenmek

Uygulama Örnekleri

- Kaynak Yönetimi
- Trafik Işığı Kontrolü
- Otonom Sürüş
- Robotik
- Kimya
- Öneri Sistemleri ve Reklamcılık
- Yatırım ve Finans
- Büyük ölçekli ve/veya karmaşık sistem kontrolü
- Oyunlar
- ve daha fazlası…

Yürümeyi Öğrenmek

İlk

Son

[Kohl and Stone, ICRA 2004]

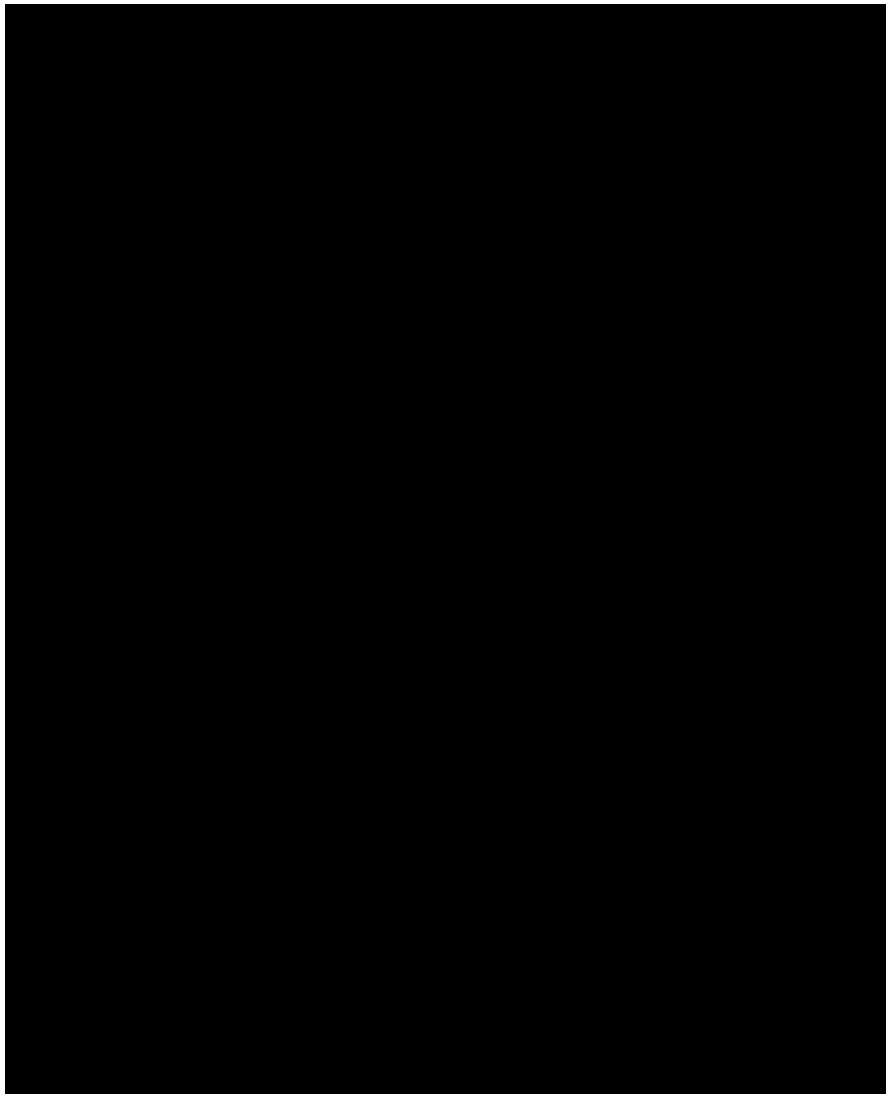
Helikopter Uçurmak

[Andrew Ng et al]

Mutfak Robotu

[Kormushev et al]

Atari Örneği

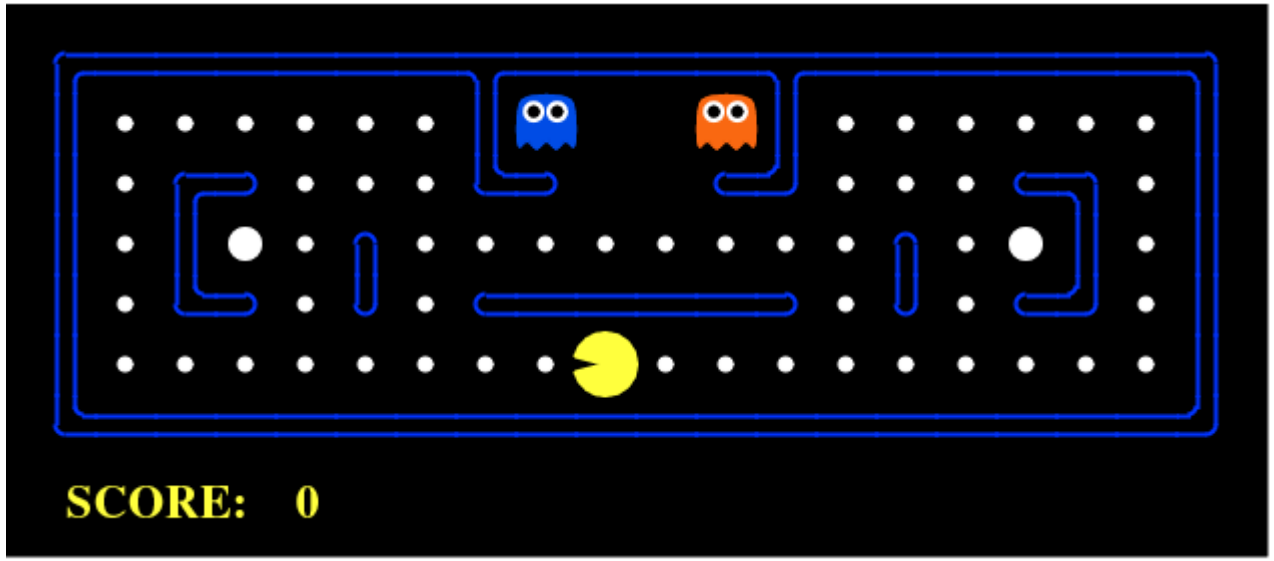


Etkileyici Kısmı: Aynı algoritma aynı parametreler ile bir çok oyunu oynamayı öğrendi

Örnek: Pacman

- Etmen
- Çevre
- Durum
- Gözlemler
- Davranışlar
- Ödül

- **Soru**: Pekiştirmeli öğrenme neden farklı?



Daha gerçekçi yapmak isteseydik

- Pacman sadece önünü görseydi?
- Davranışlarda bir miktar rastgelelik olsaydı?
- Hayalet davranışları?
- 2 Pacman olsaydı?
- Duvarlar hareketli olsaydı?

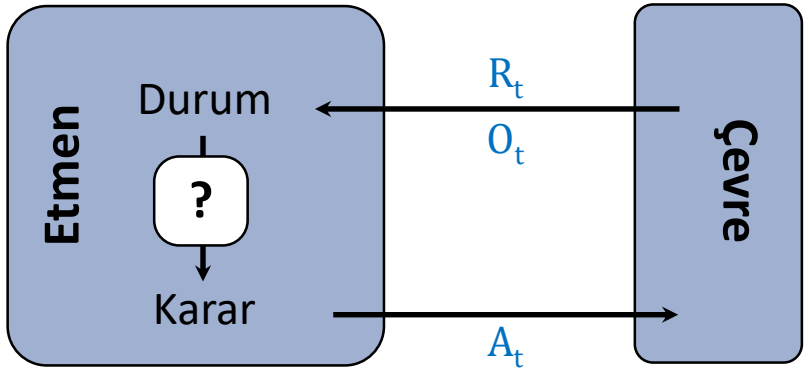
Ödüller

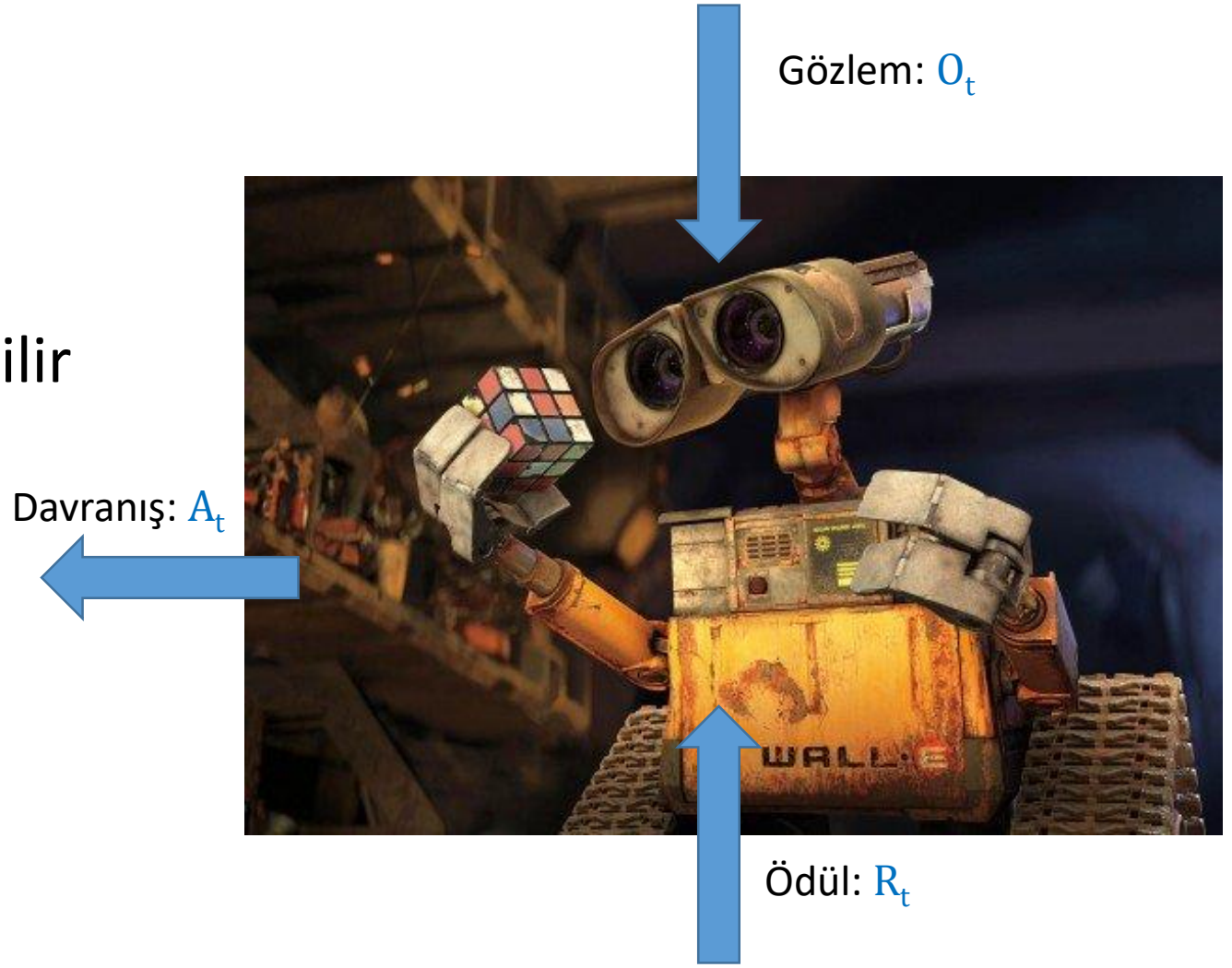
- Ödül: Anlık faydayı temsil eder
- Problemin amacını tanımlar
- Her davranıştan sonra elde edilir
- Etmenin amacı beklenen toplam ödülünü arttırmaktır

- Beyin fırtınası:
  - Robot yürümesi
  - Yatırım portföyü
  - Helikopter hareketleri
  - Tavla
  - Reklam gösterimi

Algı-Davranış-Ödül

- Her zaman adımında(t):
  - Etmen çevreyi gözlemler: O_t
  - Etmen davranış sergiler: A_t
  - Etmen ödül alır: R_t
- Bir sonraki zaman adımına geçilir





Sıralı Kararlar Vermek

- Toplam faydayı arttırmak – Gelecekle ilgili!
- Davranışların uzun dönem sonuçları
- Ödül/ceza geç gelebilir
- Anlık ödül ve uzun vade değer karşılaştırılması

- Örnekler?

Geçmiş ve Durum

- Etmenin davranışları bir geçmiş oluşturur:

$$H_t = O_1, R_1, A_1, \dots, A_{t-1}, O_t, R_t$$

- Geçmiş t zamanına kadar her şeyi içerir
- Durum, geçmişe bakarak etmenin ve çevrenin şimdiki halini özetler

$$S_t = f(H_t)$$

- Etmen, duruma bakarak davranış kararı alır

Çevre Durumu ve Etmen Durumu

- Çevre durumu, çevreyi tamamen anlatır, ancak:
  - Etmen hepsini gözlemlemeyebilir
  - Gereksiz bilgi içerebilir

- Etmen durumu, kendisinin ve çevrenin içsel olarak anlatımıdır
  - Etmenin amacı için gereken bilgileri içermelidir
  - Karar vermek için kullanılır
  - Geçmişten çıkartılır

Markov Varsayımı

- Markov Durumu:

$$P(S_{t+1}|S_t) = P(S_{t+1}|S_t, S_{t-1}, \dots, S_1)$$

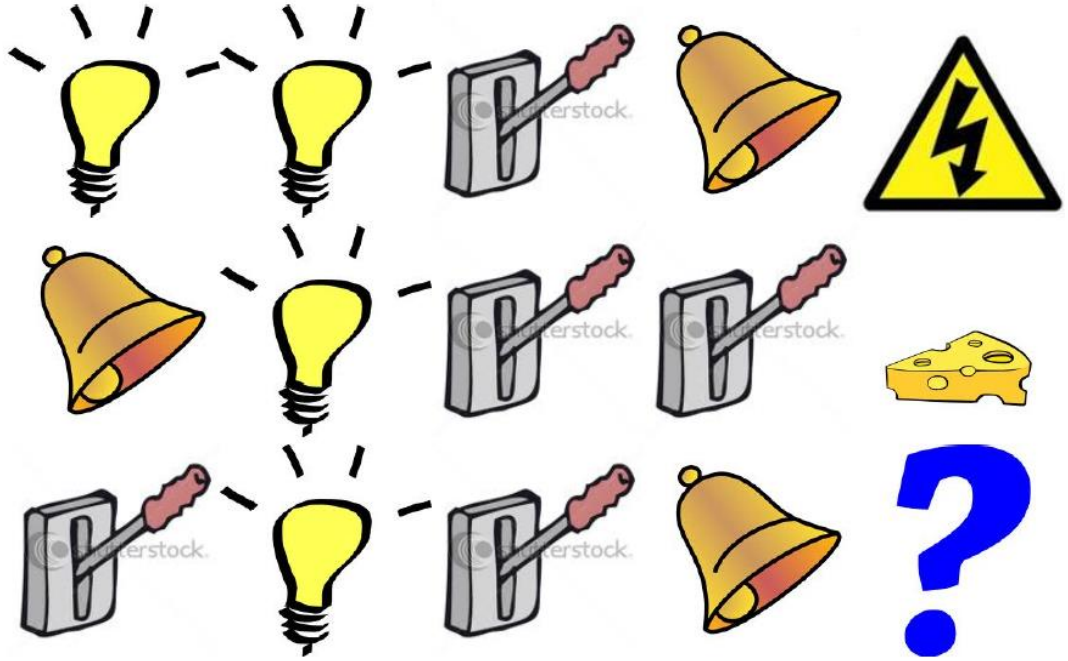
- "Şimdiki durum belirli ise gelecek geçmişten bağımsızdır"

$$H_{1:t} \rightarrow S_t \rightarrow H_{t+1:\infty}$$

- Bütün geçmişi hafızada tutmaya gerek yok!
  - Birden çok geçmiş adım tutulabilir

- Bu varsayım, gelecek bütün geçmişe bağlı olmadığı veya gözlemlenemeyen değişken olmadığı durumlarda geçerlidir

Ödüllerin Zamanlaması: Fare Örneği



İlk iki gözlem dizininden sonra, üçüncüsü için ne ödül beklenebilir?

David Silver'ın notlarından

Modeller

- Mevcut durum ve davranışa göre gelecekte ne olacak
- Geçiş Modeli: Sonraki durum nedir?

$$T(s, a, s') = \mathrm{P}(S_{t+1} = s' | S_t = s, A_t = a)$$

- Ödül Modeli: Anlık ödül nedir?

$$R(s, a) = \mathrm{E}(R_{t+1} | S_t = s, A_t = a)$$

- Farklı formları mevcuttur:

$$R(s), R(s, a), R(s, a, s')$$

- En genel model gösterimi:

$$P(S_{t+1} = s', R_{t+1} = r | S_t = s, A_t = a)$$

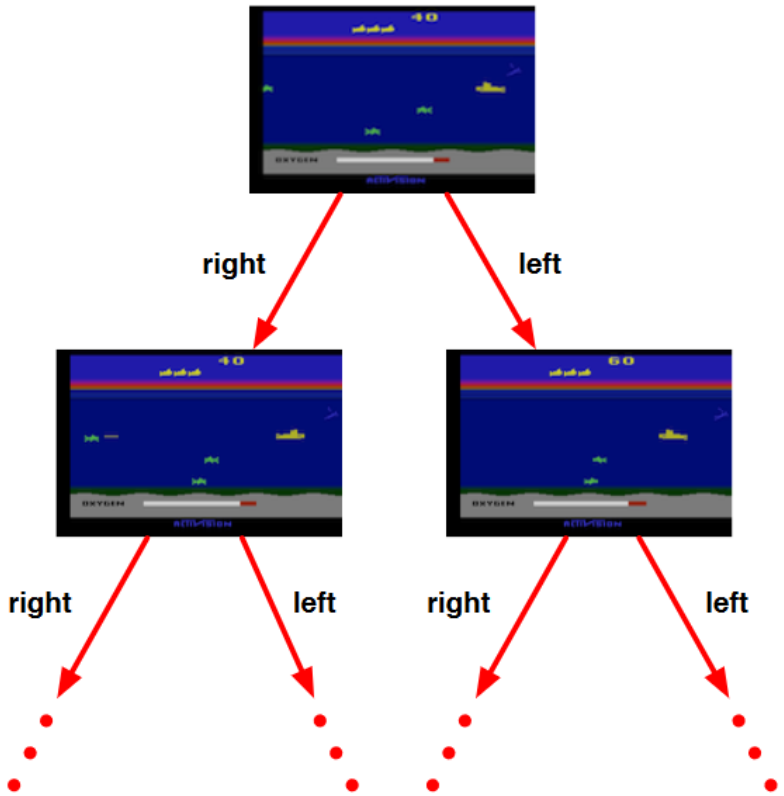
Öğrenme ve Planlama

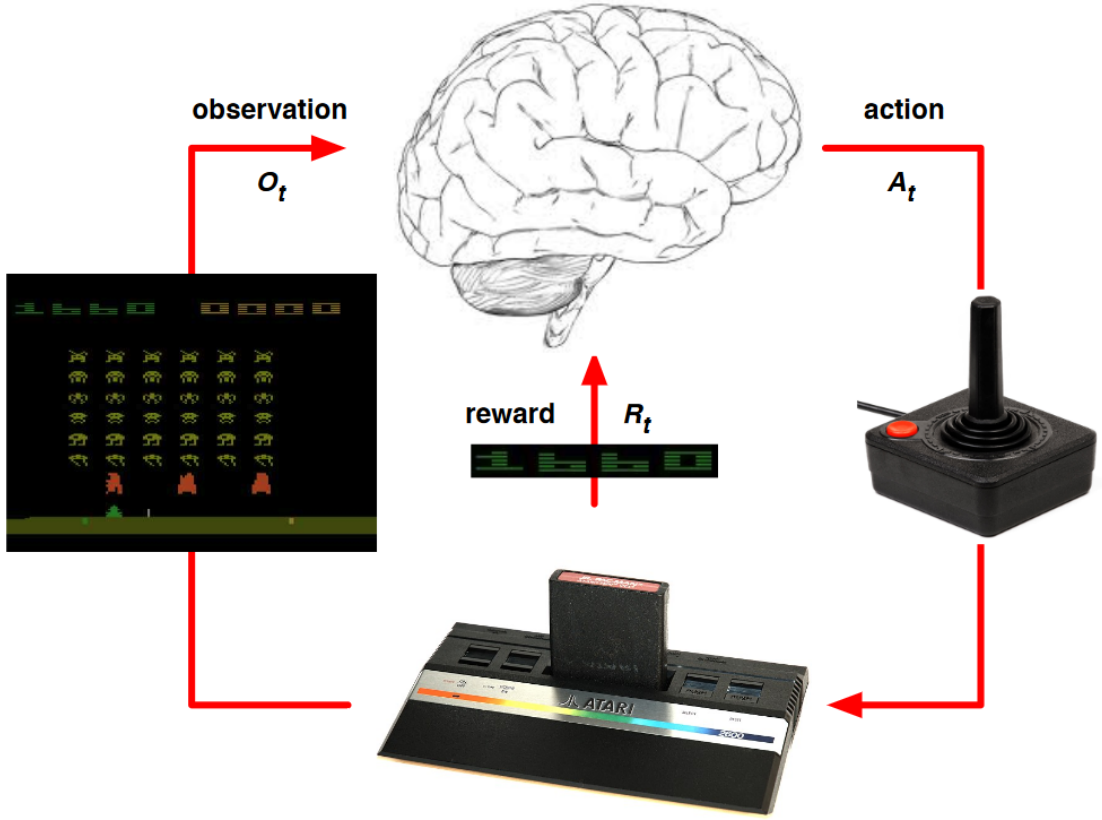
- Planlama:
  - Çevre modeli mevcut, mükemmel olmayabilir
  - Davranış sergilemeden model ile gelecek durumlar hesaplanır
  - Daha sonra en iyi duruma götüren davranışlar uygulanır

- (Pekiştirmeli) Öğrenme:
  - Geçiş ve/veya ödül modeli bilinmiyor
  - Etmen çevre ile etkileşime girmek zorunda
  - Etkileşimlerden hangi davranışın hangi durumda iyi olduğunu bulmak

- Not: Beraber de kullanılabilirler!

Atari Örneği

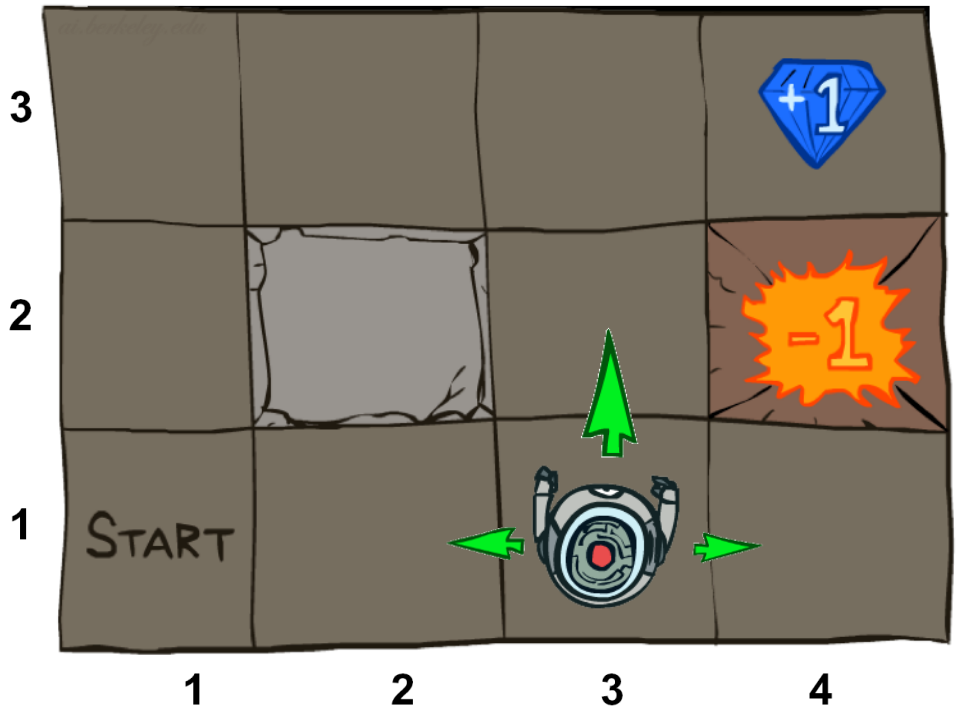




David Silver'ın notlarından

Grid Dünyası

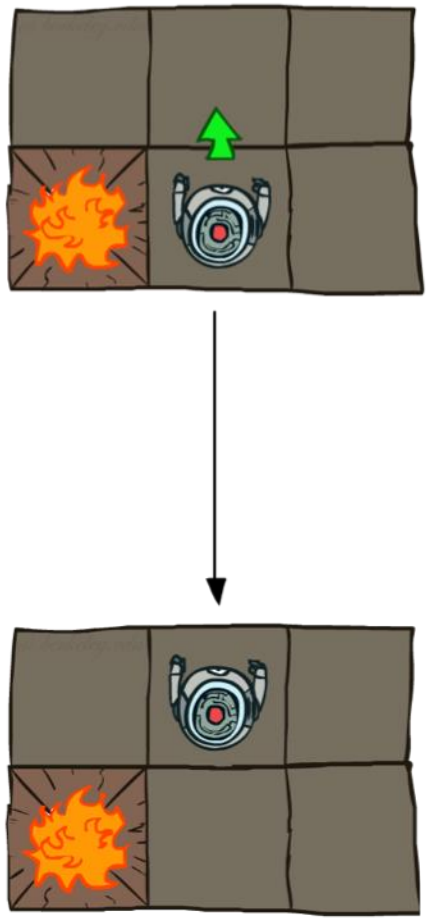
- Durum: x,y
- Davranış: Dört yön (rastgelelik)
- Ödül
  - Ufak yaşam ödülü
  - Son durum ödülleri
- Amaç: Toplam ödülü arttırmak



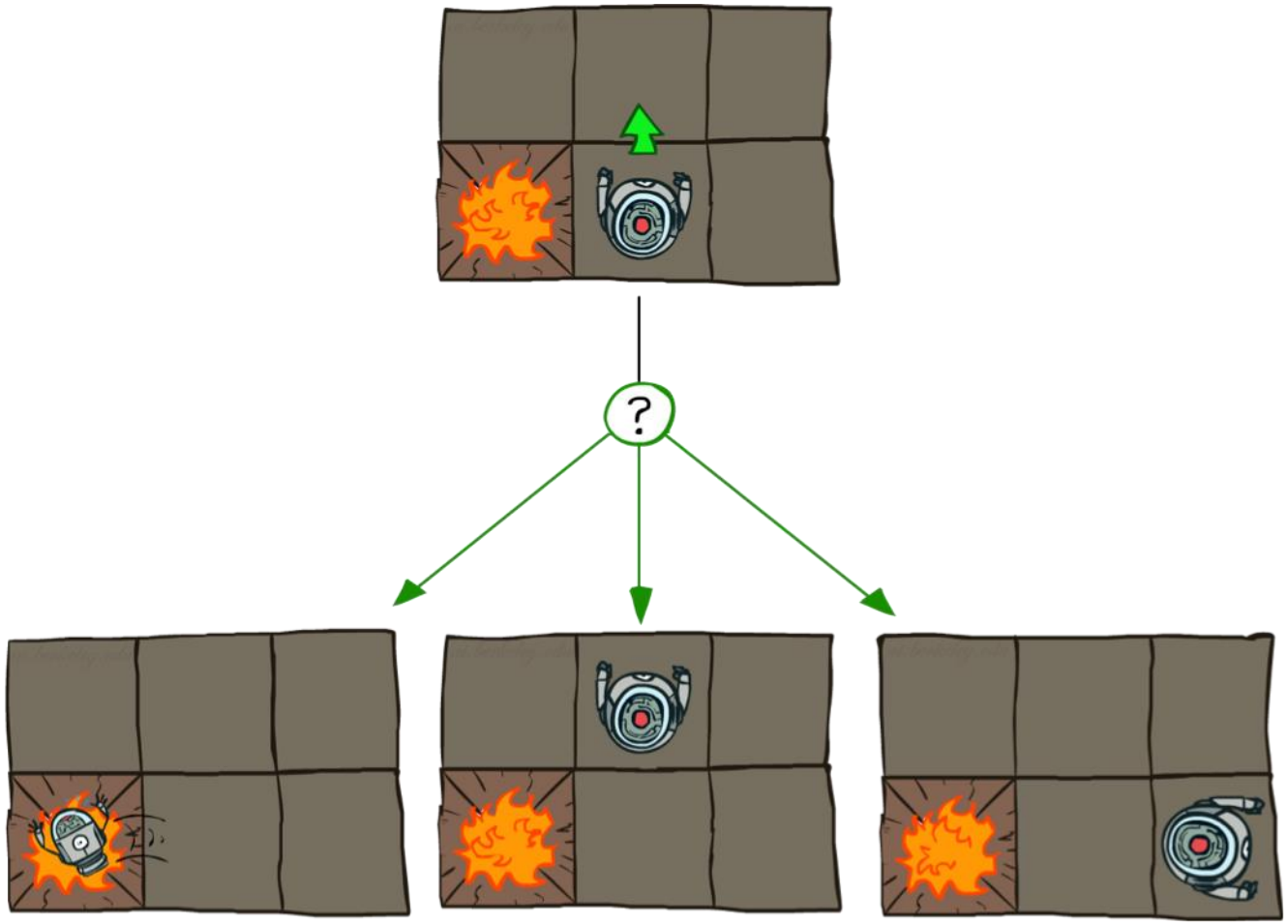
Figürler Berkeley cs188 ders notlarından

Grid Dünyası Davranışları

Deterministik

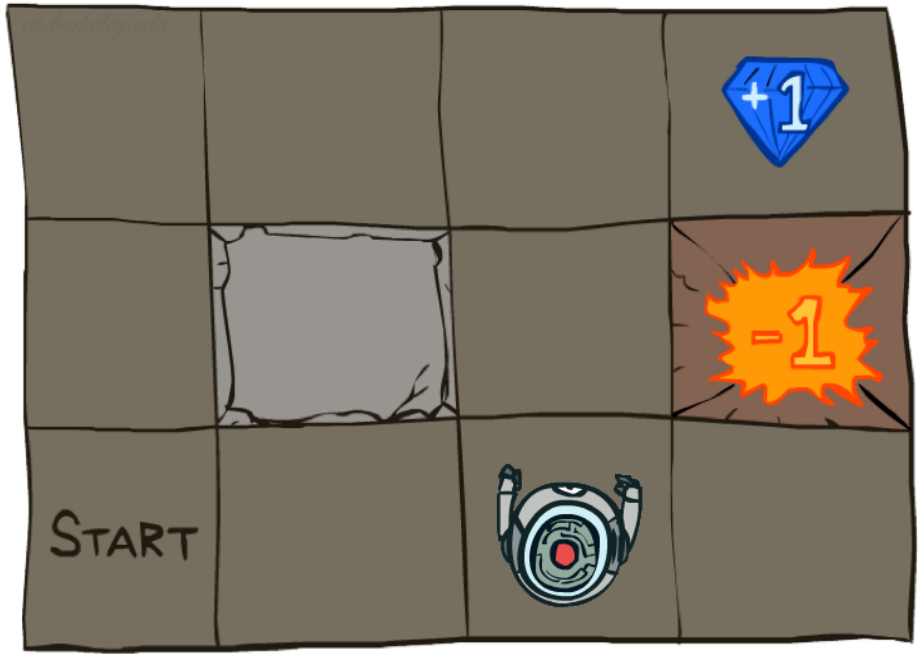


Stokastik



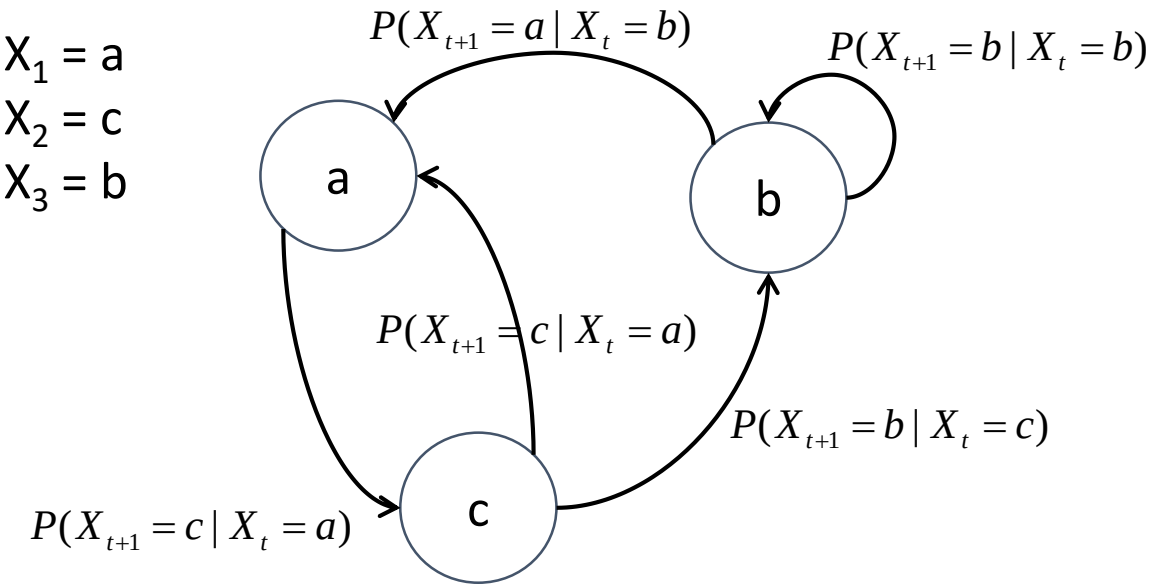
Öğrenme mi Planlama mı?

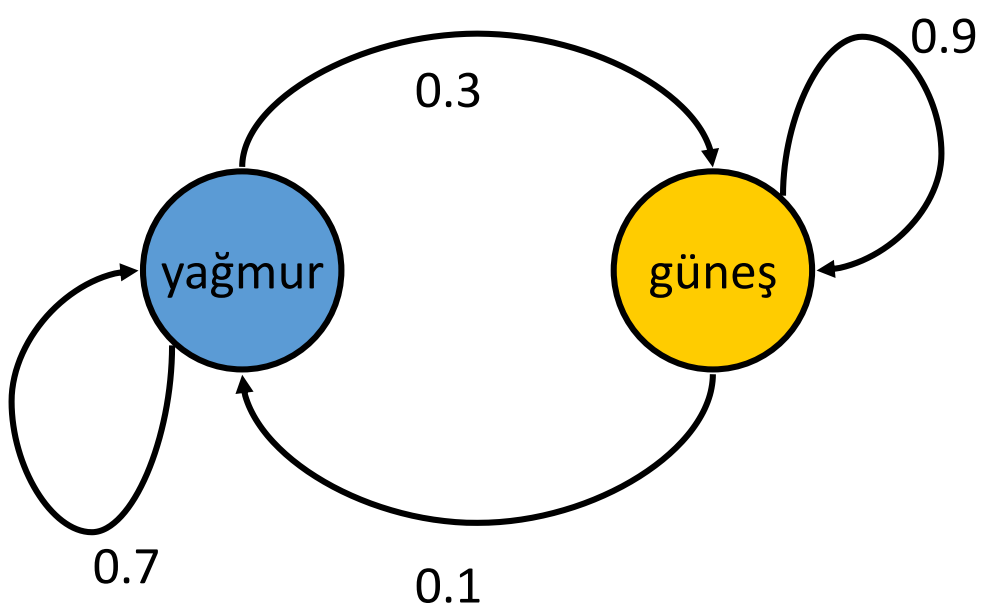
- Planlamanın dezavantajları?
- Planla ama bilgi tut
- Matematiksek olarak bu problemi Markov Karar Süreçleri ile anlatabiliriz



Markov Süreçleri

X_1 = a
X_2 = c
X_3 = b





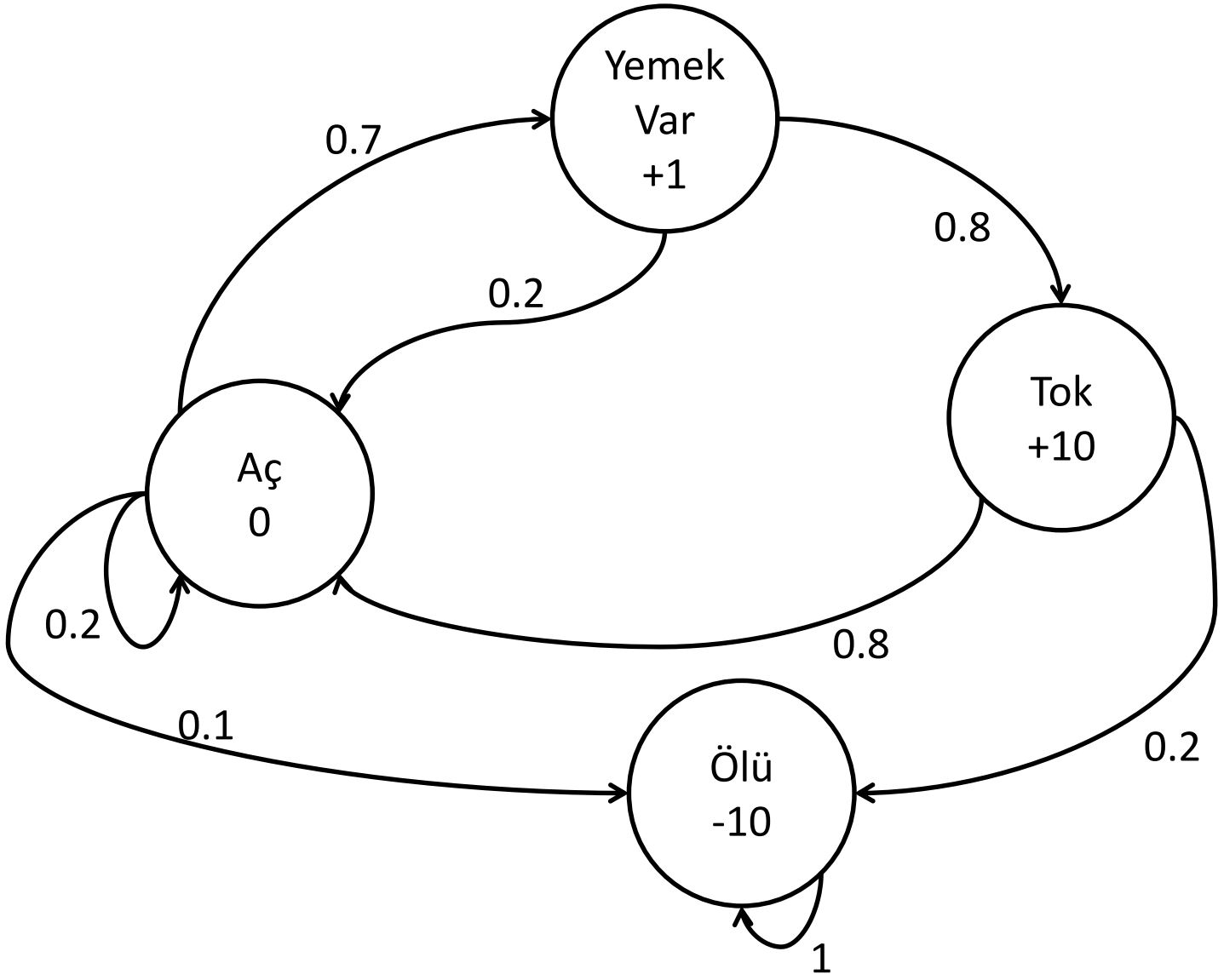
Geçiş modeli

$$A_{ij} = P(X_{t+1} = s_j \mid X_t = s_i)$$

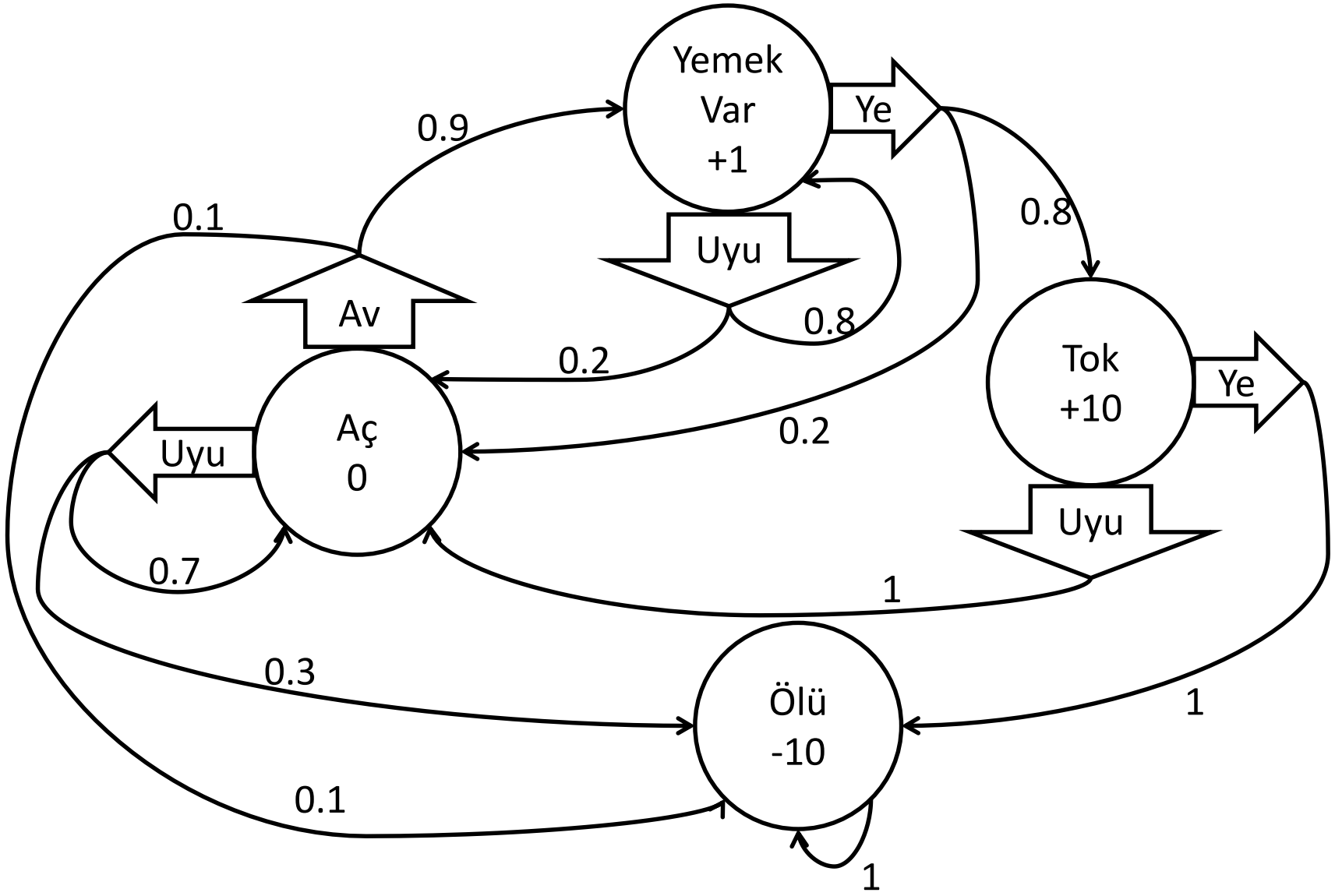
- Sınırlı sayıda durum
- Geçişler rastgele
- Sonraki durum, sadece şimdiki duruma bağlı

t \ t+1	s_1	s_2	s_3
s_1	$P(X_{t+1} = s_1 \mid X_t = s_1)$	$P(X_{t+1} = s_2 \mid X_t = s_1)$	$P(X_{t+1} = s_3 \mid X_t = s_1)$
s_2	$P(X_{t+1} = s_1 \mid X_t = s_2)$	$P(X_{t+1} = s_2 \mid X_t = s_2)$	$P(X_{t+1} = s_3 \mid X_t = s_2)$
s_3	$P(X_{t+1} = s_1 \mid X_t = s_3)$	$P(X_{t+1} = s_2 \mid X_t = s_3)$	$P(X_{t+1} = s_3 \mid X_t = s_3)$

Mağara Adamı – Markov Ödül Süreci

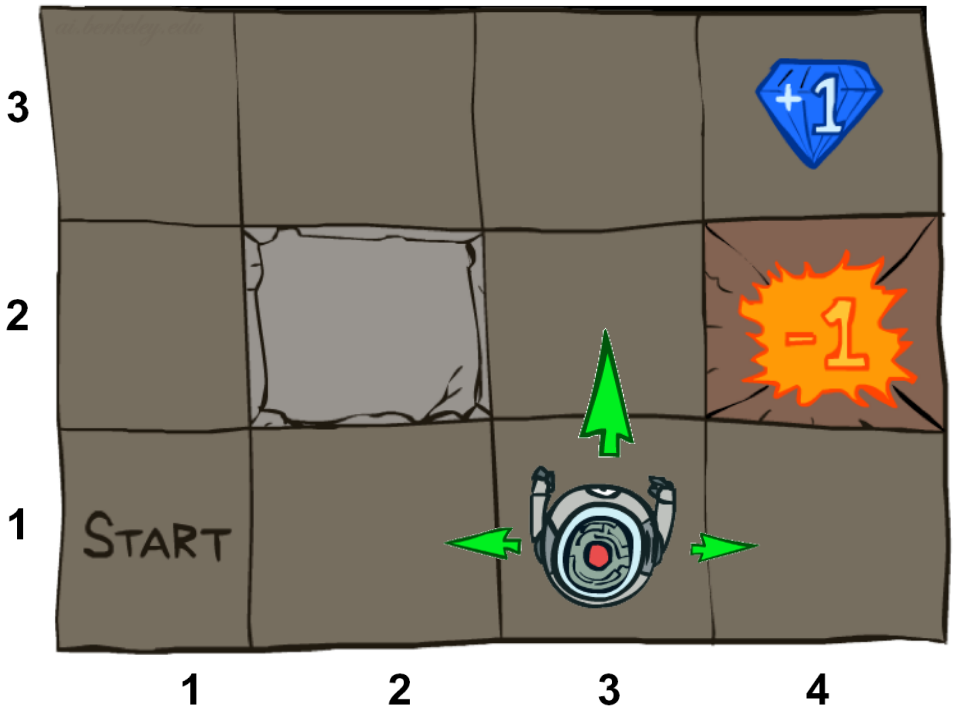


Mağara Adamı – Markov Karar Süreci



Markov Karar Süreçleri

- Durum kümesi, $s \in S$
- Davranış kümesi, $a \in A$
- Geçiş Modeli $T(s, a, s')$, $P(s' \mid s, a)$
- Ödül Fonksiyonu, $R(s, a)$
- Azaltma katsayısı, γ (?!?!?)
- Başlangıç durumu
- Bazen son durumlar



- MKS: $\{S, A, T, R, \gamma\}$
- Amaç: Toplam ödülü eniyileyecek şekilde durumları davranışlara eşleyen bir politika (π) öğrenmek

Politikalar

- Durumları davranışlara eşleştirir

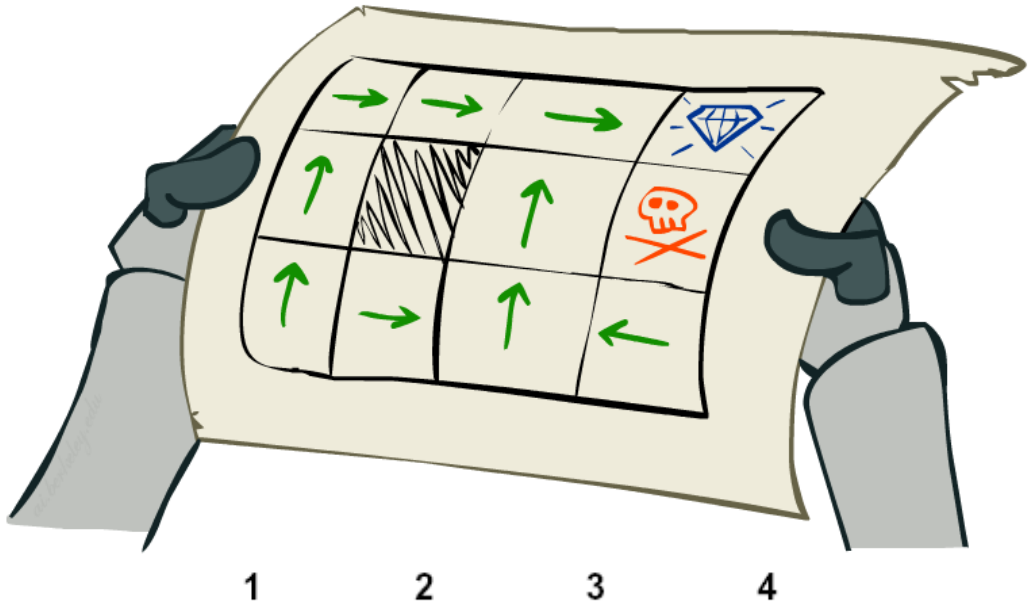
$$\pi: S \rightarrow A$$

- Deterministik ya da stokastik olabilir

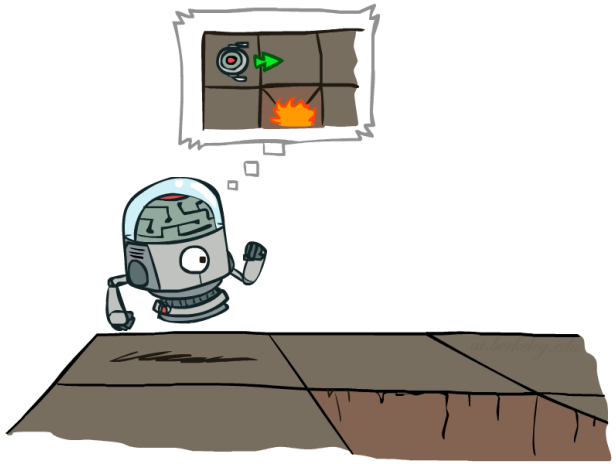
$$\pi(s) = a$$
$$\pi(a|s) = P(a|s)$$

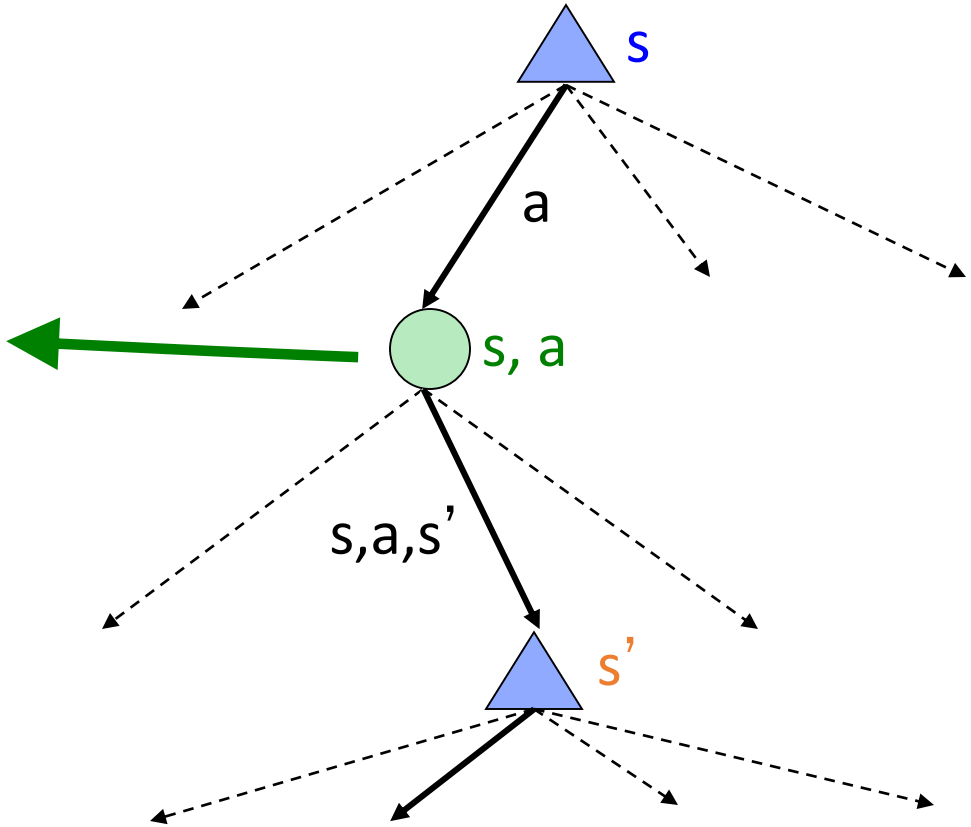
- En iyi politika π^* ile gösterilir

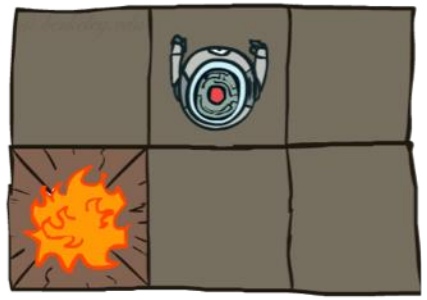
- En iyi politika nasıl hesaplanır?



MKS: Davranış ve Geçişler

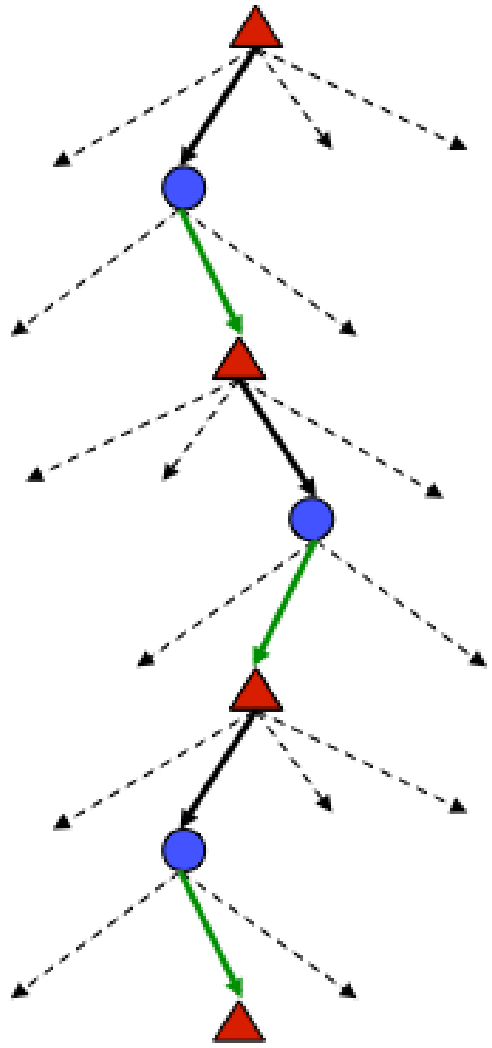






Çözüm Ufku

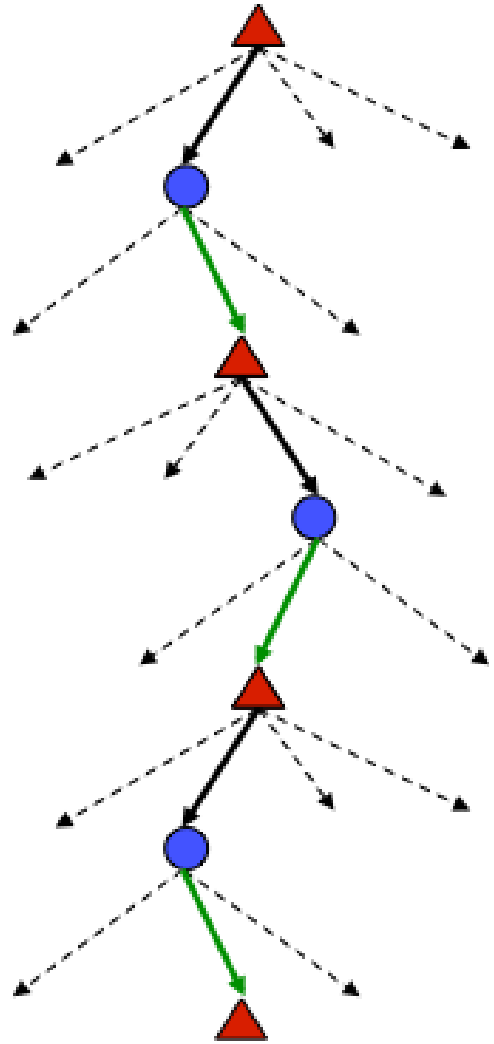
- Sonlu:
  - Etmen'in problemi çözmek için sınırlı sayıda adımı var
  - En iyi davranış hem duruma hem de ne kadar zaman kaldığına bağlı
- Sonsuz:
  - Zaman kısıtlaması yok (bitiş durumları varsa sonlu da olabilir)
  - En iyi davranış sadece duruma bağlı



Durum Serilerinin Toplam Faydası

- Politika takibi sonunda bir durum serisi elde edilir
- Her durumda bir ödül alınıyor
- Peki bir durum serisinin faydası ne?
  - Toplam ödül
  - Ortalama ödül
  - Zamanlamaya göre ile ağırlıklı ödül
  - Sonsuz seriler?

- Fikir: Azaltılmış değerlerin toplamı



Azaltmak

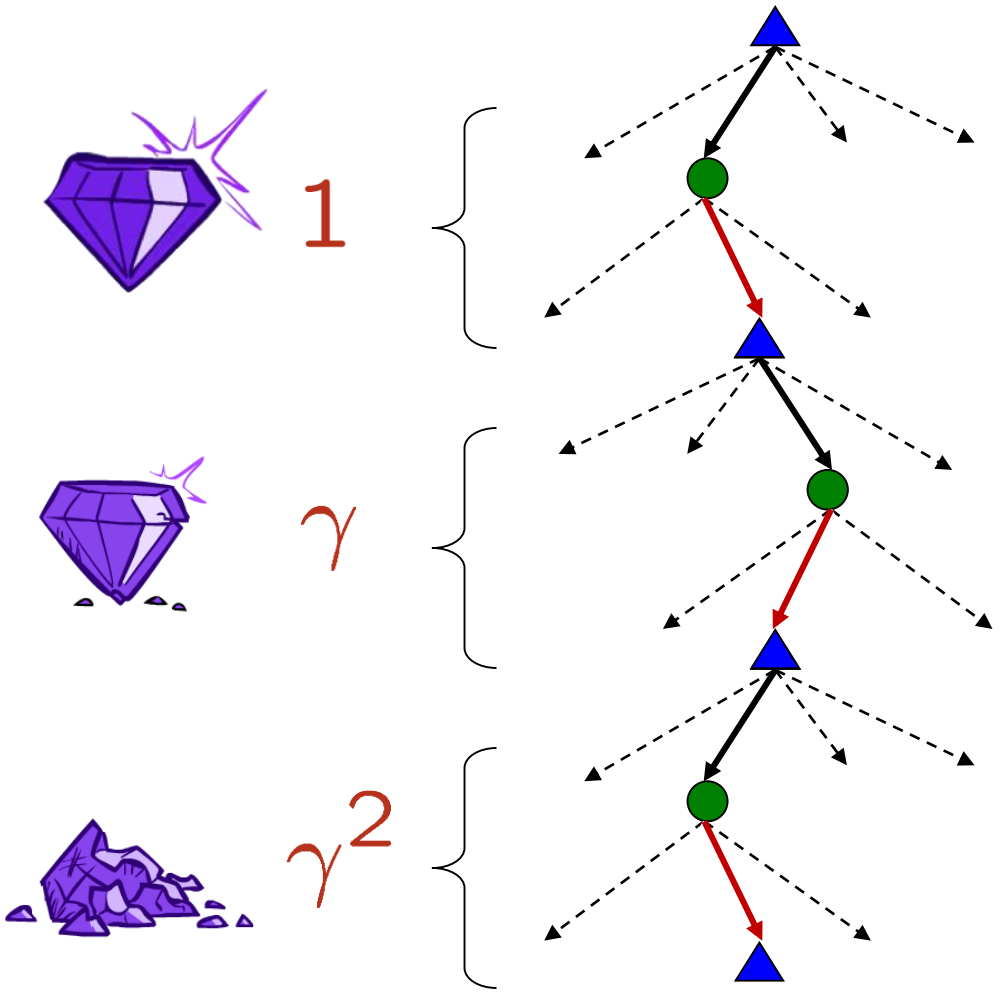
- Nasıl azaltalım?
  - Her adımda azaltma katsayısı ile çarpmak
- Neden azaltalım?
  - Yakın ödüller uzak ödüllerden daha önemli
  - Matematiksel nedenler
- Örnek: Azaltma katsayısı 0.5 ile

$U([1,2,3]) = 1 \cdot 1 + 0.5 \cdot 2 + 0.25 \cdot 3$

$U([1,2,3]) < U([3,2,1])$

$$U([r_0, \ldots r_\infty]) = \sum_{t=0}^{\infty} \gamma^t r_t \leq R_{\max}/(1-\gamma)$$

$\gamma \epsilon [0,1]$: azaltma katsayısı



Durum Değerleri

- Durum-ödül serisi

$$\sigma = \{sr_1, sr_2, sr_3, \dots\}$$

- Serinin faydası (getirisi)

$$G(\sigma) = r_1 + \gamma r_2 + \gamma^2 r_3 + \cdots = r_1 + \gamma G(\sigma'), \sigma' = \{sr_2, sr_3, \dots\}$$

- Verilen bir politika, π, için MKSnin durum-**değer** fonksiyonu, $V^\pi(s)$: s durumundan itibaren π politikasını takip etmenin faydası. ($G_t = r_{t+1} + \gamma r_{t+2} + \cdots$)

$$V^\pi(s) = E_\pi[G_t|S_t = s] = E_\pi[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|S_t = s]$$

- Verilen bir politika, π, için MKSnin davranış-**değer** fonksiyonu,$Q^\pi(s,a)$: s durumunda a davranışını uygulayıp π politikasını takip etmenin faydası.

$$Q^\pi(s,a) = E_\pi[G_t|S_t = s, A_t = a] = E_\pi[\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|S_t = s\,, A_t = a]$$

Değer Fonksiyonları

- Deterministik politika:

$$V^{\pi}(s) = Q^{\pi}(s, \pi(s)) \text{ ☺}$$

- Stokastik politika:

$$V^{\pi}(s) = \sum_{a \in \mathrm{A}} \pi(a|s) Q^{\pi}(s, a)$$

- Öbür yön

$$Q^{\pi}(s, a) = R(s, a) + \gamma \sum_{s' \in S} P(s'|s, a)\, V^{\pi}(s')$$

Bellman Denklemleri

- Durum-Değer Fonksiyonu

$$V^{\pi}(s) = Q^{\pi}\big(s, \pi(s)\big) = R\big(s, \pi(s)\big) + \gamma \sum_{s' \in S} P(s'|s, \pi(s))\, V^{\pi}(s')$$

- Davranış-Değer Fonksiyonu

$$Q^{\pi}(s, a) = R(s, a) + \gamma \sum_{s' \in S} P(s'|s, a)\, V^{\pi}(s')$$

En İyi Değerler

- Durum-Değer Fonksiyonu

$$V^*(s) = \max_{\pi}(V^{\pi}(s)) = R\big(s, \pi^*(s)\big) + \gamma \sum_{s'} P(s'|s, \pi^*(s))V^*(s')$$

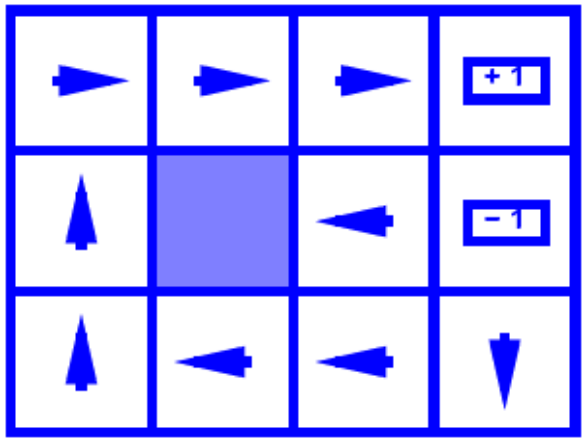
- Davranış-Değer Fonksiyonu

$$Q^*(s,a) = \max_{\pi}(Q^{\pi}(s,a)) = R(s,a) + \gamma \sum_{s'} P(s'|s,a)V^*(s')$$

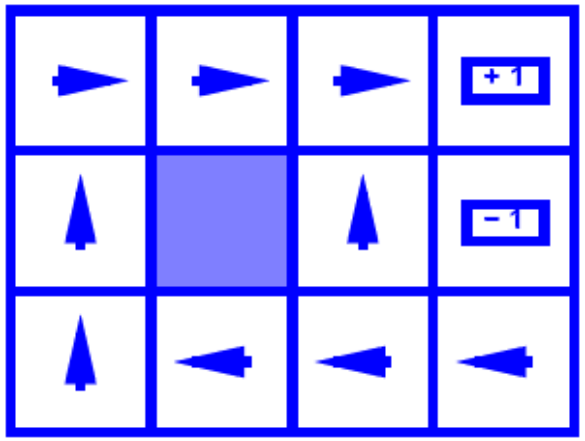
- Politika

$$\pi^*(s) = arg\max_{a} Q^*(s,a)$$

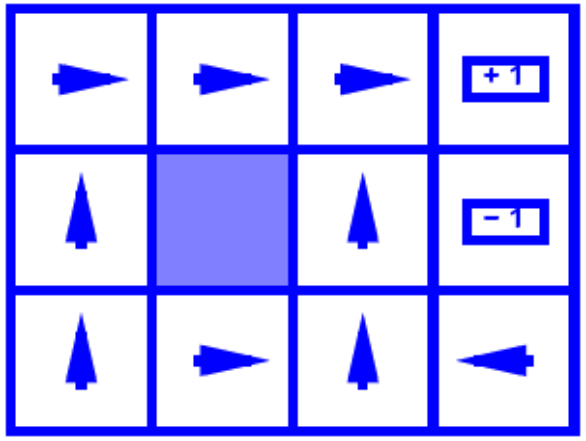
Grid Dünyası



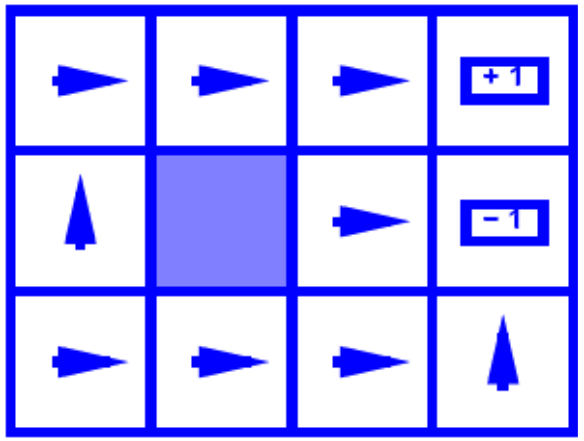
R(s) = -0.01



R(s) = -0.03



R(s) = -0.4



R(s) = -2.0

En İyi Politika Nasıl Hesaplanır?

- Değerlerden politikaya gidilebilir. Fikir: Değer fonksiyonlarını hesapla
- Sıfır değerler ile başla $V_0(s) = 0$
- Bellman Güncellemeleri:

$$V_{t+1}(s) = \max_a \left(R(s,a) + \gamma \sum_{s'} \left(P(s'|s,a) V_t(s') \right) \right)$$

- Değerler arasındaki fark bir sınıra inene kadar her durum için tekrarla
- Buna Değer İterasyonu diyoruz, Davranış-Değer fonksiyonları için de uygulanabilir

En İyi Politika Nasıl Hesaplanır?

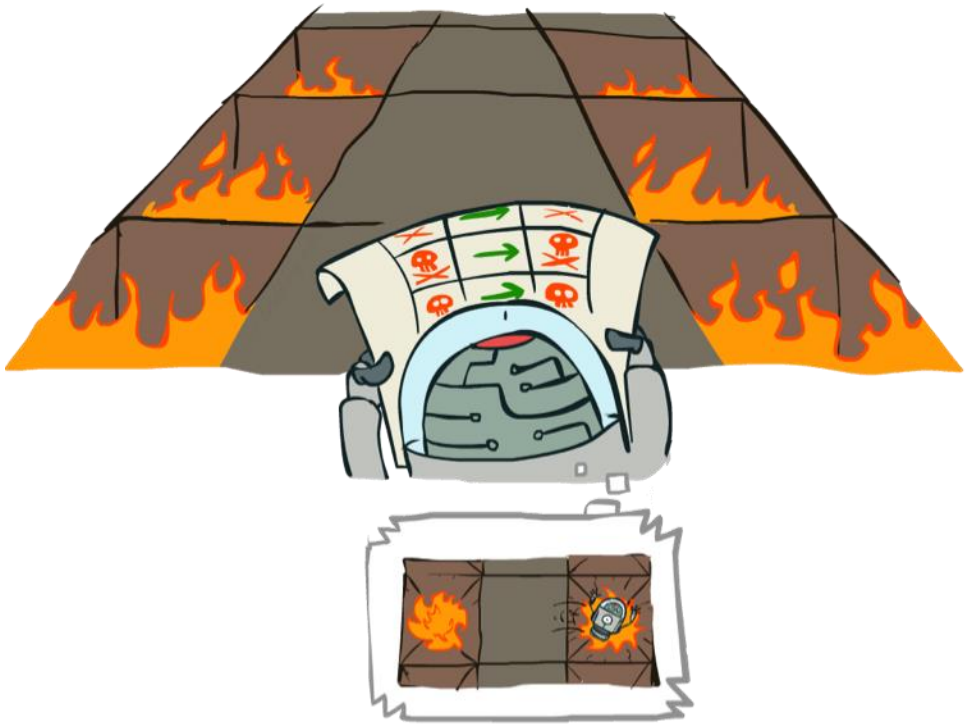
- Başka Fikir: Verilen bir politikanın (π) değer-fonksiyonunu $V^{\pi}(s)$ hesapla
- Sonra yeni politikayı bu değer fonksiyonundan çıkart

$$V^{\pi}(s) = R(s, \pi(s)\,) + \gamma \sum_{s'} P(s'|s, \pi(s)) V^{\pi}(s'))$$

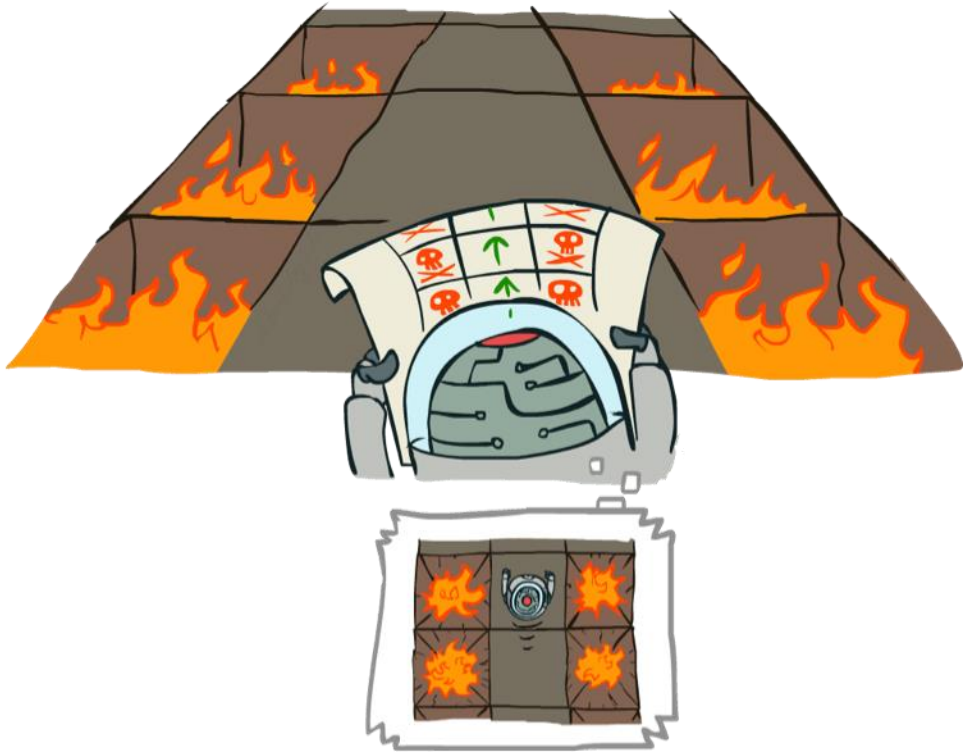
$$\pi_{t+1}(s) = \arg\max_{a} \left(R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^{\pi_t}(s') \right)$$

Politika Değerlendirmesi

Hep Sağa



Hep Öne



Politika Değerlendirme Örneği

Hep Sağa

Hep Öne

En İyi Politika Nasıl Hesaplanır?

- Başka Fikir: Verilen bir politikanın (π) değer-fonksiyonunu $V^{\pi}(s)$ hesapla
- Sonra yeni politikayı bu değer fonksiyonundan çıkart

$$V^{\pi}(s) = R(s, \pi(s)\,) + \gamma \sum_{s'} P(s'|s, \pi(s)) V^{\pi}(s'))$$

$$\pi_{t+1}(s) = \arg\max_{a} \left(R(s,a) + \gamma \sum_{s'} P(s'|s,a) V^{\pi_t}(s') \right)$$

- Politika İterasyonu:
  - Rastgele bir politikadan başla
  - Politika için değer-fonksiyonu hesapla
  - Yeni politika hesapla. Değişmez ise dur

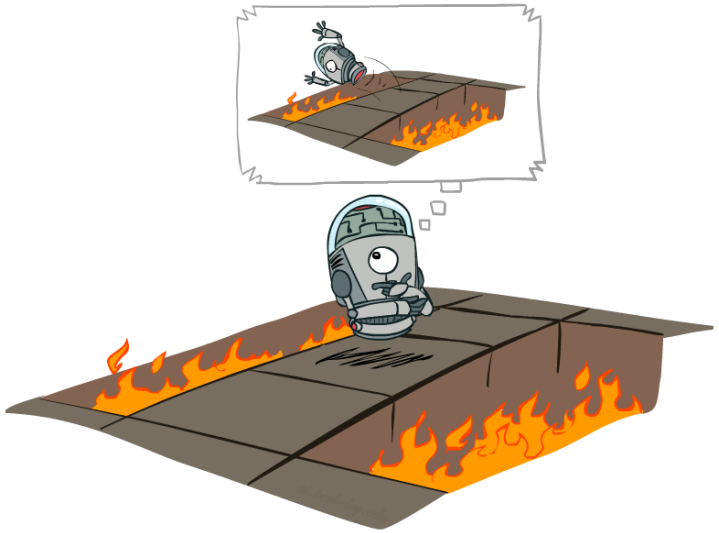
Şimdiye Kadar...

- Pekiştirmeli Öğrenme Konseptleri
  - Durum Davranış, Etmen, Çevre, Politika …
- Bellman denklemleri ve en iyi politikayı hesaplama
- Sonlu ve ayrık durum ve davranış uzayları varsayımı
- Model ve maliyet fonksiyonun bilinmesi

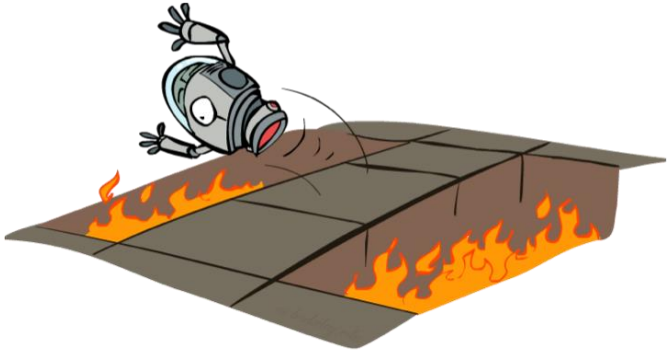
- Peki ya gerçek dünya?

Pekiştirmeli Öğrenme

- Yine problemi MKS olarak modelliyoruz ve amacımız iyi bir politika hesaplamak
- Bu sefer modeli ve/veya ödül fonksiyonuna erişimimiz yok



- Etmen gerçekten davranışlarını uygulamak zorunda!



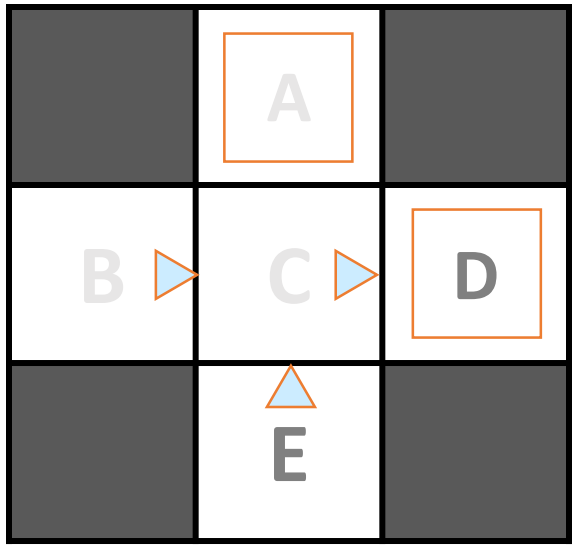
Model Tabanlı Yaklaşım

- Etmen, davranışlarını deneyerek tecrübe toplar
- Tecrübelerine göre modelleri öğrenir (gözetimli öğrenme) – Empirik MKS
- Değer ya da Politika İterasyonu ile politika hesaplar

- Artılar - Eksiler?

Örnek

Politika π



γ = 1

Gözlemler

Deneme 1

B, east, C, -1
C, east, D, -1
D, exit, x, +10

Deneme 2

B, east, C, -1
C, east, D, -1
D, exit, x, +10

Deneme 3

E, north, C, -1
C, east, D, -1
D, exit, x, +10

Deneme 4

E, north, C, -1
C, east, A, -1
A, exit, x, -10

Öğrenilmiş Modeller

$\hat{T}(s, a, s')$

T(B, east, C) = 1.00
T(C, east, D) = 0.75
T(C, east, A) = 0.25
…

$\hat{R}(s, a, s')$

R(B, east, C) = -1
R(C, east, D) = -1
R(D, exit, x) = +10
…

Modelsiz Yaklaşım

- Model olmadan politika öğrenebilir miyiz?
- Fikir: Modelsiz değer-fonksiyonlarını hesaplamak ve buradan politikayı bulmak

- Basitleştirilmiş fikir – Pasif PÖ: Verilen sabit politikanın π değerlerini modeller olmadan hesaplamak (politika değerlendirmesi)
  - Doğrudan hesaplama
  - Örneklem tabanlı hesaplama

En İyi Politika Nasıl Hesaplanır?

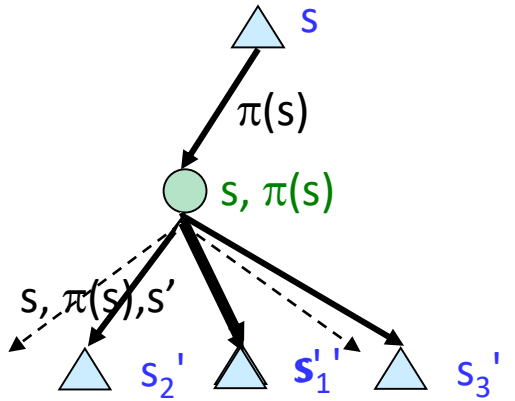
- Verilen bir MKS'den en iyi politikayı bulmak
- Bir fikir: En iyi değerleri bul (politika sonrasında kolay)

$$V_{t+1}^{\pi}(s) = \left(R(s,a) + \gamma \sum_{s'} \left(P(s'|s,a) V_t^{\pi}(s') \right) \right)$$

- Bulunulan durumda davranışları dene ve ortalama al

$$\text{örneklem}_i = R\left(s, \pi(s_i')\right) + \gamma V_t^{\pi}(s_i')$$

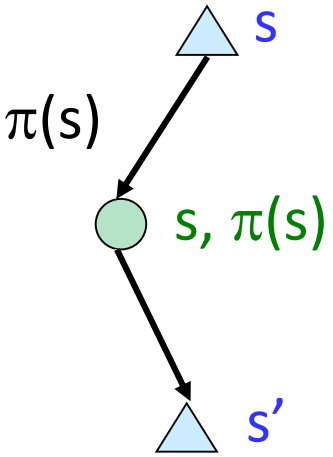
$$V_{t+1}^{\pi} = \frac{1}{n} \sum_i \text{örneklem}_i$$



Ne gibi sorunlar olabilir?

Zamansal Fark (Temporal Difference) Learning

- Her tecrübeden öğren
  - V(s)'yi her geçişte (s, a, s', r) güncelle
  - Daha olası sonraki durumlar, s', daha etkili olsun
- Zamansal Fark Öğrenimi
  - Sabit politika varsayımı devam
  - Hareketli Ortalama Alınarak:



V(s) Örneklemi: $\ddot{o}rneklem = R\big(s, \pi(s')\big) + \gamma V^{\pi}(s')$

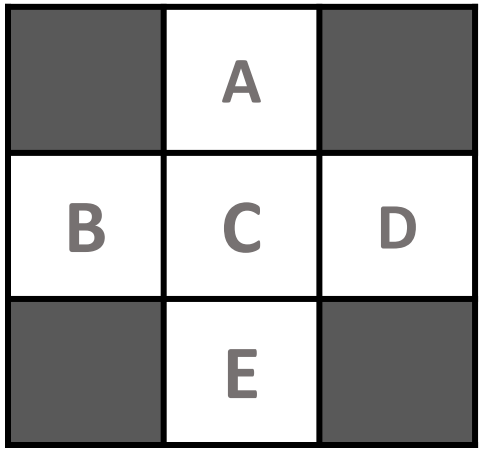
V(s) Güncellemesi: $V^{\pi}(s) \leftarrow (1-\alpha)V^{\pi}(s) + \alpha \cdot \ddot{o}rneklem$

Başka Şekilde Yazımı: $V^{\pi}(s) \leftarrow V^{\pi}(s) + \alpha(\ddot{o}rneklem - V^{\pi}(s))$

- Sabit α ile her zaman yakınsamayabilir, her adımda küçültmek gerekli

Örnek

Durumlar

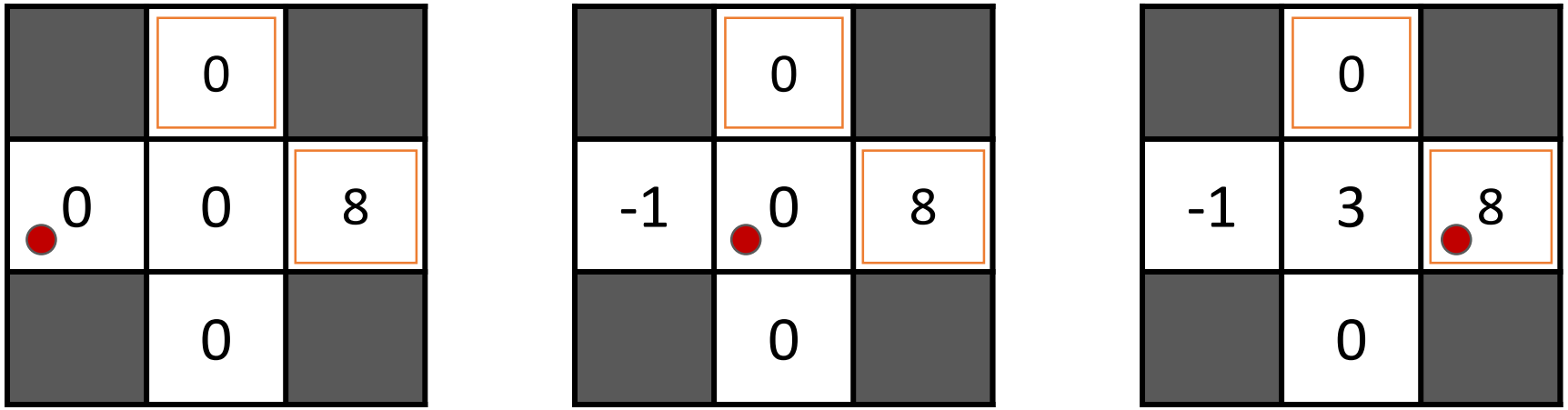


$\gamma = 1, \alpha = 1/2$

Gözlemlenen Geçişler

B, east, C, -2

C, east, D, -2



$$V^{\pi}(s) \leftarrow V^{\pi}(s) + \alpha(R\left(s, \pi(s')\right) + \gamma V^{\pi}(s') - V^{\pi}(s))$$

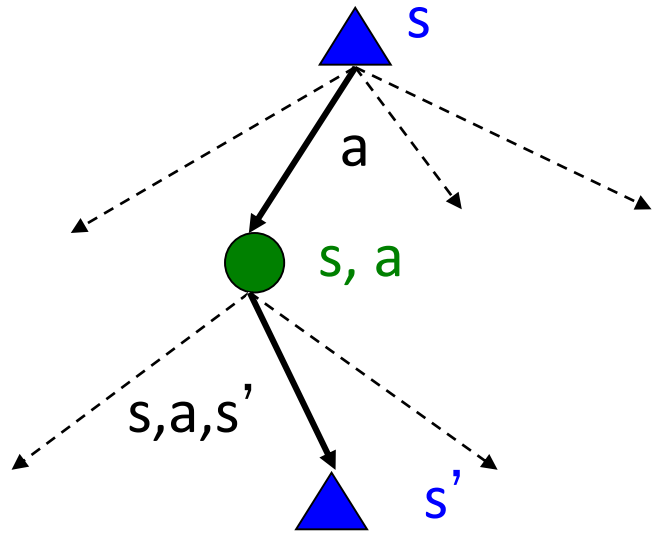
Politika Güncellemeleri?

- Bellman güncelleme denklemlerine benzer hareketli ortalama hesaplayan bir modelsiz politika değerlendirme yöntemi
- Peki yeni politika hesaplamak istersek?

$$\pi^*(s) = arg\max_a Q^*(s,a)$$

$$Q^\pi(s,a) = R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a)\, V^\pi(s')$$

- Durum-değerleri yerine davranış değerleri öğrenmek!



Q-Öğrenimi

- Örneklem tabanlı Davranış-Değeri İterasyonu

$$Q_{k+1}(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_k(s',a')$$

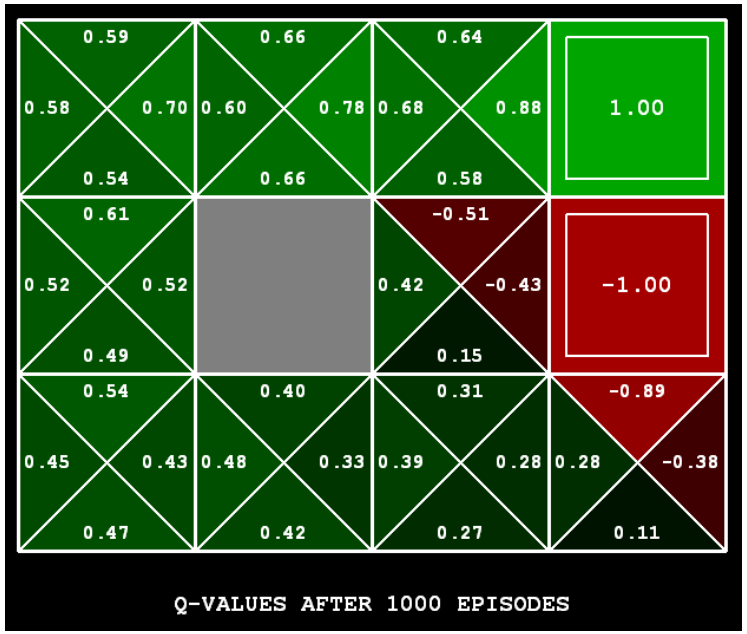
- Bir önceki ile aynı fikir Q(s,a)
  - Yeni geçiş gözlemle (s,a,s',r)
  - Eski değer: $Q(s,a)$
  - Yeni tahmin:

$$\ddot{o}rneklem = R(s,a,s') + \gamma \max_{a'}(Q(s',a'))$$

  - Değeri günceller

$$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \alpha(\ddot{o}rneklem)$$

$$Q(s,a) \leftarrow Q(s,a) + \alpha\big(\ddot{o}rneklem - Q(s,a)\big)$$



Q-Öğrenimi Özellikleri

- En iyi politika takip edilmeden bile en iyi değerler öğrenebiliyor!
- Buna politika dışı öğrenme diyoruz

- Dikkat edilmesi gerekenler:
  - Yeterince keşif yapılması (her durum ve davranış yeterince tekrarlanmalı)
  - Öğrenme katsayısı doğru azaltılmalı

Aktif Pekiştirmeli Öğrenme

- Şimdiye kadar sabit bir politika takip ettik
- Modelsiz olarak politikayı nasıl öğrenebiliriz?
- Bu durumda
  - Etmen davranış kararları almalı
  - Temel ödünleşme: keşif ve kullanma (exploration - exploitation)

- Örnek: ε-açgözlü davranış seçimi

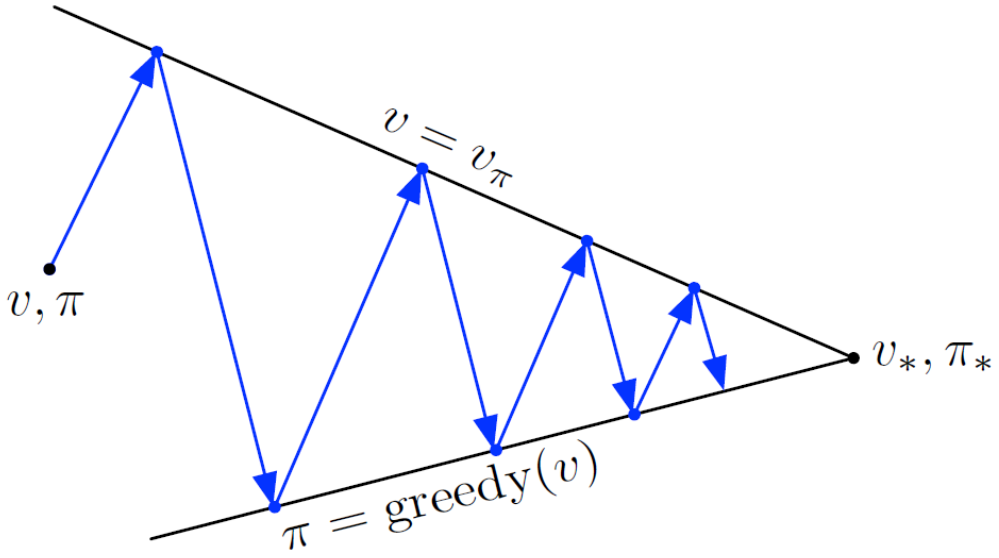
$$\pi(a|s) = \begin{cases} 1 - \epsilon + \frac{\epsilon}{m} & \text{if } \mathrm{a} = \underset{a}{\operatorname{argmax}}\big(\mathrm{Q}(s,a)\big) \\ \frac{\epsilon}{m} & \text{otherwise} \end{cases}$$

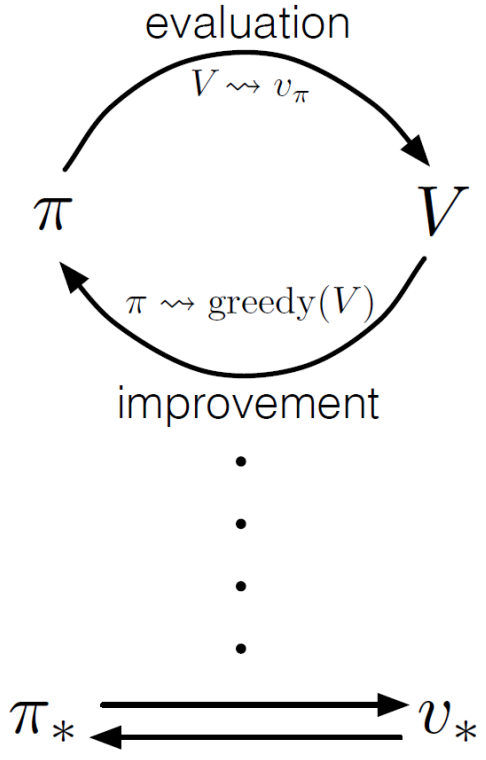
Q-Öğrenimi:

- Mevcut davranış değerlerinden ε-açgözlü bir davranış seç
- Davranış-değer fonksiyonunu güncelle

Genel Politika İterasyonu

- Politika değişmeyene kadar tekrarla:
  - Politikayı değerlendir
  - Politikayı geliştir





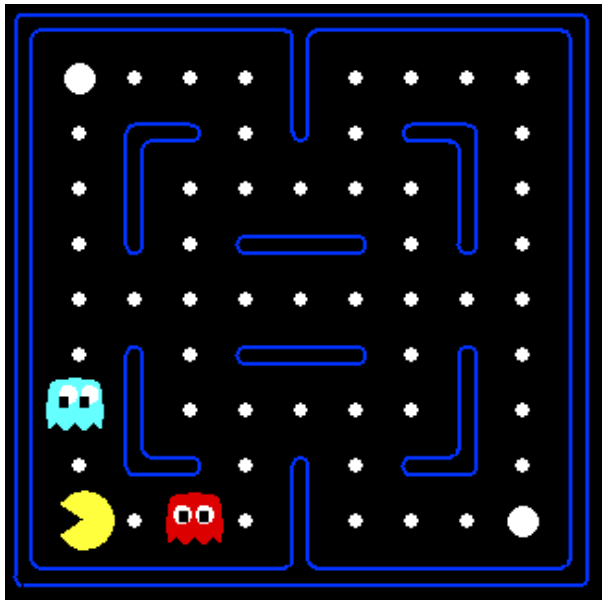
Şimdiye kadar gördüğümüz MKS'ler için bu bizi en iyi politikaya götürür!

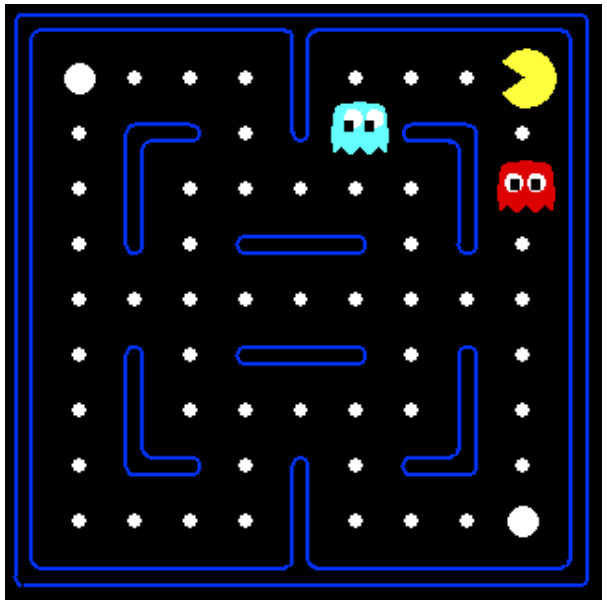
Başka Çevreler?

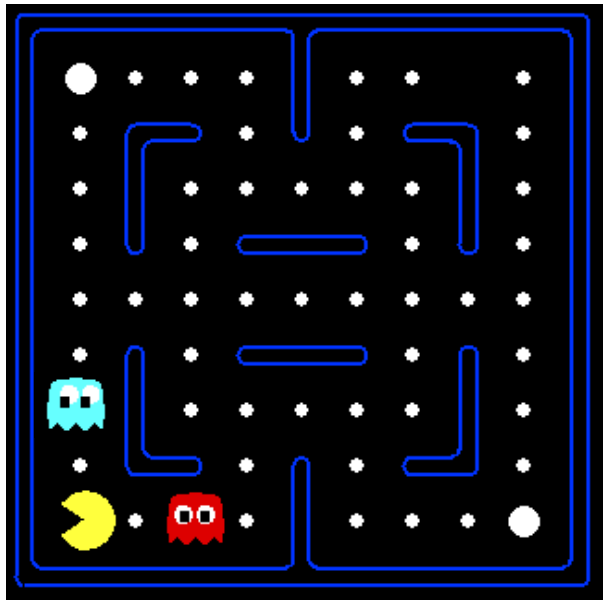
- Durum ve davranış uzaylarını biraz düşünelim:
  - Satranç
  - Otonom araç
- Günümüz problemlerinde:
  - Durum uzayları sürekli ve/veya çok büyük
  - Davranış uzayları sürekli olabilir

- Şimdiye kadar kullandığımız yöntemler *tablo* tabanlı, ne yapabiliriz?
- Fonksiyon Yaklaşıklamalı (Function Approximation) Pekiştirmeli Öğrenme!

Örnek: Pacman





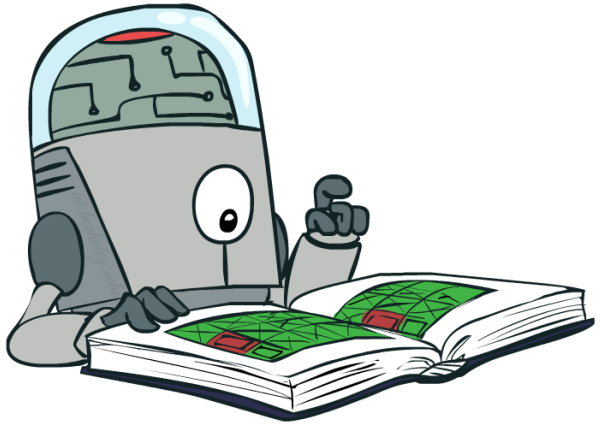


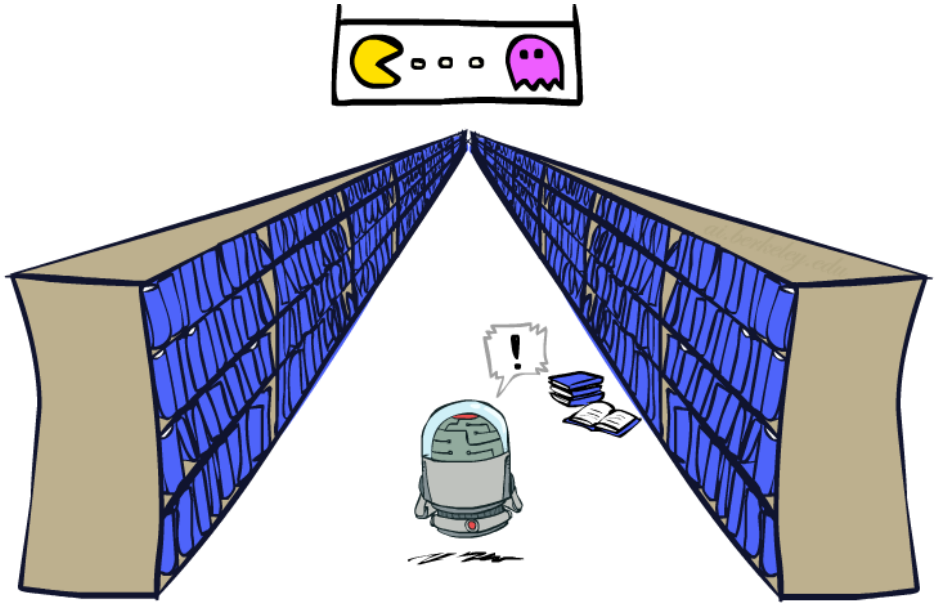
Durumlar Arası Genellemek

- Q-Öğrenimi bütün durum-davranış ikililerinin içeren bir tablo tutuyor

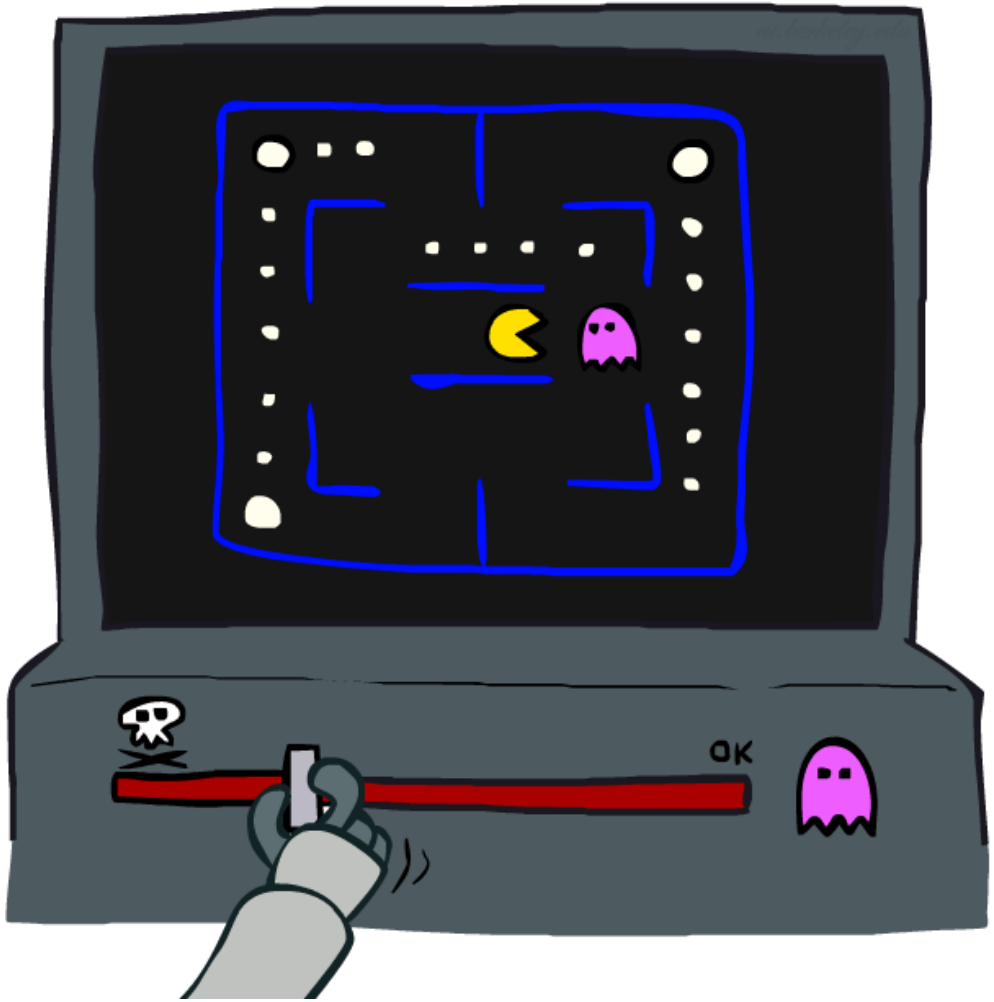
- Her durum-davranış ikilisi için öğrenmek gerçekçi değil
  - Bütün hepsine uğramak için zaman yok
  - Bütün hepsini tutmak için hafıza yok

- Genellemek istiyoruz
  - Daha ufak bir uzayda öğrenmek
  - Bilgileri genellemek



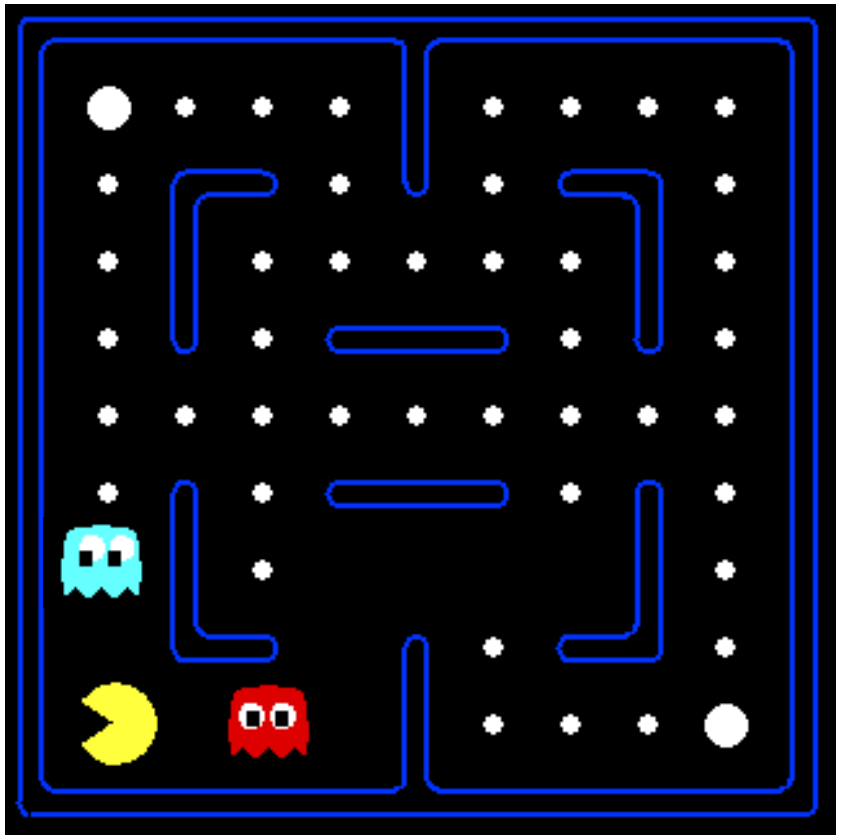


Yaklaşık Q-Öğrenimi



Öznitelik Tabanlı Yaklaşım

- Durumda bütün bilgiler yerine, sadece önemlileri tutmak
- Genel olarak gerçek sayılar kullanmak
- Pacman için ne öznitelikler olabilir?



Lineer Değer Fonksiyonları

- Öznitelikleri kullanarak bütün durumlar için fonksiyonları hesaplamak mümkün. Birkaç ağırlık değişkeni işimizi görüyor

$$V(s) = w_1 f_1(s) + w_2 f_2(s) + \ldots + w_n f_n(s)$$

$$Q(s,a) = w_1 f_1(s,a) + w_2 f_2(s,a) + \ldots + w_n f_n(s,a)$$

- İyi yanı: Birkaç sayı ile tecrübeler özetleniyor
- Kötü yanı: Öznitelikleri benzer durumlar gerçekte çok farklı olabilir (öznitelikler doğru seçilmeli)

Yaklaşık Q-Öğrenimi

$$Q(s,a) = w_1 f_1(s,a) + w_2 f_2(s,a) + \ldots + w_n f_n(s,a)$$

- Lineer Denklemler ile:

$$\text{transition} = (s, a, r, s')$$

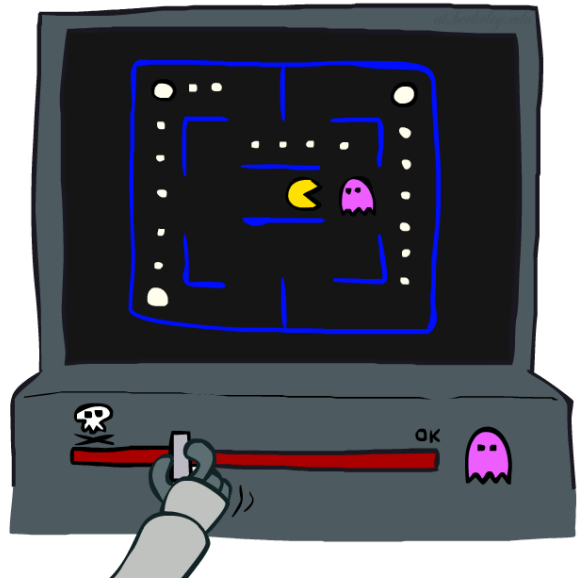
$$\text{difference} = \left[r + \gamma \max_{a'} Q(s', a')\right] - Q(s,a)$$

$$Q(s,a) \leftarrow Q(s,a) + \alpha\,[\text{difference}]$$

Tam değerler

$$w_i \leftarrow w_i + \alpha\,[\text{difference}]\, f_i(s,a)$$

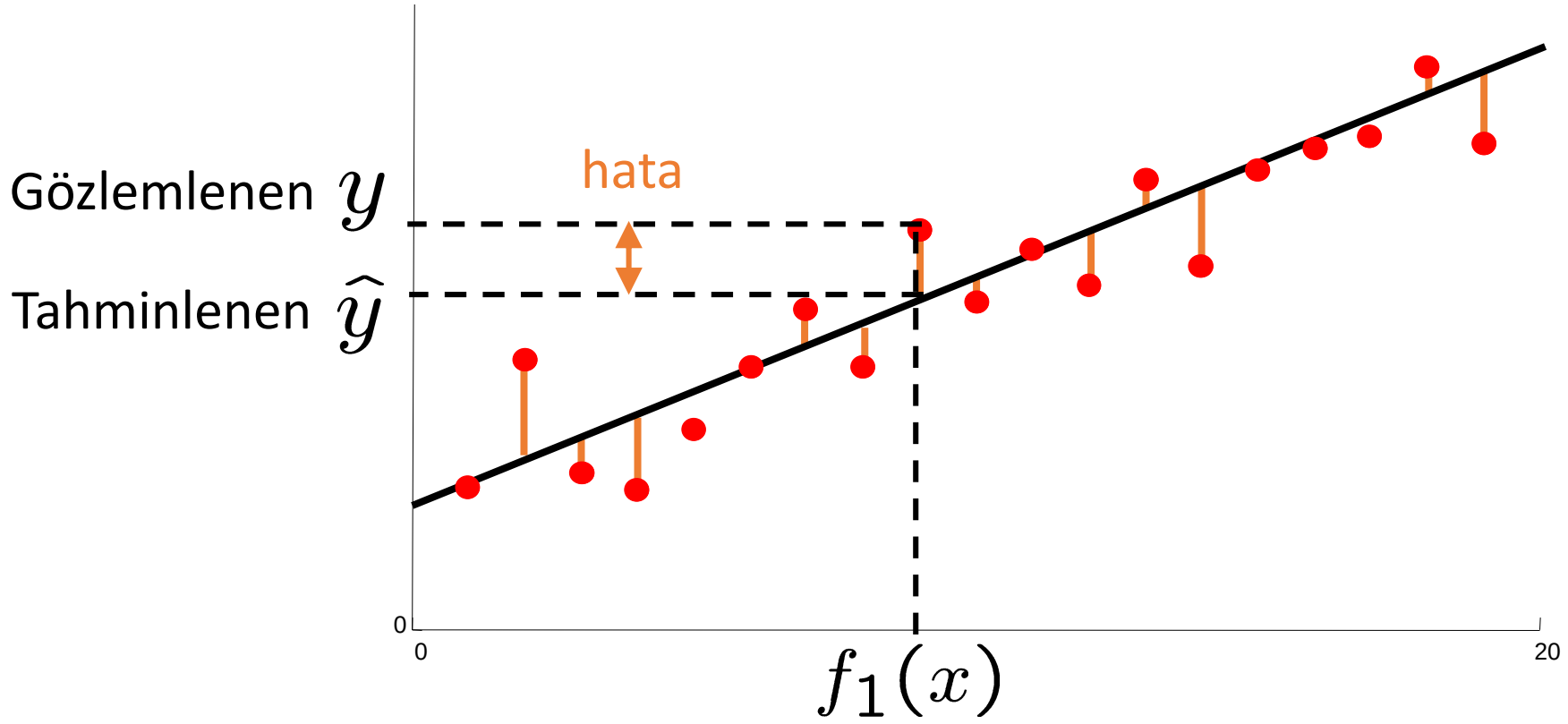
Yaklaşık değerler



- Çevrim-içi en-küçük kareler yöntemiyle eş-değer!

En-Küçük Kareler En İyilemesi

$$\text{total error} = \sum_i (y_i - \hat{y}_i)^2 = \sum_i \left(y_i - \sum_k w_k f_k(x_i) \right)^2$$



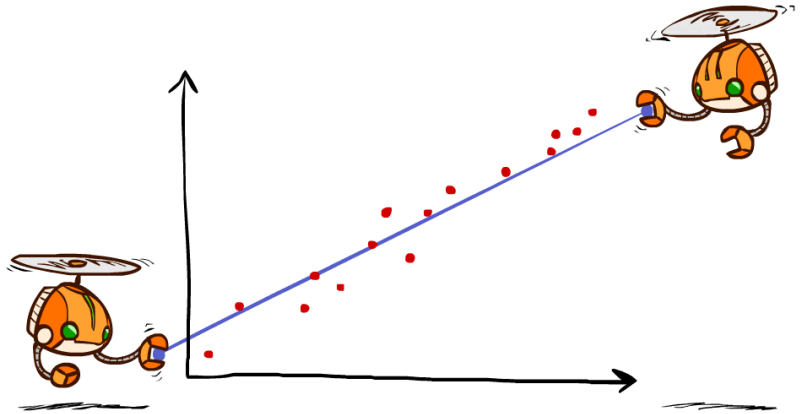
Hatayı Küçültmek

Türev al, eğimi takip et ☺

$$\text{error}(w) = \frac{1}{2}\left(y - \sum_k w_k f_k(x)\right)^2$$

$$\frac{\partial\ \text{error}(w)}{\partial w_m} = -\left(y - \sum_k w_k f_k(x)\right) f_m(x)$$

$$w_m \leftarrow w_m + \alpha\left(y - \sum_k w_k f_k(x)\right) f_m(x)$$



Yaklaşık Q-Öğrenimine Uygulanması

$$w_m \leftarrow w_m + \alpha\left[r + \gamma \max_a Q(s', a') - Q(s, a)\right] f_m(s, a)$$

Daha neler var?

- Politika gradyanları:
  - Politikayı fonksiyon yaklaşıklamalı olarak al: Girdi olarak durum, çıktı olarak davranış
  - Toplam değeri arttıracak şekilde öğren
  - Durumlar ve davranışlar bir metrik uzayda ise Doğal Politika Gradyanları
- Derin pekiştirmeli öğrenme:
  - Değer ve/veya politika için derin ağlar kullan
- Tersine pekiştirmeli öğrenme:
  - Gösterimlerden maliyet fonksiyonunu öğren
  - Mevcut bir yaklaşım ile pekiştirmeli öğrenme yap
- Ayrık zaman yerine sürekli zaman kullanmak
- Değerleri öğren, bu değerler ile plan yap (örn: TD-Gammon, AlphaGo)
- …