5강 과제

학번 : 201685-010100

이름 : 지용기

5강에서 Clustering 방법들을 개괄적으로 소개하였다. 그 중 Decision Tree (의사결정나무) 는 최근 벌어진 바둑경기에서 9단 이세돌을 이긴 알파고(AlphaGo)의 핵심적인 알고리즘 중 하나이다. 첨부된 PPT 파일을 통해 이 의사결정나무가 어떻게 작동하는지 그 원리를 Tenis playing의 예를 통해 확인하고 간략히 기술하도록 하라.

데이터의 생성

In [11]:

```
Outlook <- c( "Sunny", "Sunny", "Overcast", "Rain", "Rain", "Rain", "Overcast",
              "Sunny", "Sunny", "Rain", "Sunny", "Overcast", "Overcast", "Rain"  )
Temp <- c("Hot", "Hot", "Hot", "Mild", "Cool", "Cool", "Cool",
          "Mild", "Cool", "Mild", "Mild", "Mild", "Hot", "Mild" )
Humidity <- c("High", "High", "High", "High", "Normal", "Normal", "Normal", "High",
               "Normal", "Normal", "Normal", "High", "Normal", "High" )
Wind <- c("Weak", "Strong", "Weak", "Weak", "Weak", "Strong", "Weak",
          "Weak", "Weak", "Strong", "Strong", "Strong", "Weak", "Strong")
PayTennis <- c("No", "No", "Yes", "Yes", "Yes", "No", "Yes", "No", "Yes", "Yes", "Yes", "Y
es", "Yes", "No" )
training.example <- data.frame(Outlook, Temp, Humidity, Wind, PayTennis)
```

In [12]:

```
training.example
```

Out[12]:

	Outlook	Temp	Humidity	Wind	PayTennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Weak	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Strong	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No

In [13]:

```
summary( training.example )
```

Out[13]:

```
    Outlook     Temp     Humidity     Wind    PayTennis
 Overcast:4   Cool:4   High  :7   Strong:6   No :5
 Rain    :5   Hot :4   Normal:7   Weak  :8   Yes:9
 Sunny   :5   Mild:6
```

attach(training.example)

루트 노드의 선택

In [4]:

```
tab.Outook <- table(Outlook, PayTennis)
tab.Temp <- table(Temp, PayTennis)
tab.Humidity <- table(Humidity, PayTennis)
tab.Wind <- table(Wind, PayTennis)
```

In [5]:

```
tab.Outook
prop.table(tab.Outook,1)
```

Out[5]:

```
          PayTennis
Outlook    No Yes
  Overcast  0   4
  Rain      2   3
  Sunny     3   2
```

Out[5]:

```
          PayTennis
Outlook     No Yes
  Overcast 0.0 1.0
  Rain     0.4 0.6
  Sunny    0.6 0.4
```

In [38]:

```
entropy.Outook <- - log2(1)  - 0.4 * log2(0.4) - 0.6  * log2( 0.6 )  - 0.4  *  log2( 0.4 )
entropy.Outook
```

Out[38]:

1.49972183240961

In [33]:

```
tab.Temp
prop.table(tab.Temp, 1)
```

Out[33]:

```
      PayTennis
Temp   No Yes
  Cool  1   3
  Hot   2   2
  Mild  2   4
```

Out[33]:

```
      PayTennis
Temp          No       Yes
  Cool 0.2500000 0.7500000
  Hot  0.5000000 0.5000000
  Mild 0.3333333 0.6666667
```

In [40]:

```
entropy.Temp <- - 0.25*log2(0.25) - 0.75*log2(0.75) - 0.5*log2( 0.5 ) - 0.5*log2( 0.5 ) - 0.33*log2( 0.33 ) - 0.66*log2( 0.66 )
entropy.Temp
```

Out[40]:

2.73474557417124

In [41]:

```
tab.Humidity
prop.table(tab.Humidity, 1)
```

Out[41]:

```
        PayTennis
Humidity No Yes
  High    4   3
  Normal  1   6
```

Out[41]:

```
        PayTennis
Humidity        No       Yes
  High   0.5714286 0.4285714
  Normal 0.1428571 0.8571429
```

In [42]:

```
entropy.Humidity <- - 0.57*log2(0.57) - 0.42*log2(0.42) - 0.14*log2(0.14) - 0.85*log2(0.85)
entropy.Humidity
```

Out[42]:

1.58430264530786

In [43]:

```
tab.Wind
prop.table(tab.Wind, 1)
```

Out[43]:

```
        PayTennis
Wind     No Yes
  Strong  3   3
  Weak    2   6
```

Out[43]:

```
        PayTennis
Wind       No  Yes
  Strong 0.50 0.50
  Weak   0.25 0.75
```

In [44]:

```
entropy.Wind <- - 0.5*log2(0.5) - 0.5*log2(0.5) - 0.25*log2(0.25) - 0.75*log2(0.75)
entropy.Wind
```

Out[44]:

1.81127812445913

In [45]:

```
entropy.Outook
entropy.Temp
entropy.Humidity
entropy.Wind
```

Out[45]:

1.49972183240961

Out[45]:

2.73474557417124

Out[45]:

1.58430264530786

Out[45]:

1.81127812445913

- 4개의 변수중에서 엔트로피가 가장 낮은 변수는 Outook 이므로 루트가 됨.

***Outook* 변수중에서 *Sunny*로 연결되는 변수의 선택**

In [22]:

```
training.example.sunny <- training.example[ Outlook == "Sunny",  ]
training.example.sunny

tab.sunny.Temp <- with(training.example.sunny,  table(Temp, PayTennis) )
prop.table(tab.sunny.Temp, 1)
entropy.sunny.Temp <-  - log2( 1 )- log2( 1 ) - 0.5*log2( 0.5 ) - 0.5*log2( 0.5 )
entropy.sunny.Temp
```

Out[22]:

	Outlook	Temp	Humidity	Wind	PayTennis
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes

Out[22]:

```
      PayTennis
Temp     No Yes
  Cool 0.0 1.0
  Hot  1.0 0.0
  Mild 0.5 0.5
```

Out[22]:

1

In [24]:

```
tab.sunny.Humidity <- with(training.example.sunny,  table(Humidity, PayTennis) )
prop.table(tab.sunny.Humidity, 1)
entropy.sunny.Humidity <-  - log2( 1 )- log2( 1 )
entropy.sunny.Humidity
```

Out[24]:

```
        PayTennis
Humidity No Yes
  High    1   0
  Normal  0   1
```

Out[24]:

0

In [26]:

```
tab.sunny.Wind <- with(training.example.sunny,  table(Wind, PayTennis) )
prop.table(tab.sunny.Wind, 1)
entropy.sunny.Wind <-  - 0.5*log2( 0.5 ) - 0.5*log2( 0.5 ) - 0.66*log2( 0.33 )
entropy.sunny.Wind
```

Out[26]:

```
        PayTennis
Wind            No       Yes
  Strong 0.5000000 0.5000000
  Weak   0.6666667 0.3333333
```

Out[26]:

2.05564496647474

- Outook 변수중에서 Sunny일때는 하위 노드로 Humidity 를 갖을때 가장 엔트로피가 낮고, PayTennis를 완벽히 구분이 됩니다.

Outook 변수중에서 *Overcast*일때

In [27]:

```
training.example.Overcast <- training.example[ Outlook == "Overcast",  ]
training.example.Overcast
```

Out[27]:

	Outlook	Temp	Humidity	Wind	PayTennis
3	Overcast	Hot	High	Weak	Yes
7	Overcast	Cool	Normal	Weak	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes

- Overcast일때는 PayTennis가 모두 Yes로 구분이 되기 때문에 하위 노드가 필요가 없음.

Outook 변수중에서 *Rain*일때

In [4]:

```
training.example.Rain <- training.example[ Outlook == "Rain",  ]
training.example.Rain
```

Out[4]:

	Outlook	Temp	Humidity	Wind	PayTennis
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
10	Rain	Mild	Normal	Strong	Yes
14	Rain	Mild	High	Strong	No

In [6]:

```
tab.Rain.Temp <- with(training.example.Rain,  table(Temp, PayTennis) )
prop.table(tab.Rain.Temp, 1)
entropy.Rain.Temp <-  - 0.5*log2( 0.5 )- log2( 0.5 ) - 0.33*log2( 0.33 ) - 0.66*log2( 0.66
)
entropy.Rain.Temp
```

Out[6]:

```
      PayTennis
Temp          No       Yes
  Cool 0.5000000 0.5000000
  Hot
  Mild 0.3333333 0.6666667
```

Out[6]:

2.42346744971211

In [7]:

```
tab.Rain.Humidity <- with(training.example.Rain,  table(Humidity, PayTennis) )
prop.table(tab.Rain.Humidity, 1)
entropy.Rain.Humidity <-  - 0.5*log2( 0.5 )- log2( 0.5 ) - 0.33*log2( 0.33 ) - 0.66*log2(
0.66 )
entropy.Rain.Humidity
```

Out[7]:

```
        PayTennis
Humidity        No       Yes
  High   0.5000000 0.5000000
  Normal 0.3333333 0.6666667
```

Out[7]:

2.42346744971211

In [9]:

```
tab.Rain.Wind <- with(training.example.Rain,  table(Wind, PayTennis) )
prop.table(tab.Rain.Wind, 1)
entropy.Rain.Wind <-  - 0.66*log2( 0.66 )- 0.33*log2( 0.33 ) - log2( 1 )
entropy.Rain.Wind
```

Out[9]:

```
        PayTennis
Wind            No       Yes
  Strong 0.6666667 0.3333333
  Weak   0.0000000 1.0000000
```

Out[9]:

0.923467449712108

- Outook 변수중에서 Rain일때는 하위 노드로 Wind 를 갖을때 가장 엔트로피가 낮습니다.
- 그래서 하위노드로 Wind로 하며 Wind는 완벽히 PayTennis을 완벽히 구분을 못하므로, 하위 노드를 갖을 수 있습니다.

In [ ]:

```

```