

빅데이터분석응용실습 HW10

- 원형 그래프 출력 및 네트워크 분석지표 계산하기-

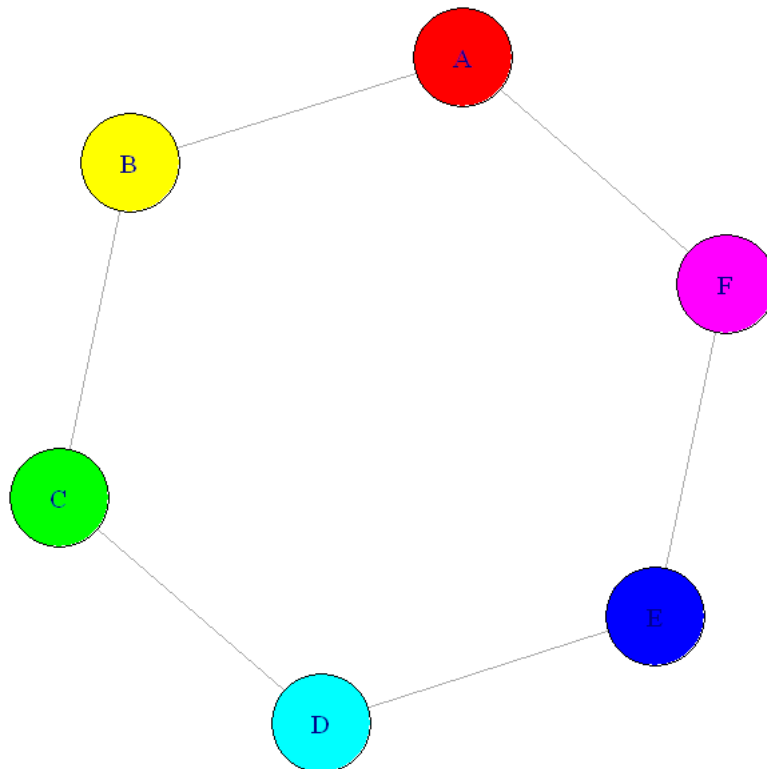
영어영문학과(휴먼ICT) 201614010 박도준

1. 원형 그래프 출력

- 출력 코드

```
G.ring <- make_ring(6, directed=FALSE, circular=TRUE) %>%  
  set_vertex_attr("name", value=c('A', 'B', 'C', 'D', 'E', 'F'))  
plot(G.ring, vertex.color=rainbow(6), vertex.size=30)
```

- 출력 결과



2. 원형 그래프의 연결 정도

2-1. 연결 정도 중심성 (비정규형)

- 출력 코드

```
degree(G.ring, normalized=FALSE)
```

- 출력 결과

A 2

B 2

C 2

D 2

E 2

F 2

2-2. 연결 정도 중심성 (정규형)

- 출력 코드

```
degree(G.ring, normalized=TRUE)
```

- 출력 결과

A 0.4

B 0.4

C 0.4

D 0.4

E 0.4

F 0.4

2-3. 연결 정도 중심화

- 출력 코드

```
CD <- centralization.degree(G.ring, normalized=FALSE)
CD$centralization
```

- 출력 결과

0

2-4. 정규화 된 연결정도 중심화

- 출력 코드

```
Tmax <- centralization.degree.tmax(G.ring)
CD$centralization / Tmax
```

- 출력 결과

0

3. 원형 그래프의 근접

3-1. 근접 중심성 (비정규형)

- 출력 코드

```
closeness(G.ring, normalized=False)
```

- 출력 결과

```
A  0.111111111111111
B  0.111111111111111
C  0.111111111111111
D  0.111111111111111
E  0.111111111111111
F  0.111111111111111
```

3-2. 근접 중심성 (정규형)

- 출력 코드

```
closeness(G.ring, normalized=True)
```

- 출력 결과

```
A  0.555555555555556
B  0.555555555555556
C  0.555555555555556
D  0.555555555555556
E  0.555555555555556
F  0.555555555555556
```

3-3. 근접 중심화

- 출력 코드

```
CC <- centralization.closeness(G.ring, normalized=FALSE)
CC$centralization
```

- 출력 결과

4.44089209850063e-16

3-4. 정규화 된 근접 중심화

- 출력 코드

```
CC$centralization / (6-1)
```

- 출력 결과

8.88178419700125e-17

4. 원형 그래프의 중개

4-1. 중개 중심성 (비정규화)

- 출력 코드

```
betweenness(G.ring, normalized=FALSE)
```

- 출력 결과

A 2

B 2

C 2

D 2

E 2

F 2

4-2. 중개 중심성 (정규화)

- 출력 코드

```
betweenness(G.ring, normalized=TRUE)
```

- 출력 결과

A 0.2

B 0.2

C 0.2

D 0.2

E 0.2

F 0.2

4-3. 중개 중심화

- 출력 코드

```
CB <- centralization.betweenness(G.ring, normalized=FALSE)  
CB$centralization
```

- 출력 결과

0

4-4. 정규화 된 중개 중심화

- 출력 코드

```
CB$centralization / CB$theoretical_max
```

- 출력 결과

0

5. 원형 그래프의 밀도

- 출력 코드

```
graph.density(G.ring)
```

- 출력 결과

0.4

6. 원형 그래프의 평균 거리

- 출력 코드

```
average.path.length(G.ring)
```

- 출력 결과

1.8