

19.12.2020

YILDIZ TEKNİK ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ



BLM3021 ALGORİTMA ANALİZİ 3.ÖDEV-1.SORU RAPORU

Dinamik Programlama ile Maç Serisi Kazanma Olasılığı Hesaplama

AHMET SAİD SAĞLAM

17011501

KONU

A ve B takımlarının beraberlik ihtimali olmayan maçlar oynaması halinde, N maçlık seri için A takımının maç kazanma olasılığı p ve seriyi kazanmak için kazanması gereken maç sayısı i , B takımının maç kazanma olasılığı $(1-p)$ ve seriyi kazanmak için kazanması gereken maç sayısı j iken, A takımının seriyi kazanma olasılığı olan $P(i, j)$ 'nin dinamik hesaplama yöntemiyle hesaplanması ödevin konusudur.

ÇÖZÜM

Çözüm 1.a)

Öncelikle problemin çözülebilmesi için rekürans bağıntısı yazılmalıdır. Dinamik programlama yöntemi kullanılacağından, ana problem alt problemlere ayrılmalı ve bu alt problemler çözülerek sonuca ulaşılmalıdır. Bu sorudaki ana problem A takımının seriyi kazanma olasılığının hesaplanmasıdır. A takımının başlangıç durumundaki seriyi kazanma olasılığı, serinin bitimine 0 maç kala durumlar ilklendirilerek bu ilklendirmeden sonra adım adım hesaplama yaparak elde edilebilir. Bunun için N maçlık seri baz alınır, $(N/2) + 1$ 'e $(N/2) + 1$ 'lik kare matris bizi çözüme ulaştırır. Matrisin satırları A'nın seriyi kazanmak için alması gereken galibiyet sayısını gösterirken sütunları ise B'nin seriyi kazanmak için alması gereken galibiyet sayısını gösterir. Matrisin son gözü $P(N/2, N/2)$ A takımının başlangıçtaki seriyi kazanma olasılığını verir çünkü başlangıçta iki takım da henüz maç yapmamış ve $N/2$ galibiyete ihtiyaç duymaktadırlar.

A takımı bir galibiyet aldığında seriyi kazanmak için $i - 1$ galibiyet daha almalı iken B takımı da bir galibiyet aldığında seriyi kazanmak için $j - 1$ galibiyet daha almalıdır. Bu durumlar göz önünde bulundurulduğunda ilklendirmeler ve rekürans bağıntısı şu şekilde yapılabilir :

- $j > 0$ olduğu sürece : $P(0, j) = 1$ 'dir. Bunun nedeni A takımının seriyi kazanmak için maç kazanmasına gerek yoksa, seriyi kazanma olasılığının %100 yani 1 olmasıdır.
- $i > 0$ olduğu sürece : $P(i, 0) = 0$ 'dir. Bunun nedeni A takımının seriyi kazanmak için hala maç kazanması gerekirken, B takımının seriyi kazanmak için maç kazanmasına gerek yok ise yani B seriyi çoktan kazanmışsa, A takımının seriyi kazanma ihtimali 0'dır yani yoktur.

Buradaki i ve j değerlerinin ikisinin de aynı anda 0 olması mümkün değildir ve ilklendirilemez. Bu, iki takımın da seriyi kazanmasının kesin olduğu anlamına gelir ve böyle bir olasılık mevcut değildir.

A takımının maç kazanma olasılığı p ve seriyi kazanmak için kazanması gereken maç sayısı i , B takımının maç kazanma olasılığı $(1-p)$ ve seriyi kazanmak için kazanması gereken maç sayısı j olduğunda rekürans bağıntısı olarak :

- i ve $j > 0$ olduğunda : $P(i, j) = p \cdot P(i - 1, j) + (1-p) \cdot P(i, j - 1)$

yazılabilir. A takımının o andaki seriyi kazanma olasılığı iki farklı durumun oluşma olasılığının toplamıdır. Bu durumlardan biri A'nın seriyi kazanması için bir maç eksik kazanma sayısı ile seriyi kazanma olasılığının, A'nın maç kazanma olasılığıyla çarpılmasıdır. Bu sayede A'nın kesinlikle seriyi kazanacağı durumdan itibaren başlangıç durumuna adım adım gidilebilir. Durumlardan diğeri ise B'nin seriyi kazanması için bir maç eksik kazanma sayısı ile seriyi kazanma olasılığının, B'nin maç kazanma olasılığıyla çarpılmasıdır. Bu iki durum toplandığında matrisin ilgili gözünün değeri yani A'nın o gözdeki seriyi kazanma olasılığı elde edilebilir çünkü matrisin o gözdeki değerine iki farklı olasılık sebep olabilir ve bunlar toplanmalıdır.

Çözüm 1.b)

A takımının bir maçı kazanma olasılığının 0.6 olduğu durumda, A takımının 7 maçlık seriyi kazanma olasılığı bir önceki çözümde anlatılan yöntemle matris oluşturarak rahatlıkla hesaplanabilir.

- $p = 0.6$, A takımının bir maç kazanma olasılığı
- $(1-p) = 0.4$, B takımının bir maç kazanma olasılığı
- i = A takımının seriyi kazanması için kazanması gereken maç sayısı
- j = B takımının seriyi kazanması için kazanması gereken maç sayısı
- $P(i, j)$ = A takımının seriyi kazanma olasılığı

Oluşturulacak matris 5×5 boyutunda olmalıdır. $[0,1,2,3,4]$ şeklinde her takım için 0 galibiyetten seriyi kazanma ihtimali olan 4 galibiyete kadar sınırları geniş olmalıdır.

j

19011501

ust

olasılık dışı durum

| | | | | | |
|---|---|------|------|------|------|
| | 0 | 1 | 2 | 3 | 4 |
| 0 | X | 1 | 1 | 1 | 1 |
| 1 | 0 | 0,6 | 0,84 | 0,94 | 0,97 |
| 2 | 0 | 0,36 | 0,65 | 0,82 | 0,91 |
| 3 | 0 | 0,22 | 0,48 | 0,68 | 0,82 |
| 4 | 0 | 0,13 | 0,34 | 0,54 | 0,71 |

ilk satır, B'nin serisi kazanma ihtimali alınması durumu

$P = 0,6$
 $(1-P) = 0,4$

ilk sütun
A'nın serisi kazanma ihtimali alınması durumu

$P(4,4)$

$$1-) P(1,1) = (0,6) + \underbrace{P(0,1)}_{1} + (0,4) * \underbrace{P(1,0)}_{0} = 0,6$$

$$2-) P(1,2) = (0,6) + \underbrace{P(0,2)}_{1} + (0,4) * \underbrace{P(1,1)}_{0,6} = 0,84$$

$$6-) P(2,2) = (0,6) + \underbrace{P(1,2)}_{0,84} + (0,4) * \underbrace{P(2,1)}_{0,36} = 0,648 \approx 0,65$$

$$16-) P(4,4) = (0,6) + \underbrace{P(3,4)}_{0,63} + (0,4) * \underbrace{P(4,3)}_{0,54} = 0,708 \approx \boxed{0,71}$$

↳ Başlangıçta, A takımının serisi kazanma olasılığı

Matris çözüldüğünde A takımının 0.6 maç kazanma olasılığıyla 7 maçlık seriyi kazanma olasılığının 0.71 olduğu görülür.

Çözüm 1.c)

kazanmaOrani Fonksiyonu

```
32 //dinamik programlama yöntemiyle A takımının seriyi kazanma yöntemini hesaplayan fonksiyon
33 double kazanmaOrani(double prob_a, int total_matches) {
34     double prob_b = 1 - prob_a;
35     int i; //A takımının kazandığı maç sayısı // dongu degiskeni
36     int j; //B takımının kazandığı maç sayısı
37     int enough_win = (total_matches / 2) + 1; //seriyi kazanmak için kazanılması gereken maç sayısı
38     double ret_value;
39
40     double **Matches = (double**) malloc((enough_win + 1) * sizeof(double*)); //matris tanımlanır ve matrisin satırları için yer açılır
41     //matris, n sütunları için yer açılır
42     for(i = 0; i <= enough_win; i++) {
43         Matches[i] = (double*) malloc((enough_win + 1) * sizeof(double));
44     }
45
46     //matris 1. satır ve sütundan itibaren gezilerek değerler hesaplanır
47     for (i = 1; i <= enough_win; i++) {
48         Matches[i][0] = 0; //matrisin ilk sütunu ilklendirilir
49         for(j = 1; j <= enough_win; j++) {
50             Matches[i][j] = 1; //matrisin ilk satırı ilklendirilir
51             Matches[i][j] = (prob_a * Matches[i-1][j]) + (prob_b * Matches[i][j-1]); //matrisin ilgili gözü hesaplanır
52         }
53     }
54
55     printf("\nCozum Matrisi : \n\n");
56     printDoubleMatrix(Matches, (enough_win + 1), (enough_win + 1)); //matris ekrana yazdırılır
57
58     ret_value = Matches[enough_win][enough_win]; //donus degeri degiskende saklanır
59     //dinamik olarak acilan yer free edilir
60     for(i = 0; i <= enough_win; i++) {
61         free(Matches[i]);
62     }
63     free(Matches);
64
65     return ret_value; //matrisin son gözü yani sonuç döndürülür
66 }
```

Kazanma olasılığını hesaplamak için C dilinde kazanmaOrani adlı bir fonksiyon yazılmıştır. Bu fonksiyon içerisine A takımının kazanma ihtimali ile birlikte oynanacak toplam maç sayısını alır ve A takımının seriyi kazanma ihtimalini döndürür. A takımının seriyi kazanma olasılığı, 1.a çözümünde rekürans bağıntısı anlatılan ve 1.b çözümünde gösterilen matrisin C dilinde implementasyonunun yapılmasıyla hesaplanmıştır.

PROGRAM ÇIKTILARI

```
E:\YTU\BLM3-GUZ\alg analizi\odev-3\17011501_1.exe
Serideki toplam mac sayisini giriniz : 7
A takiminin kazanma ihtimalini giriniz : 0.6
Cozum Matrisi :
0.000  1.000  1.000  1.000  1.000
0.000  0.600  0.840  0.936  0.974
0.000  0.360  0.648  0.821  0.913
0.000  0.216  0.475  0.683  0.821
0.000  0.130  0.337  0.544  0.710

A takiminin seriyi kazanma olasiligi : 0.710
Yeni hesaplama yapmak icin 1'e, cikmak icin 0'a basiniz...
```

```
E:\YTU\BLM3-GUZ\alg analizi\odev-3\17011501_1.exe
Serideki toplam mac sayisini giriniz : 7
A takiminin kazanma ihtimalini giriniz : 0.4
Cozum Matrisi :
0.000  1.000  1.000  1.000  1.000
0.000  0.400  0.640  0.784  0.870
0.000  0.160  0.352  0.525  0.663
0.000  0.064  0.179  0.317  0.456
0.000  0.026  0.087  0.179  0.290

A takiminin seriyi kazanma olasiligi : 0.290
Yeni hesaplama yapmak icin 1'e, cikmak icin 0'a basiniz...
```

Yeni hesaplama yapmak için 1'e, çıkmak için 0'a basınız...

1

Serideki toplam mac sayısını giriniz : 11

A takımının kazanma ihtimalini giriniz : 0.3

Cozum Matrisi :

| | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|
| 0.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 | 1.000 |
| 0.000 | 0.300 | 0.510 | 0.657 | 0.760 | 0.832 | 0.882 |
| 0.000 | 0.090 | 0.216 | 0.348 | 0.472 | 0.580 | 0.671 |
| 0.000 | 0.027 | 0.084 | 0.163 | 0.256 | 0.353 | 0.448 |
| 0.000 | 0.008 | 0.031 | 0.070 | 0.126 | 0.194 | 0.270 |
| 0.000 | 0.002 | 0.011 | 0.029 | 0.058 | 0.099 | 0.150 |
| 0.000 | 0.001 | 0.004 | 0.011 | 0.025 | 0.047 | 0.078 |

A takımının seriyi kazanma olasılığı : 0.078

Yeni hesaplama yapmak için 1'e, çıkmak için 0'a basınız...

0

Process exited after 83.05 seconds with return value 0

Press any key to continue . . .

SOURCE CODE

```
1.  /*
2.  @file
3.  BLM3021 2020-2021 GUZ ODEV-3 / Soru-1
4.  Bu programda dinamik programlama yöntemi kullanılarak iki takım arasında geçen maç s
   erisini, takımlardan biri olan A takımının kazanma olasılığı hesaplanmıştır.
5.  Maçlarda bereberlik söz konusu değildir.
6.
7.  @author
8.  İsim: Ahmet Said SAĞLAM
9.  Öğrenci No: 17011501
10. Tarih: 19.12.2020
11. E-Mail: l1117501@std.yildiz.edu.tr
12. Compiler: TDM-GCC 4.9.2 64 bit-Release
13. IDE: DEV-C++ (version 5.11)
14. İşletim Sistemi: Windows 10 Pro 64 bit
15. */
16.
17. #include <stdio.h>
18. #include <math.h>
19. #include <stdlib.h>
20.
21. //icerisine aldığı double tipindeki matrisi yazdırır
22. void printDoubleMatrix(double **matrix, int row_count, int column_count) {
23.     int i,j;
24.     for (i=0; i < row_count; i++) {
25.         for(j=0; j < column_count; j++) {
26.             printf("%.3f\t", matrix[i][j]);
27.         }
28.         printf("\n");
29.     }
30. }
31.
32. //dinamik programlama yöntemiyle A takımının seriyi kazanma yöntemini hesaplayan fon
   ksion
33. double kazanmaOrani(double prob_a, int total_matches) {
34.     double prob_b = 1 - prob_a; //B takımının 1 macı kazanma olasılığı
35.     int i; //A takımının kazandığı mac sayısı // dongu degiskeni
36.     int j; //B takımının kazandığı mac sayısı
37.     int enough_win = (total_matches / 2) + 1; //seriyi kazanmak icin kazanılması g
   ereken mac sayısı
38.     double ret_value;
39.
40.     double **Matches = (double**) malloc((enough_win + 1) * sizeof(double*)); //ma
   tris tanimlanir ve matrisin satirlari icin yer acilir
41.     //matris,n sutunlari icin yer acilir
42.     for(i = 0; i <= enough_win; i++) {
43.         Matches[i] = (double*) malloc((enough_win + 1) * sizeof(double));
44.     }
45.
46.     //matris 1. satır ve sütundan itibaren gezilerek değerler hesaplanır
47.     for (i = 1; i <= enough_win; i++) {
48.         Matches[i][0] = 0; //matrisin ilk sutunu ilklendirilir
49.         for(j = 1; j <= enough_win; j++) {
50.             Matches[0][j] = 1; //matrisin ilk satırı ilklendirilir
51.             Matches[i][j] = (prob_a * Matches[i-1][j]) + (prob_b * Matches[i][j-
   1]); //matrisin ilgili gözü hesaplanır
52.         }
53.     }
54.
55.     printf("\nCozum Matrisi : \n\n");
```



```
56.     printDoubleMatrix(Matches, (enough_win + 1), (enough_win + 1));    //matris ekr
ana yazdirilir
57.
58.     ret_value = Matches[enough_win][enough_win];    //donus degeri degiskende saklan
ir
59.
60.     //dinamik olarak acilan yer free edilir
61.     for(i = 0; i <= enough_win; i++) {
62.         free(Matches[i]);
63.     }
64.     free(Matches);
65.
66.     return ret_value;    //matrisin son gözü yani sonuç döndürülür
67. }
68.
69.
70.
71. int main() {
72.     double prob_a; //A takımının bir maci kazanma olasiligi
73.     int total_matches; //serideki toplam mac sayisi
74.     int cont = 1;
75.     //kullanici istedikce donen while dongusu
76.     while(cont) {
77.         printf("Serideki toplam mac sayisini giriniz : ");
78.         scanf("%d",&total_matches);
79.         printf("\nA takiminin kazanma ihtimalini giriniz : ");
80.         scanf("%lf",&prob_a);
81.         printf("\n\nA takiminin seriyi kazanma olasiligi : %.3lf\n\n",kazanmaOrani(p
rob_a,total_matches));
82.         printf("Yeni hesaplama yapmak icin 1'e, cikmak icin 0'a basiniz...\n");
83.         scanf("%d",&cont);
84.         printf("\n");
85.     }
86.     return 0;
87. }
```