



**YTU**

YILDIZ TEKNİK ÜNİVERSİTESİ  
Fen Edebiyat Fakültesi

# YAPAY ZEKAYA GİRİŞ

6. Hafta

# Genel Bilgiler

## □ Ders içeriği:

HAFTALIK KONULAR VE İLGİLİ ÖN HAZIRLIK ÇALIŞMALARI		
HAFTALAR	KONULAR	Ön Hazırlık
1	Yapay Zeka Nedir: Tarihçesi, Gelişimi, Türleri, ve Geleceği?	Ders Kitabı 4, Bölüm 1
2	Yapay Zeka Alanındaki Son Gelişmeler ve Uygulamaları.	Ders Kitabı 3
3	Veri Nedir? Veri Ön İşleme, Veri Manipülasyonu ve Performans Değerlendirme Metrikleri.	Ders Kitabı 5, Bölüm 5
4	Optimizasyon Nedir? Tek-Çok Amaçlı Problemler ve Zorlukları, Yapay Zeka Tabanlı Arama Algoritmalarına Giriş.	Ders Kitabı 6
5	Optimizasyon Algoritmalarına Giriş ve Uygulamalar.	Ders Kitabı 6
6	Genetik Algoritmalara Giriş.	Ders Kitabı 3, Bölüm 4
7	Yapay Sinir Ağlarına Giriş.	Ders Kitabı 3, Bölüm 5
8	<b>Arasınav</b>	
9	Yapay Zeka Tabanlı Problemlerin Modellenmesi: Sınıflandırma Algoritmaları.	Ders Kitabı 2, Bölüm 2 Ders Kitabı 3, Bölüm 1-2-3
10	Yapay Zeka Tabanlı Problemlerin Modellenmesi: Regresyon Algoritmaları.	Ders Kitabı 2, Ders Kitabı 3, Bölüm 6
11	Yapay Zeka Tabanlı Problemlerin Modellenmesi: Kümeleme Algoritmaları.	Ders Kitabı 5, Bölüm 9
12	Derin Öğrenmeye Giriş.	Ders Kitabı 5, Bölüm 14
13	Doğal Dil İşlemeye Giriş.	Ders Kitabı 4
14	Yapay Zeka Genel Uygulamalar I	Ders Kitabı 2
15	Yapay Zeka Genel Uygulamalar II	Ders Kitabı 3
16	<b>Final Sınavı</b>	

# Genetik Algoritma

---

## ❑ Genetik Algoritma

- Genetik Algoritma (Genetic Algorithm - GA): Doğadaki canlıların geçirdiği süreci örnek almaktadır. İyi nesillerin kendi yaşamlarını korurken, kötü nesillerin yok edilmesi ilkesine dayanır.
- Matematiksel modellemenin yapılamadığı ya da kesin çözümün olmadığı problemlerde genetik algorithmadan yararlanılabilir.

# Genetik Algoritma

---

## ❑ Genetik Algoritma

- GA, anne ve baba bireyden (bir önceki nesill) doğan yeni bireylerin koşullara uyum sağlayıp yaşamlarını devam ettirmesine dayanır. Yeni bireyler, anne ve babasından gelen iyi genleri bünyelerinde koruyabilecekleri gibi kötü genleri de almış olabilirler. Bu durumda kötü genleri almış olan bireyler varlıklarını sürdüremeyeceklerdir.

# Genetik Algoritma

---

## ❑ Genetik Algoritma

- GA, uygunluk (fitness) işlevi, yeni çözümler üretmek için çaprazlama ve değiştirme gibi operatörleri kullanır. GA'nın önemli özelliklerinden biri de bir grup üzerinde çözüm araması ve bu sayede çok sayıda çözümün içinde en iyiyi seçmesidir.
- GA evrimsel yaklaşım ilkeleri ışığında rastlantısal arama tekniklerini kullanarak çözüm bulmaya çalışan, parametre kodlama temeline dayanan sezgisel bir arama tekniğidir.

# Genetik Algoritma

---

## ❑ Genetik Algoritma

- Yapay Zeka alanında kullanılan Genetik Algoritma, bir tür en iyi noktayı arayan algoritmadır. Bir probleme çözüm aramakla ilgilenir.
- Gerçek hayatta karşılaşılan problemlerin (kuru yük gemisine konteynerler nasıl yerleştirilmeli, bir noktadan başka bir noktaya nasıl gidilebilir ya da en uygun teslimat rotası oluşturma gibi),
- Söz konusu problem bir arama problemine dönüştürülebilirse, bu problemi Genetik Algoritma ile çözebilir.



# Genetik Algoritma

---

## ❑ Genetik Algoritma

- Genetik Algoritma (GA), permütasyon tabanlı bir optimizasyon yapar ve olasılıklar üzerinden yakınsama kriterleri altında arama yapan bir fonksiyondur. Doğada gözlemlenen evrimsel sürece benzer bir şekilde çalışan, arama ve eniyileme yöntemidir.
  - Genetik Algoritma, biyolojik evrimin temel prensiplerinden ilham alan güçlü bir evrimsel stratejidir. Araştırmacı öncelikle değişken tipini ve ele aldığı problemi doğru tanımlamalı ve kodlamasını bu tanımlamaya göre yapmalıdır. Ardından algoritmanın girdilerinden olan uygunluk fonksiyonu (fitness) tanımlanır ve optimize edilmesi gereken amaç fonksiyonu bu fonksiyondur. Geçiş ve Mutasyon gibi genetik operatörler, evrim sürecinin birçok aşamasında stokastik olarak uygulanmaktadır, bu yüzden gerçekleşme olasılıkları belirlenmelidir. Son olarak yakınsama kriterleri sağlanmalıdır ve optimal maliyet ile problem çözülmelidir. GA, problemin çözülebilmesi için lokal değil, global bir araştırma yapar. Problemi etkileyen çok fazla etken varsa, çözümde Genetik Algoritma kullanılması literatürce önerilmektedir..
-

# Genetik Algoritma

---

## □ Tanımlar

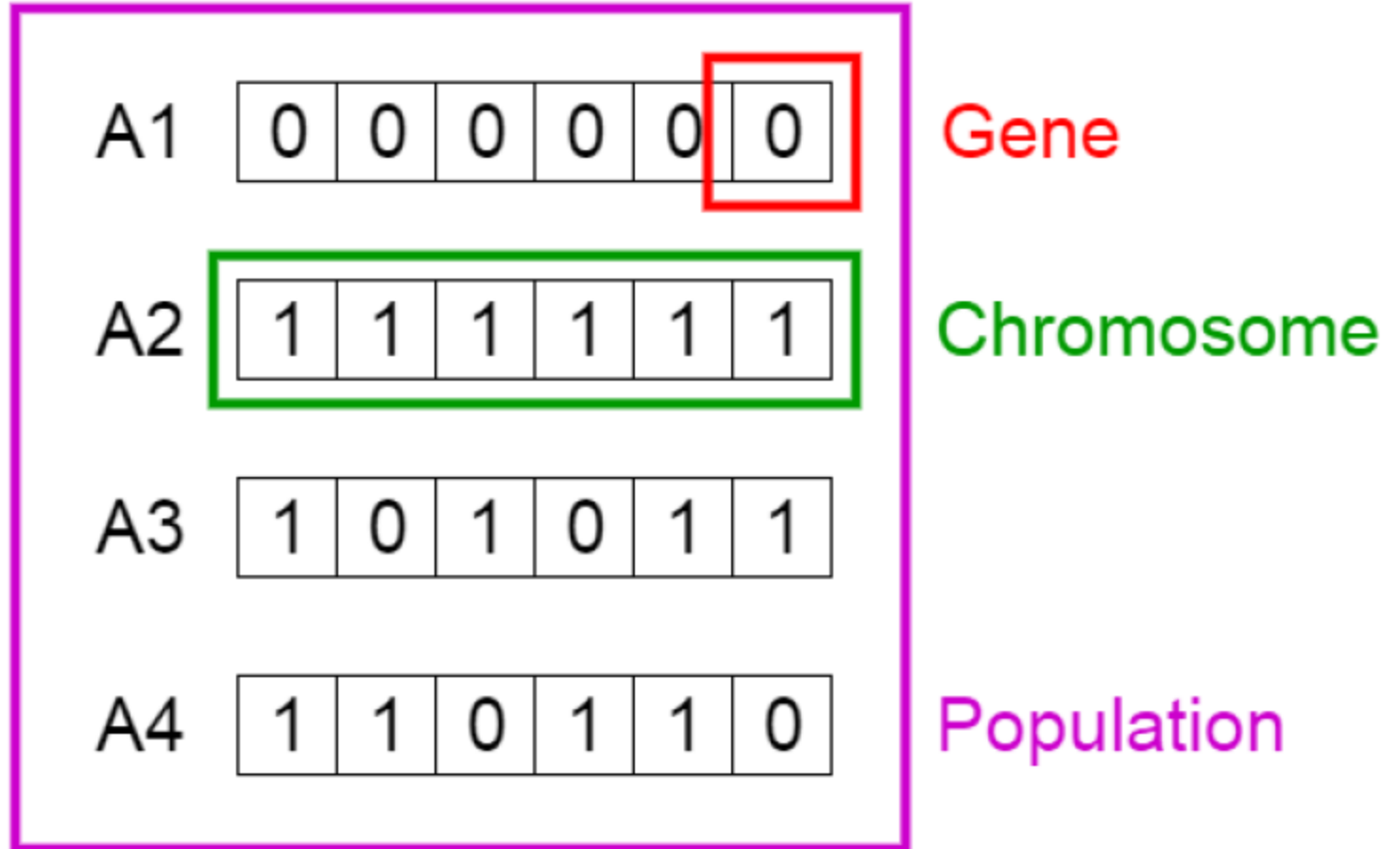
- ***Kromozom:*** Bir bireyin tam ifadesidir. Yöneylem Araştırması problemlerine göre amaç fonksiyonunda yer alan değişkenlerin her birinin alacağı değerler kromozomlarda temsil edilir. Bir çocuğun kromozomu ebeveynlerinden gelen özellikleri içerir.
- ***Gen:*** Bir kromozom içerisindeki tek bir özelliktir. Dikkat edilmesi gereken husus, bir çocuğun kromozomu genlerden oluşur ve bu genler rastlantı olarak ebeveynlerden gelmektedir.
- ***Popülasyon:*** Genetik algoritmalar problemlerin çözümü için evrimsel süreci bilgisayar ortamında taklit ederler. Problem için olası pek çok çözümü temsil eden bir çözüm kümesi genetik algoritma terminolojisinde nüfus veya popülasyon adını alır. Popülasyonlar vektör, kromozom veya birey adı verilen sayı dizilerinden oluşur.



# Genetik Algoritma

## ❑ Tanımlar

- *Kromozom*
- *Gen*
- *Popülasyon*



# Genetik Algoritma

---

## ❑ Tanımlar

- **Nitelik:** Bir genin alabileceği değerdir. Daha ziyade nitel değişkenlerde ortaya çıkmaktadır. Örneğin saç renginin gende temsil edildiği durumlarda bu genin nitelik'leri kırmızı, sarı, siyah ve kahverengi gibi olabilecektir.

Bireyler	Nitelik (Değer)	Kodlanmış Hali
C1	77	0100 1101
C2	96	0110 0000
C3	19	0001 0011
C4	08	0000 1000
C5	56	0011 1000
C6	195	1100 0011
C7	0	0000 0000
C8	255	1111 1111

## □ Tanımlar

- 
- ```
graph TD; 1((1)) --> 2((2)); 1((1)) --> 3((3)); 1((1)) --> 4((4)); 2((2)) --> 3((3)); 3((3)) --> 4((4)); 4((4)) --> 5((5)); 5((5)) --> 6((6)); 6((6)) --> 10((10)); 6((6)) --> 12((12)); 7((7)) --> 8((8)); 8((8)) --> 9((9)); 9((9)) --> 11((11)); 10((10)) --> 11((11)); 10((10)) --> 12((12)); 11((11)) --> 12((12)); depo[depo] --> 7((7)); depo[depo] --> 8((8)); depo[depo] --> 9((9)); depo[depo] --> 10((10)); depo[depo] --> 11((11)); depo[depo] --> 12((12));
```

0 7 8 9 11 12 0 2 3 1 4 5 6 10 0

# Genetik Algoritma

---

## ❑ Tanımlar

- *Uygunluk*: Belli bir kromozom ya da kromozomlar grubunun amaç fonksiyonundaki performansıdır.

# Genetik Algoritma

---

## ❑ Tanımlar

- *Ebeveyn Seçimi*: Genellikle uygunluk değeri daha yüksek olanın daha yüksek olasılıkla çocuk üretimi için seçilmesi işlemidir. Bazen tamamen rastlantısal olarak seçildiği de olur.

# Genetik Algoritma

---

## □ Tanımlar

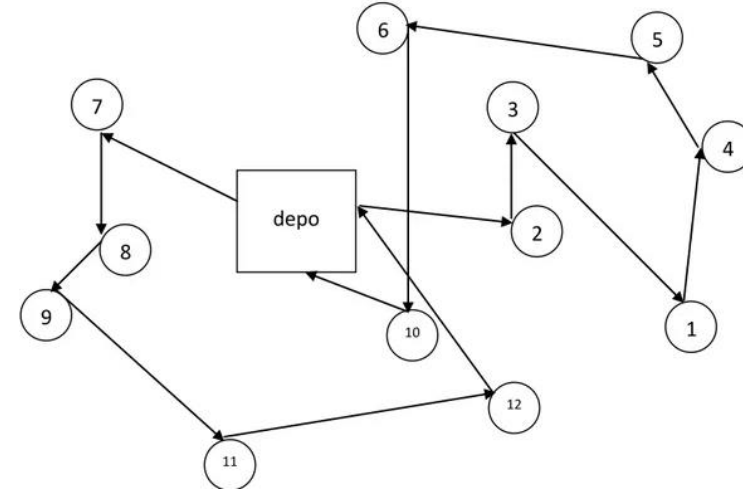
- *Çaprazlama*: Seçilmiş iki ebeveynden rastlantı olarak farklı genlerin çocuğa aktarılması işlemidir.

# Genetik Algoritma

## ❑ Tanımlar

- **Mutasyon:** Bir çocuğun rastlantı olarak genlerinin değerlerinin değiştirilmesi, başka bir ifadeyle bozulmasıdır. Doğal mutasyon gibi her zaman olmaz, düşük bir olasılıkla (örneğin 0.05'ten daha küçük) gerçekleşir.

| Birey    | Nitelik (Değer) | Kodlanmış Hali |
|----------|-----------------|----------------|
| C1       | 0100 1101       | 77             |
| Mutasyon | 0101 1101       | 313            |



0 7 8 9 11 12 0 2 3 1 4 5 6 10 0

0 7 8 9 11 12 0 2 1 3 4 5 6 10 0



# Genetik Algoritma

---

## ❑ Tanımlar

- ***Sonlandırma Koşulu.*** Bir GA probleminde iterasyonların kaç kez tekrarlanacağı, bir başka ifade ile kaç adet yeni nesil oluşturulacağı, problemde tanımlanan sonlandırma koşulu ile ilişkilidir. Önceden algoritmanın sonlandırılacağı iterasyonu belirlemek bir yöntem olabileceği gibi, ardarda belirli sayıda iterasyon çözümde iyileşme olmaması durumunda algoritmanın sonlandırılması da alternatif bir yöntemdir.

# Genetik Algoritma

---

## ❑ Uygulama Alanları

- *Optimizasyon*
  - *Bilgisayar Ağları Tasarımı*
  - *Finansal Modelleme*
  - *Üretim Yönetimi*
  - *Montaj Hattı Dengeleme*
  - *Çizelgeleme*
  - *Tesis Yerleşim Problemi*
  - *Atama Problemi*
  - *Taşıma Problemi*
  - *Araç Rotalama Problemi*
-

# Genetik Algoritma

---

## ❑ Genetik Algoritma Operatörler

- *Yeniden Üretme/Seçim (Reproduction)*
- *Çaprazlama*
- *Mutasyon*

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

GA uygulamalarında daha yüksek uygunluğa sahip bireylerin daha fazla çoğalacağı temeline dayalı doğal seleksiyon düşüncesi yeniden üretme veya seleksiyon/seçim olarak adlandırılır.

Yeniden üretim veya seçim operatörü yeni bir birey yaratmak için bireylerin popülasyondan nasıl seçileceği ve bunlardan hangi sayıda yeni birey oluşturulacağı konusuyla ilgilidir.

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

Seçim, değerlendirme fonksiyonu değerlerine göre popülasyondaki bireylerin seçilip ayrılması yöntemidir. Bu, daha yüksek uygunluk değeri birey için daha fazla seçilme şansı anlamına gelmektedir.

Seçim baskısı, daha iyi bireylerin (uygunluk değeri yüksek olan) seçildiği bir seviyeyi ifade eder. Daha yüksek seçim baskısında daha da iyi bireyler belirlenir. Seçim baskısı genetik algoritmayı daha iyi nesiller için popülasyon uygunluğunu artırmaya sağlar.

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

- Rulet Tekerleği Seçimi
- Rastgele Seçim
- Sıralama (Rank) Seçimi
- Turnuva Seçimi

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

***Rulet Tekerleği Seçimi:*** Bu yöntemin ana prensibi, bireylerin uygunluk değerlerine orantılı olarak ağırlıklandırılmış dilimlerden oluşan bir rulet tekerleği ile doğrusal bir arama yapılmasıdır. Hedef değer, popülasyondaki uygunluklar toplamının olasılıklı oranından oluşan bir settir.

Bu metot ortalama güçlü bir seçim tekniğidir, çünkü uygun bireylerin seçilme garantisi yoktur ancak seçilme olasılıkları yüksektir



# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

***Rulet Tekerleği Seçimi:*** Bir bireyin beklenen değeri popülasyonun gerçek uygunluğuna bölünerek bulunan uygunluk değeridir. Her bir bireye oransal uygunluğu ile orantılı olarak rulet tekerleğinde bir dilim ayrılır. Tekerlek N defa döndürülür. Burada N popülasyondaki birey sayısıdır. Her bir turda tekerleğin işaretçisi altında kalan birey gelecek nesil için eşleşme havuzuna seçilir.

# Genetik Algoritma

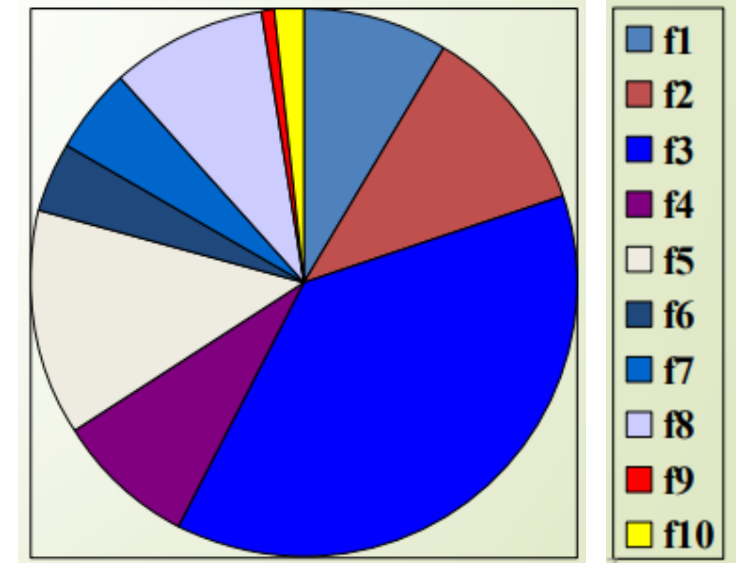
---

## ❑ Yeniden Üretme/Seçim (reproduction)

### *Rulet Tekerleği Seçimi*

N adet bireyin uygunluk değerlerinin  $f_1, f_2, \dots, f_n$  olduğunu varsayıldığında,  $i$ . bireyin seçilme olasılığı  $p_i$ ,

$$p_i = \frac{f_i}{\sum_{j=1}^N f_j}$$



# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

*Rastgele Seçim:* Bu teknikte gelecek nesli oluşturmak için aile popülasyondan rastgele seçilerek belirlenir.

N adet bireyin uygunluk değerlerinin  $f_1, f_2, \dots, f_n$  olduğunu varsayıldığında,  $i$ . bireyin seçilme olasılığı  $p_i$ ,

$$p_i = \frac{1}{N}$$

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

***Sıralama Seçim:*** Uygunluk değerlerinin çok fazla farklılık sergilediği durumda rulet tekerleği seçimini uygulamak zor olmaktadır. Eğer en iyi kromozomun uygunluğu %90 ise tekerlekte de %90 yer kaplayacaktır ve bu durumda diğer bireylerin seçilme şansı çok az olmaktadır.

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

***Sıralama Seçim:*** Seçilme olasılığı mevcut popülasyondaki bireylerin uygunluğunun sıralamadaki yerine göre belirlenmektedir. Popülasyonu en iyiden en kötü bireye doğru sıralar. Her bir bireyin beklenen değeri onun mutlak uygunluk değeri ile değil sıralamadaki yerine göre belirlenmektedir. Uygunluklar arasındaki mutlak farklılıklar dikkate alınmadığı için sıralama seçiminde uygunlukları derecelendirme gereksinimi de bulunmamaktadır.

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

**Sıralama Seçim:** Bu seçim yakınsamayı yavaşlatır ancak çok hızlı yakınsamayı da önler. Çeşitliliği korur ve daha başarılı arama sağlar.

N adet bireyin uygunluk değerlerine  $f_1, f_2, \dots, f_n$  göre sıralaması  $s_1, s_2, \dots, s_n$  olduğu varsayıldığında,  $i$ . bireyin seçilme olasılığı  $p_i$ ,

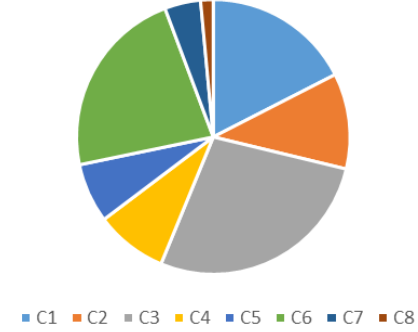
$$p_i = \frac{s_i}{\sum_{j=1}^N f_j}$$

# Genetik Algoritma

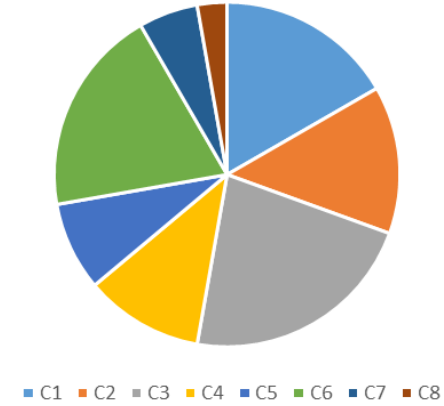
## ❑ Örnek

| Bireyler | $f_i$      | Rulet Tekeleği |                 | Sıralama  |          |                 |
|----------|------------|----------------|-----------------|-----------|----------|-----------------|
|          |            | $p_i$          | Kümülatif $p_i$ | $s_i$     | $p_i$    | Kümülatif $p_i$ |
| C1       | 140        | 0,1750         | 0,1750          | 6         | 0,1667   | 0,1667          |
| C2       | 90         | 0,1125         | 0,2875          | 5         | 0,1389   | 0,3056          |
| C3       | 220        | 0,2750         | 0,5625          | 8         | 0,2222   | 0,5278          |
| C4       | 68         | 0,0850         | 0,6475          | 4         | 0,1111   | 0,6389          |
| C5       | 56         | 0,0700         | 0,7175          | 3         | 0,0833   | 0,7222          |
| C6       | 180        | 0,2250         | 0,9425          | 7         | 0,1944   | 0,9167          |
| C7       | 34         | 0,0425         | 0,9850          | 2         | 0,0556   | 0,9722          |
| C8       | 12         | 0,0150         | 1,0000          | 1         | 0,0278   | 1,0000          |
|          | <b>800</b> | <b>1</b>       |                 | <b>36</b> | <b>1</b> |                 |

Rulet Tekerleği



Sıralama





# Genetik Algoritma

## ❑ Örnek

| Bireyler | $f_i$ | Rulet Tekeleği |                 | Sıralama |        |                 |  |        |
|----------|-------|----------------|-----------------|----------|--------|-----------------|--|--------|
|          |       | $p_i$          | Kümülatif $p_i$ | $s_i$    | $p_i$  | Kümülatif $p_i$ |  |        |
| C1       | 140   | 0,1750         | 0,1750          | 6        | 0,1667 | 0,1667          |  | 0,6447 |
| C2       | 90    | 0,1125         | 0,2875          | 5        | 0,1389 | 0,3056          |  |        |
| C3       | 220   | 0,2750         | 0,5625          | 8        | 0,2222 | 0,5278          |  |        |
| C4       | 68    | 0,0850         | 0,6475          | 4        | 0,1111 | 0,6389          |  |        |
| C5       | 56    | 0,0700         | 0,7175          | 3        | 0,0833 | 0,7222          |  |        |
| C6       | 180   | 0,2250         | 0,9425          | 7        | 0,1944 | 0,9167          |  |        |
| C7       | 34    | 0,0425         | 0,9850          | 2        | 0,0556 | 0,9722          |  |        |
| C8       | 12    | 0,0150         | 1,0000          | 1        | 0,0278 | 1,0000          |  |        |
|          | 800   | 1              |                 | 36       | 1      |                 |  |        |

# Genetik Algoritma

## ❑ Örnek

| Bireyler | $f_i$ | Rulet Tekeleği |                 | Sıralama |        |                 |  |        |
|----------|-------|----------------|-----------------|----------|--------|-----------------|--|--------|
|          |       | $p_i$          | Kümülatif $p_i$ | $s_i$    | $p_i$  | Kümülatif $p_i$ |  |        |
| C1       | 140   | 0,1750         | 0,1750          | 6        | 0,1667 | 0,1667          |  | 0,5507 |
| C2       | 90    | 0,1125         | 0,2875          | 5        | 0,1389 | 0,3056          |  |        |
| C3       | 220   | 0,2750         | 0,5625          | 8        | 0,2222 | 0,5278          |  |        |
| C4       | 68    | 0,0850         | 0,6475          | 4        | 0,1111 | 0,6389          |  |        |
| C5       | 56    | 0,0700         | 0,7175          | 3        | 0,0833 | 0,7222          |  |        |
| C6       | 180   | 0,2250         | 0,9425          | 7        | 0,1944 | 0,9167          |  |        |
| C7       | 34    | 0,0425         | 0,9850          | 2        | 0,0556 | 0,9722          |  |        |
| C8       | 12    | 0,0150         | 1,0000          | 1        | 0,0278 | 1,0000          |  |        |
|          | 800   | 1              |                 | 36       | 1      |                 |  |        |

# Genetik Algoritma

## ❑ Örnek

| Bireyler | $f_i$ | Rulet Tekeleği |                 | Sıralama |        |                 |  |        |
|----------|-------|----------------|-----------------|----------|--------|-----------------|--|--------|
|          |       | $p_i$          | Kümülatif $p_i$ | $s_i$    | $p_i$  | Kümülatif $p_i$ |  |        |
| C1       | 140   | 0,1750         | 0,1750          | 6        | 0,1667 | 0,1667          |  | 0,4227 |
| C2       | 90    | 0,1125         | 0,2875          | 5        | 0,1389 | 0,3056          |  |        |
| C3       | 220   | 0,2750         | 0,5625          | 8        | 0,2222 | 0,5278          |  |        |
| C4       | 68    | 0,0850         | 0,6475          | 4        | 0,1111 | 0,6389          |  |        |
| C5       | 56    | 0,0700         | 0,7175          | 3        | 0,0833 | 0,7222          |  |        |
| C6       | 180   | 0,2250         | 0,9425          | 7        | 0,1944 | 0,9167          |  |        |
| C7       | 34    | 0,0425         | 0,9850          | 2        | 0,0556 | 0,9722          |  |        |
| C8       | 12    | 0,0150         | 1,0000          | 1        | 0,0278 | 1,0000          |  |        |
|          | 800   | 1              |                 | 36       | 1      |                 |  |        |

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

*Turnuva Seçimi:* Rulet tekerleri seçiminden farklı olarak turnuva seçimi bireyler arasında bir turnuva düzenleyerek seçim baskısı sağlar. Kazanan birey uygunluk değerine göre belirlenir.

Genelde turnuva iki birey arasında düzenlenir ancak keyfi olarak seçilen belirli bir sayıdaki birey arasında da uygulanabilir.

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

### *Turnuva Seçimi*

- *Turnuva Seçimi Parametresi*: Turnuva büyüklüğü ( $t$ )
- Bir ebeveyn seçmek için:
  - Popülasyondan rassal olarak  $t$  adet birey seç
  - Seçilen  $t$  adet birey içinden en iyi uygunluk değerine sahip olanı al.

# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

### *Turnuva Seçimi*

- *Örnek*

| Bireyler | $f_i$ |  |        |        |  |          |
|----------|-------|--|--------|--------|--|----------|
|          |       |  | Aday 1 | Aday 2 |  | Ebeven 1 |
| C1       | 140   |  | 8      | 4      |  | C4       |
| C2       | 90    |  | 12     | 68     |  |          |
| C3       | 220   |  | C8     | C4     |  |          |
| C4       | 68    |  |        |        |  |          |
| C5       | 56    |  | Aday 1 | Aday 2 |  | Ebeven 2 |
| C6       | 180   |  | 2      | 7      |  | C2       |
| C7       | 34    |  | 90     | 34     |  |          |
| C8       | 12    |  | C2     | C7     |  |          |

# Genetik Algoritma

## ❑ Yeniden Üretme/Seçim (reproduction)

### *Turnuva Seçimi*

- *Örnek*

| Bireyler | $f_i$ |  |        |        |  |          |
|----------|-------|--|--------|--------|--|----------|
|          |       |  | Aday 1 | Aday 2 |  | Ebeven 1 |
| C1       | 140   |  | 4      | 2      |  | C2       |
| C2       | 90    |  | 68     | 90     |  |          |
| C3       | 220   |  | C4     | C2     |  |          |
| C4       | 68    |  |        |        |  |          |
| C5       | 56    |  | Aday 1 | Aday 2 |  | Ebeven 2 |
| C6       | 180   |  | 5      | 1      |  | C1       |
| C7       | 34    |  | 56     | 140    |  |          |
| C8       | 12    |  | C5     | C1     |  |          |



# Genetik Algoritma

---

## ❑ Yeniden Üretme/Seçim (reproduction)

### *Turnuva Seçimi*

- *Örnek*

| Bireyler | $f_i$ |  |        |        |  |          |
|----------|-------|--|--------|--------|--|----------|
|          |       |  | Aday 1 | Aday 2 |  | Ebeven 1 |
| C1       | 140   |  | 8      | 4      |  | C4       |
| C2       | 90    |  | 12     | 68     |  |          |
| C3       | 220   |  | C8     | C4     |  |          |
| C4       | 68    |  |        |        |  |          |
| C5       | 56    |  | Aday 1 | Aday 2 |  | Ebeven 2 |
| C6       | 180   |  | 7      | 1      |  | C1       |
| C7       | 34    |  | 34     | 140    |  |          |
| C8       | 12    |  | C7     | C1     |  |          |

# Genetik Algoritma

---

## ❑ Çaprazlama

Çaprazlama, uygunluk değerlerine göre seçilen bireyler arasında gen alışverişinde bulunulmasını ifade eder. Mevcut gen havuzunun potansiyelini arttırmak için, bir önceki nesilden daha iyi nitelikler içeren yeni bireyler elde edebilmek için çaprazlama operatörü kullanılır.

Bu aşamada hangi kromozomların hangi kromozomlar ile çaprazlanacağı geleneksel genetik algoritma uygulamalarında rassal olarak belirlenmekte ve çeşitli çaprazlama kurallarına göre çaprazlama yapılmaktadır.

# Genetik Algoritma

---

## ❑ Çaprazlama Türleri

- *Tek Nokta Çaprazlama*
- *Çift/Çok Nokta Çaprazlama*
- *Tekdüze (Uniform) Çaprazlama*
- *Sıralı Kromozomlar için Çaprazlama*

# Genetik Algoritma

---

## ❑ Çaprazlama

***Tek Nokta Çaprazlama:*** Bu çaprazlama yöntemine göre kromozom uzunluğundan küçük bir çaprazlama noktası belirlenir. Ebeveyn kromozomların çaprazlama noktasının (aşağıda X ile gösterilmiştir) sağındaki veya solundaki genleri aynen çocuk kromozoma aktarılır. Böylece çocuk1 ve çocuk2 biçiminde aşağıdaki gibi iki çocuk oluşur.

# Genetik Algoritma

---

❑ Çaprazlama

*Tek Nokta Çaprazlama*

|                            |   |   |   |   |   |   |   |   |                          |   |   |   |   |   |   |   |   |
|----------------------------|---|---|---|---|---|---|---|---|--------------------------|---|---|---|---|---|---|---|---|
| <i>Ebeveyn<sub>1</sub></i> | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | <i>Çocuk<sub>1</sub></i> | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 |
| X                          |   |   |   |   |   |   |   |   |                          |   |   |   |   |   |   |   |   |
| <i>Ebeveyn<sub>2</sub></i> | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | <i>Çocuk<sub>2</sub></i> | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

# Genetik Algoritma

---

## ❑ Çaprazlama

***Tek Nokta Çaprazlama:*** Tek-nokta çaprazlama tekniği kromozomun herhangi bir kodlama kullanılmadığı durumlar için kullanılabilir. Çaprazlama noktası rastlantı olarak belirlenir. Ancak bu kez  $X \in [0,1]$  arasında bir doğal sayıdır. Buna göre  $E1$  ve  $E2$  ebeveynleri temsil edecek olursa, çocuk çaprazlama sonucu aşağıdaki gibi oluşacaktır.

$$cocuk = \frac{E_1 + E_2}{2} + |E_1 - E_2| * w * (X - 0.5)$$

# Genetik Algoritma

---

## □ Çaprazlama

### *Tek Nokta Çaprazlama*

- $cocuk = \frac{E_1 + E_2}{2} + |E_1 - E_2| * w * (X - 0.5)$

- $E_1 = 5$

- $E_2 = 8$

- $C = (5 + 8) / 2 + |5 - 8| * 0.2 * (1 - 0.5) = 6.8$

# Genetik Algoritma

---

## ❑ Çaprazlama

***Çift Nokta Çaprazlama:*** Çift nokta çaprazlama, tek-nokta çaprazlama tekniğine oldukça benzer bir tekniktir. Tek fark adından da anlaşılabilceği gibi iki adet çaprazlama noktası (X1 ve X2) rastlantı olarak seçilmektedir. Çocuklara genler aktarılırken çaprazlama noktalarının dışında kalanlar bir ebeveyn kromozomdan, arasında kalanlar ise diğer ebeveyn kromozomdan gelir. Böylece çocuk1 ve çocuk2 biçiminde aşağıdaki gibi iki çocuk oluşur.



# Genetik Algoritma

---

❑ Çaprazlama

*Çift Nokta Çaprazlama*

|                            |    |   |   |   |    |   |   |   |                          |   |   |   |   |   |   |   |   |
|----------------------------|----|---|---|---|----|---|---|---|--------------------------|---|---|---|---|---|---|---|---|
| <i>Ebeveyn<sub>1</sub></i> | 1  | 0 | 1 | 0 | 0  | 1 | 0 | 1 | <i>Çocuk<sub>1</sub></i> | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |
|                            | X1 |   |   |   | X2 |   |   |   |                          |   |   |   |   |   |   |   |   |
| <i>Ebeveyn<sub>2</sub></i> | 1  | 1 | 0 | 1 | 1  | 0 | 0 | 1 | <i>Çocuk<sub>2</sub></i> | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |

# Genetik Algoritma

---

## ❑ Çaprazlama

*Tekdüze (Uniform) Çaprazlama:* Bu teknikte bir çaprazlama noktası kullanılmaz. Onun yerine ebeveynlerinden genler sırasıyla çocuğa kopyalanır. Söz konusu kopyalamada her bir gen anlık bir olasılığına göre ebeveynlerden birisinden gelmektedir. Genellikle her iki ebeveyne eşit olasılık verilir. Bu yöntemde tek bir çocuk oluşmaktadır. Örneğin  $i$  ise  $i$  genini birinci ebeveynden,  $j$  ise  $j$  genini ikinci ebeveynden kopyalanır.

# Genetik Algoritma

---

## □ Çaprazlama

### *Tekdüze (Uniform) Çaprazlama*

|                            |     |     |     |     |     |     |     |     |              |   |   |   |   |   |   |   |   |
|----------------------------|-----|-----|-----|-----|-----|-----|-----|-----|--------------|---|---|---|---|---|---|---|---|
| <i>Ebeveyn<sub>1</sub></i> | 1   | 0   | 1   | 0   | 0   | 1   | 0   | 1   |              |   |   |   |   |   |   |   |   |
| $X_i$                      | 0.3 | 1.0 | 0.4 | 0.8 | 0.5 | 0.4 | 0.2 | 0.8 | <i>Çocuk</i> | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 |
| <i>Ebeveyn<sub>2</sub></i> | 1   | 1   | 0   | 1   | 1   | 0   | 0   | 1   |              |   |   |   |   |   |   |   |   |

$X_i < 0,5$  ise  $E1'$ den özellik al, değil ise  $E2'$ den özellik al.

# Genetik Algoritma

---

## ❑ Çaprazlama

*Sıralı Kromozomlar için Çaprazlama:* Kromozom kodlanması bazı hallerde sırayı temsil eden bir kodlamadır. Bilindiği gibi genler kromozomda bir sıraya sahiptir. Bir genin aldığı değer o sırada yer alması gereken nesneyi temsil edecektir.

# Genetik Algoritma

---

## ❑ Çaprazlama

*Sıralı Kromozomlar için Çaprazlama:* Her bir iş bir harf ile temsil edilecek olursa, "ACDBEGFH" kromozomu, söz konusu makinede ilk olarak A işinin yapılacağı, son olarak da H işinin yapılacağını temsil eder ve "sıralı kromozom" olarak adlandırılır.

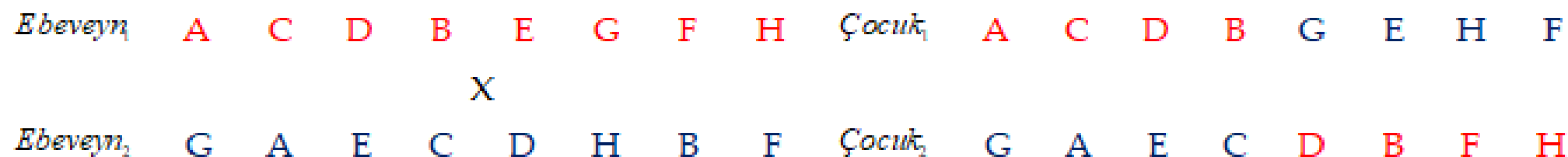
Sıralı kromozomların çaprazlanması için rastlantı bir çaprazlama noktası (x) seçilir. Birinci ebeveynden çaprazlama noktasına kadar tüm genleri aynen çocuğa aktarılır. Kalan kısım için ise sırasıyla diğer ebeveynden çocuğa henüz kopyalanmamış genler kopyalanır. Böylece aşağıdaki gibi iki çocuk oluşturulur.

# Genetik Algoritma

---

## ☐ Çaprazlama

*Sıralı Kromozomlar için Çaprazlama*



# Genetik Algoritma

---

## ❑ Mutasyon

Bir çocuğun rastlantı olarak genlerinin değerlerinin değiştirilmesi, başka bir ifadeyle bozulmasıdır. Doğal mutasyon gibi her zaman olmaz, düşük bir olasılıkla (örneğin 0.05'ten daha küçük) gerçekleşir.

Mutasyon, belirlenmiş bir ihtimal dahilinde yeni türetilmiş olan bireylerde yapılan küçük çaplı farklılaştırmalardır. Çaprazlama işlemini takip eden bir işlemidir ve bir bireyde rastgele değişimi sağlayarak yeni birey oluşmasına olanak verir. Bu aşama doğal mutasyon sürecinde olduğu gibi nesillerin farklılaşmasında önemli bir etkidir.

0 7 8 9 11 12 0 2 1 3 4 5 6 10 0



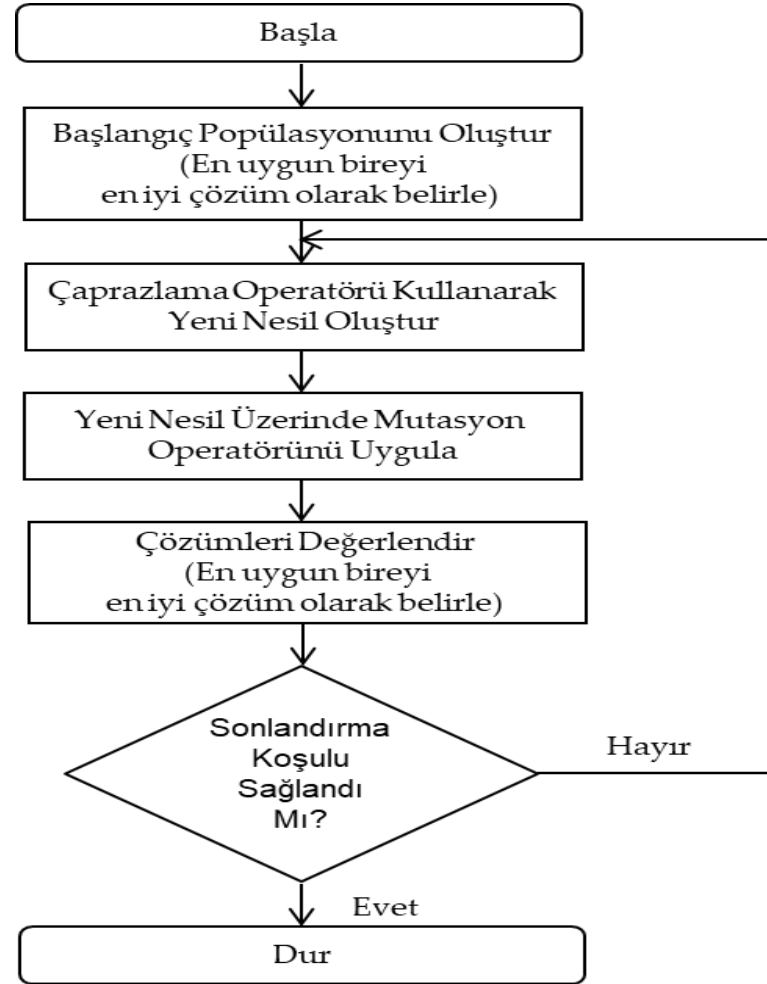
# Genetik Algoritma

---

- ❑ Rasgele seçilen bireylerden oluşan ilk popülasyon oluşturularak işlemlere başlanır. Popülasyonda yer alan her birey, ilgilenilen problem için potansiyel bir çözüm sunmaktadır. Bu bireylerin sunmuş oldukları çözümler genetik operatörler kullanılarak yavaş yavaş geliştirilmeye çalışılır. Bunun için yeni nesiller üretilir. Üretilen her bir yeni nesildeki bireyler uygunluk ölçütleri kullanılarak değerlendirilir ve daha iyi çözüm sunan bireyler belirlenmeye çalışılır. Yeni nesiller üretilmesi işlemi önceden belirlenmiş bir sonlandırma koşulu sağlanıncaya kadar devam ettirilir.

# Genetik Algoritma

## Genetik Algoritmalar Süreci



# Genetik Algoritma

---

*Örnek:* Bir dağcı, sırt çantasında bazı eşyaları/aletleri taşımak istemektedir. Bazı aletler çok ağırken, bazıları çok hafif ama her aletin önemi değeri birbirine eşit değildir. Dağcı daha önemli aletleri yanına almak istiyor fakat çantasında en fazla 15 kilogram taşıyabiliyor. Sırt çantasına konulabilecek toplamda 7 farklı çeşit alet bulunmaktadır. Aşağıdaki tabloda aletlerin önem düzeyleri ve ağırlıkları verilmiştir. Dağcı en fazla 15 kilogram alarak en yüksek önem değerine sahip hangi aletleri çantasına almalıdır.

| <u>Aletler</u> | <u>Aletin Önem Düzeyi</u> | <u>Aletin Ağırlığı (Kg)</u> |
|----------------|---------------------------|-----------------------------|
| 1. Alet        | 8                         | 3                           |
| 2. Alet        | 7                         | 2                           |
| 3. Alet        | 10                        | 7                           |
| 4. Alet        | 11                        | 6                           |
| 5. Alet        | 8                         | 4                           |
| 6. Alet        | 9                         | 10                          |
| 7. Alet        | 5                         | 3                           |

# Genetik Algoritma

Örnek :

| Popülasyon |         |         |         |         |         |         |         |           |
|------------|---------|---------|---------|---------|---------|---------|---------|-----------|
| 1. Alet    | 2. Alet | 3. Alet | 4. Alet | 5. Alet | 6. Alet | 7. Alet | Ağırlık | Önem      |
| 0          | 0       | 0       | 0       | 0       | 0       | 0       | 0       | 0         |
| 1          | 0       | 1       | 0       | 1       | 0       | 0       | 14      | 26        |
| 1          | 0       | 0       | 1       | 0       | 0       | 1       | 12      | 24        |
| 1          | 0       | 0       | 1       | 1       | 0       | 0       | 13      | 27        |
| 1          | 0       | 1       | 0       | 0       | 0       | 0       | 10      | 18        |
| 1          | 1       | 0       | 1       | 0       | 0       | 0       | 11      | 26        |
| 0          | 0       | 0       | 0       | 1       | 1       | 1       | 17      | Çözüm Yok |
| 1          | 0       | 0       | 0       | 1       | 0       | 0       | 7       | 16        |
| 1          | 0       | 0       | 0       | 1       | 1       | 1       | 20      | Çözüm Yok |
| 0          | 1       | 1       | 0       | 0       | 0       | 1       | 12      | 22        |
| 0          | 1       | 0       | 0       | 0       | 0       | 0       | 2       | 7         |
| 1          | 1       | 0       | 0       | 1       | 0       | 1       | 12      | 28        |
| 1          | 0       | 0       | 0       | 1       | 1       | 1       | 20      | Çözüm Yok |
| 1          | 1       | 1       | 0       | 1       | 1       | 0       | 26      | Çözüm Yok |
| 0          | 1       | 1       | 0       | 0       | 0       | 1       | 12      | 22        |
|            |         |         |         |         |         |         |         | 28        |

# Genetik Algoritma

Örnek :

| Popülasyon |         |         |         |         |         |         |         |           |
|------------|---------|---------|---------|---------|---------|---------|---------|-----------|
| 1. Alet    | 2. Alet | 3. Alet | 4. Alet | 5. Alet | 6. Alet | 7. Alet | Ağırlık | Önem      |
| 1          | 0       | 0       | 0       | 0       | 0       | 1       | 6       | 13        |
| 1          | 0       | 0       | 0       | 0       | 0       | 1       | 6       | 13        |
| 1          | 1       | 0       | 1       | 1       | 0       | 0       | 15      | 34        |
| 1          | 0       | 0       | 1       | 0       | 0       | 0       | 9       | 19        |
| 1          | 1       | 0       | 1       | 0       | 0       | 1       | 14      | 31        |
| 0          | 0       | 1       | 0       | 0       | 0       | 0       | 7       | 10        |
| 0          | 0       | 0       | 1       | 1       | 0       | 0       | 10      | 19        |
| 0          | 0       | 0       | 1       | 1       | 1       | 0       | 20      | Çözüm Yok |
| 0          | 0       | 1       | 0       | 1       | 0       | 0       | 11      | 18        |
| 0          | 0       | 1       | 0       | 1       | 1       | 0       | 21      | Çözüm Yok |
| 0          | 0       | 1       | 1       | 1       | 1       | 1       | 30      | Çözüm Yok |
| 0          | 0       | 1       | 0       | 0       | 0       | 1       | 10      | 15        |
| 0          | 0       | 1       | 1       | 0       | 0       | 1       | 16      | Çözüm Yok |
| 1          | 0       | 0       | 0       | 0       | 0       | 1       | 6       | 13        |
| 0          | 1       | 0       | 0       | 0       | 1       | 0       | 12      | 16        |
|            |         |         |         |         |         |         |         | 34        |

# Genetik Algoritma

---

## *Weka Uygulaması*

WEKA yazılımında Genetik Algoritma, Özellik Seçimi (Select Attributes) işlemi için kullanılmaktadır.

Özellik seçimi, öncelikle açıklayıcı olmayan veya gereksiz tahmin unsurlarını (değişkenlerini) modelden kaldırmaya odaklanır.

# Genetik Algoritma

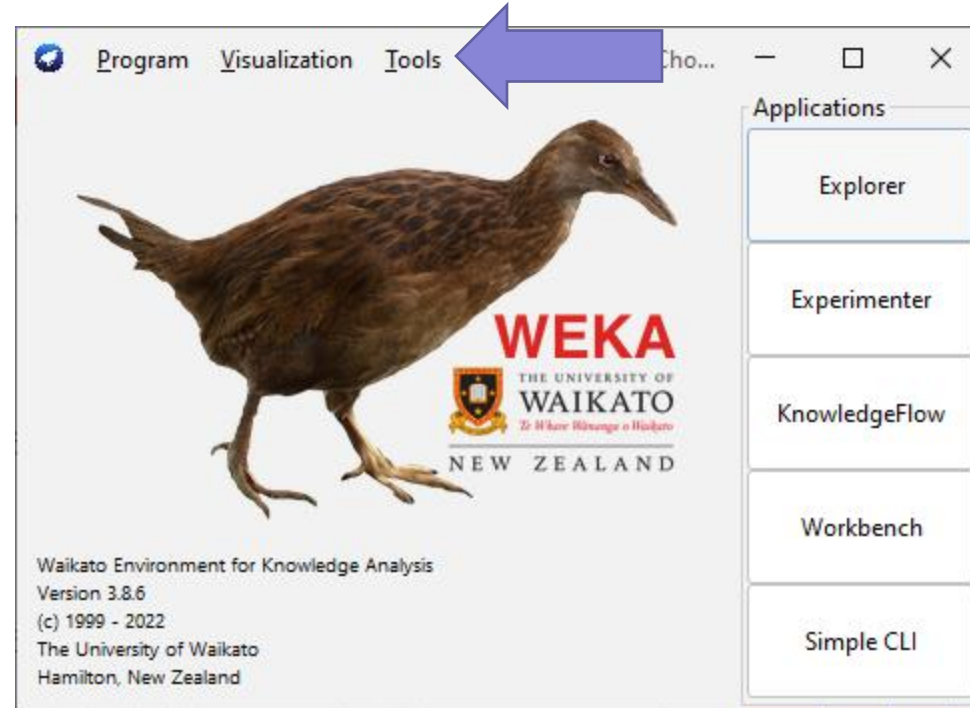
---

## *Weka Uygulaması: Neden Değişken Seçimi Yapılır?*

- Gereğinden fazla değişkenin olduğu bir modelin açıklanması ve yorumlanması oldukça karmaşık olabilir. Bu yüzden değişken seçimi ile modeli daha kolay anlaşılabilir bir biçime dönüştürülebilir.
  - Değişken seçimi ile modelde oluşabilecek overfitting (ezberleme, sadece oluşturulduğu veri seti için iyi çalışan ama farklı veri setleri için başarısı önemli ölçüde düşen) durumunun önüne geçilebilir.
  - Bir veri setinde değişkenlerin bazıları kendi aralarında yüksek bir ilişkiye (multicollinearity) sahip olabilir. Bu durumda bu değişkenlerden birisinin silinmesi demek gereksiz bir değişkenden kurtulmamız anlamına gelirken model başarısını da artırmış oluruz.
  - Değişkenlerin seçimi ile azalan girdi değişken sayısı sayesinde makine öğrenmesi süreci içinde veri seti eğitim aşamasının daha hızlı gerçekleşebilmesi sağlanabilir.
  - Doğru değişken seçimi ile model başarı oranı artırılabilir.
-

# Genetik Algoritma

## *Weka Uygulaması*

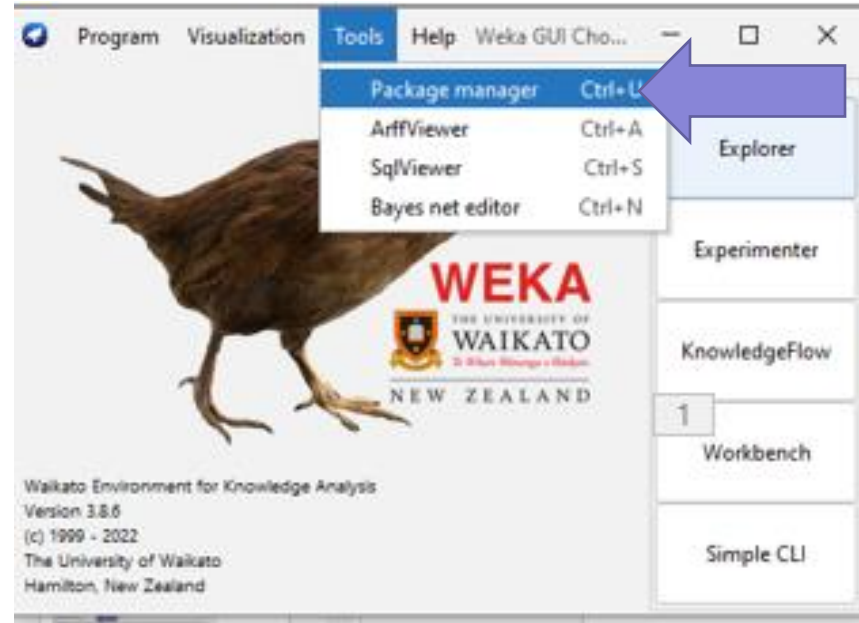




# Genetik Algoritma

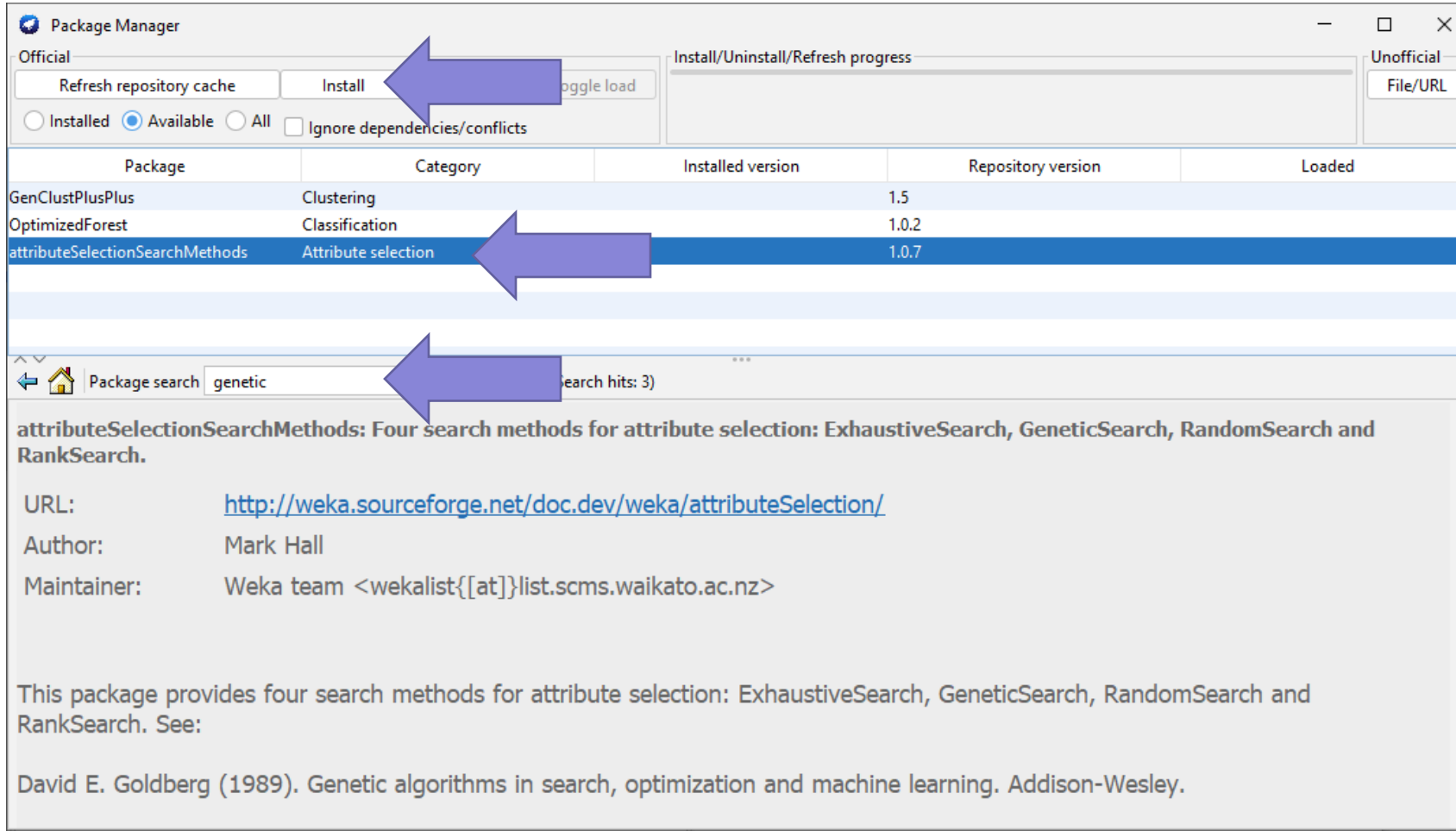
---

## *Weka Uygulaması*



# Genetik Algoritma

## *Weka Uygulaması*



The screenshot shows the Weka Package Manager interface. At the top, there are buttons for 'Refresh repository cache', 'Install', and 'Toggle load'. Below these are radio buttons for 'Installed', 'Available' (selected), and 'All', along with a checkbox for 'Ignore dependencies/conflicts'. A progress bar for 'Install/Uninstall/Refresh progress' is visible. The main table lists packages with columns for Package, Category, Installed version, Repository version, and Loaded. The 'attributeSelectionSearchMethods' package is highlighted in blue. Below the table, there is a search bar with 'genetic' entered and 'search hits: 3)'. The details for the selected package are shown below the search bar.

| Package                         | Category            | Installed version | Repository version | Loaded |
|---------------------------------|---------------------|-------------------|--------------------|--------|
| GenClustPlusPlus                | Clustering          |                   | 1.5                |        |
| OptimizedForest                 | Classification      |                   | 1.0.2              |        |
| attributeSelectionSearchMethods | Attribute selection |                   | 1.0.7              |        |

Package search: genetic search hits: 3)

**attributeSelectionSearchMethods:** Four search methods for attribute selection: ExhaustiveSearch, GeneticSearch, RandomSearch and RankSearch.

URL: <http://weka.sourceforge.net/doc.dev/weka/attributeSelection/>

Author: Mark Hall

Maintainer: Weka team <wekalist@[at]list.scms.waikato.ac.nz>

This package provides four search methods for attribute selection: ExhaustiveSearch, GeneticSearch, RandomSearch and RankSearch. See:

David E. Goldberg (1989). Genetic algorithms in search, optimization and machine learning. Addison-Wesley.

# Genetik Algoritma

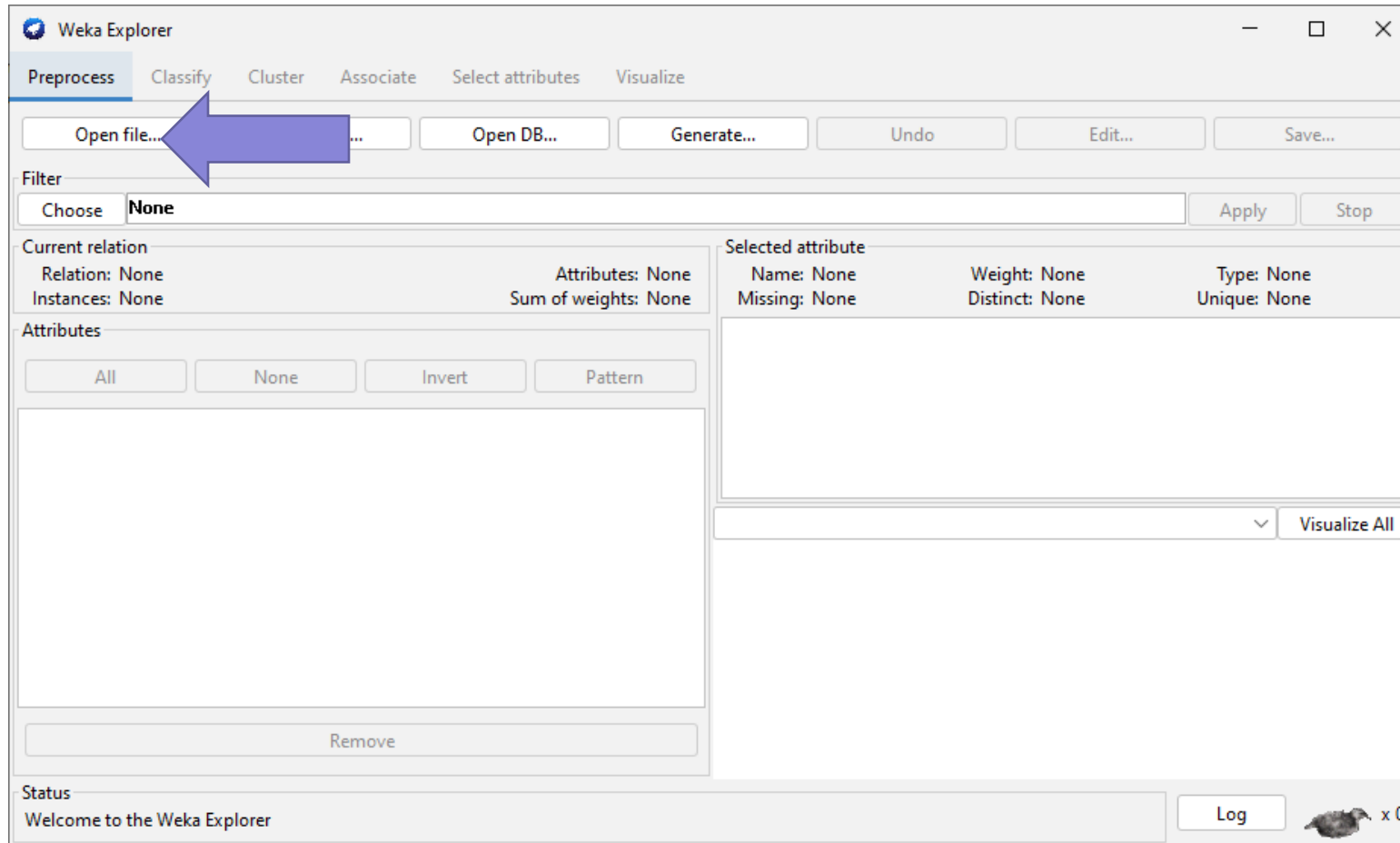
---

## *Weka Uygulaması*



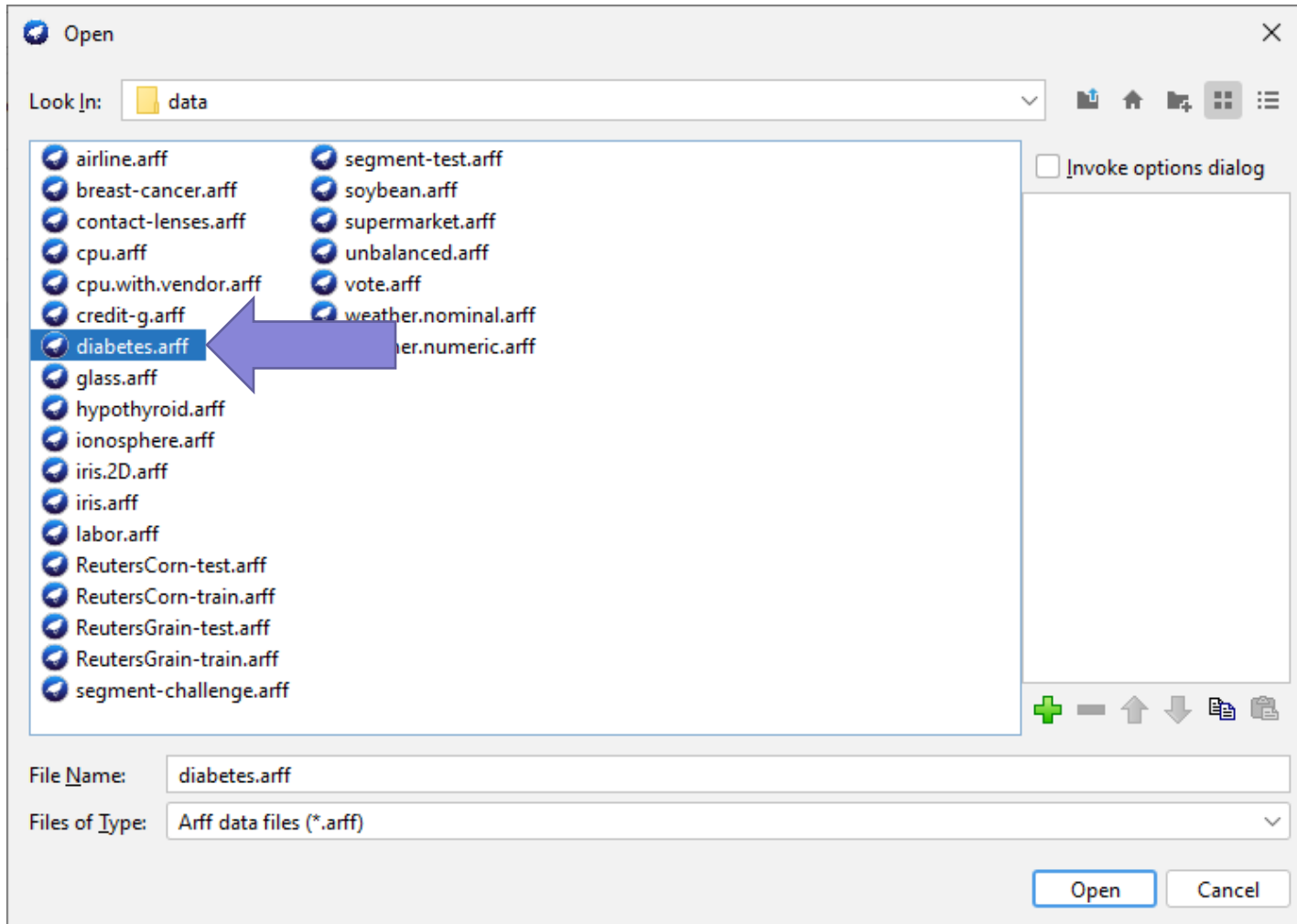
# Genetik Algoritma

## *Weka Uygulaması*



# Genetik Algoritma

## *Weka Uygulaması*



# Genetik Algoritma

## Weka Uygulaması

Weka Explorer

Preprocess Classify Cluster Associate Select attributes

Open file... Open URL... Open DB... Generate... Undo Edit... Save...

Filter  
Choose None Apply Stop

Current relation  
Relation: pima\_diabetes  
Instances: 768  
Attributes: 9  
Sum of weights: 768

Attributes  
All None Invert Pattern

| No. | Name                                     |
|-----|------------------------------------------|
| 1   | <input checked="" type="checkbox"/> preg |
| 2   | <input type="checkbox"/> plas            |
| 3   | <input type="checkbox"/> pres            |
| 4   | <input type="checkbox"/> skin            |
| 5   | <input type="checkbox"/> insu            |
| 6   | <input type="checkbox"/> mass            |
| 7   | <input type="checkbox"/> pedi            |
| 8   | <input type="checkbox"/> age             |
| 9   | <input type="checkbox"/> class           |

Remove

Status  
OK

Selected attribute  
Name: preg  
Missing: 0 (0%)  
Distinct: 17  
Type: Numeric  
Unique: 2 (0%)

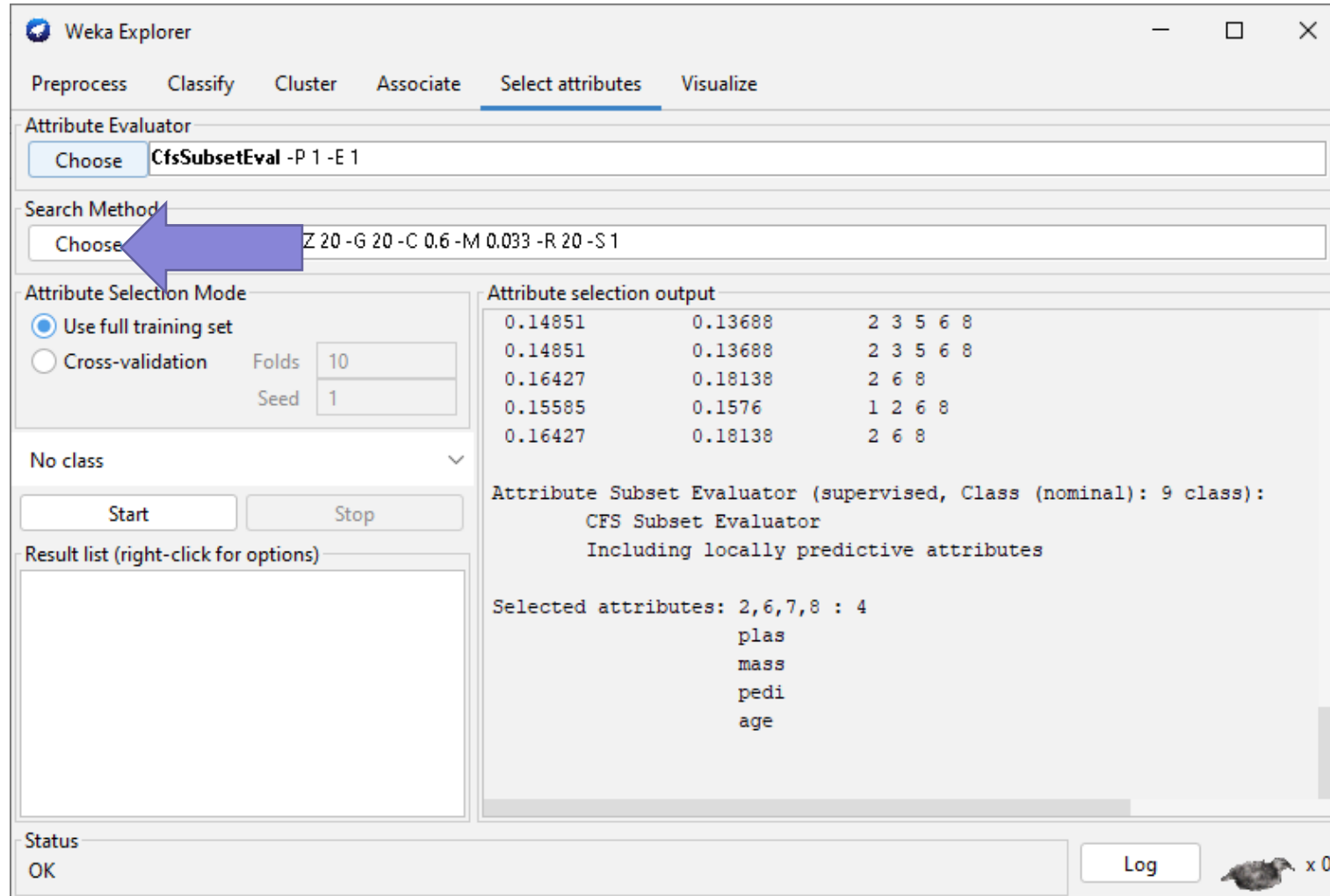
| Statistic | Value |
|-----------|-------|
| Minimum   | 0     |
| Maximum   | 17    |
| Mean      | 3.845 |
| StdDev    | 3.37  |

Class: class (Nom) Visualize All

Log x 0

# Genetik Algoritma

## *Weka Uygulaması*



The screenshot shows the Weka Explorer application window. The 'Select attributes' tab is active. The 'Attribute Evaluator' is set to 'CfsSubsetEval -P 1 -E 1'. The 'Search Method' is set to 'Z 20 -G 20 -C 0.6 -M 0.033 -R 20 -S 1', with a blue arrow pointing to the 'Choose' button. The 'Attribute Selection Mode' has 'Use full training set' selected. The 'No class' dropdown is open. The 'Start' button is visible. The 'Result list (right-click for options)' is empty. The 'Attribute selection output' pane displays the following text:

```
0.14851      0.13688      2 3 5 6 8
0.14851      0.13688      2 3 5 6 8
0.16427      0.18138      2 6 8
0.15585      0.1576       1 2 6 8
0.16427      0.18138      2 6 8

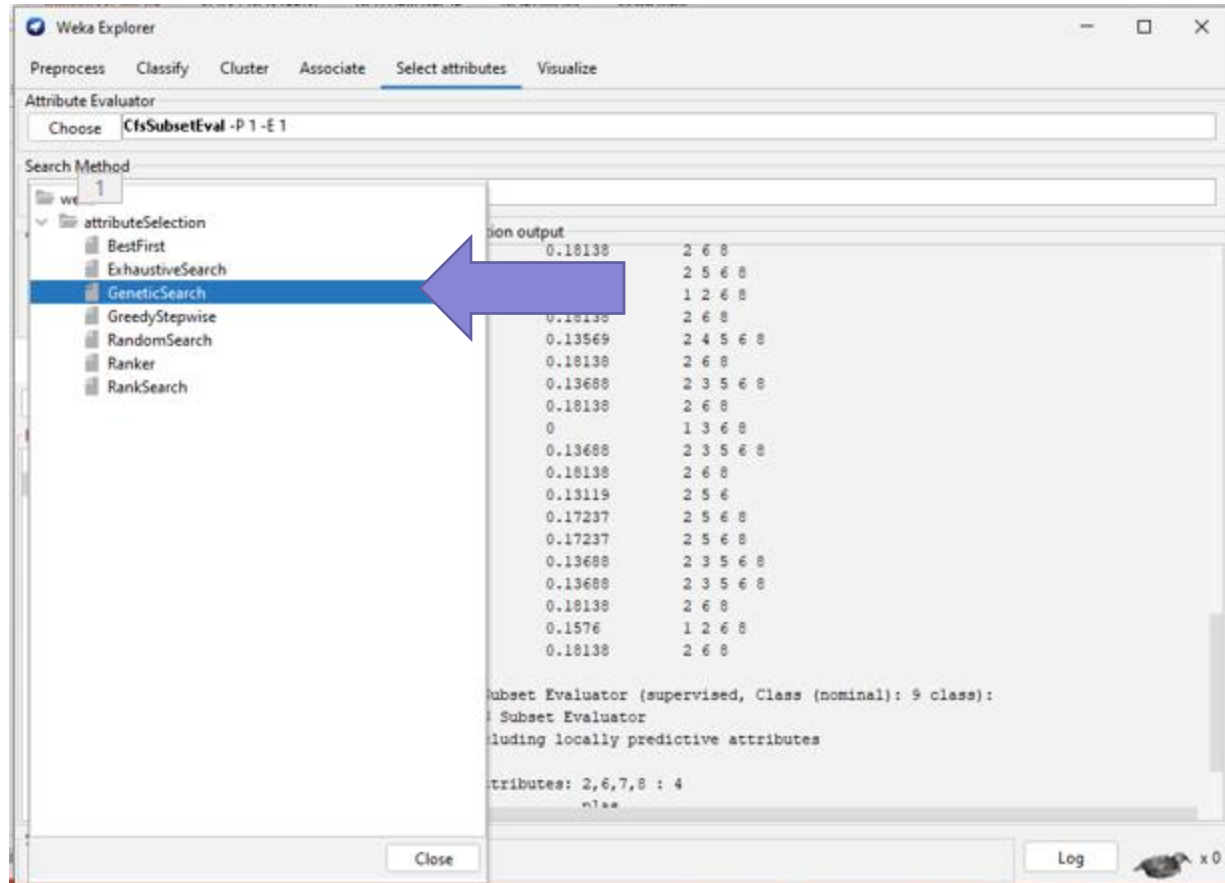
Attribute Subset Evaluator (supervised, Class (nominal): 9 class):
  CFS Subset Evaluator
  Including locally predictive attributes

Selected attributes: 2,6,7,8 : 4
      plas
      mass
      pedi
      age
```

The 'Status' bar at the bottom shows 'OK' and a 'Log' button.

# Genetik Algoritma

## *Weka Uygulaması*



Selected attributes: 2,6,7,8 : 4

plas

mass

pedi

age



# Kaynakça

---

- ❑ALP Selçuk, ÖZ Ersoy, Makine Öğrenmesinde Sınıflandırma Yöntemleri ve R Uygulamaları, Nobel Yayınları, Ankara, 2019.
  - ❑CURA Tunçhan, Modern Sezgisel Teknikler ve Uygulamaları, Papatya Yayıncılık, İstanbul, 2008.
  - ❑ELMAS Çetin, Yapay Zeka Uygulamaları, Seçkin Yayıncılık, Gözden Geçirilmiş 2. Baskı, Ankara, 2010.
  - ❑KARABOĞA Derviş, Yapay Zeka Optimizasyon Algoritmaları, Nobel Kitapevi, 5. Baskı, Ankara, 2017.
  - ❑SATMAN Mehmet Hakan, Genetik Algoritmalar, Türkmen Kitabevi, İstanbul, 2019.
  - ❑YILMAZ Atınç, Yapay Zeka, Kodlab, 4. Baskı, İstanbul, 2018.
  - ❑<https://www.veribilimiokulu.com/genetik-algoritma/>
-