

# 선수 유형 추천시스템

## Recommender System

Factor Analysis & Hierarchical Clustering을  
이용한 독자적 알고리즘 개발



Team 명륜동

# INDEX

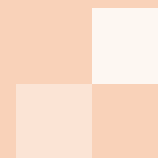


요인분석, 계층군집분석을 활용한 선수유형 추천 시스템

선수 유형 추천 시스템

# 1 주제 소개 & 선정배경

- 1) KBL 자체 선수 추천 시스템
- 2) NBA 유사 선수 추천 시스템



## 01 주제 선정 배경

# 주제 선정 배경

(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천



올해로 스무살 KBL, 향후 흥행을 위해서 더 필요한 것은?



## 01 주제 선정 배경

# 주제 선정 배경

농구에 관한 국민들 관심 증대 !

(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천



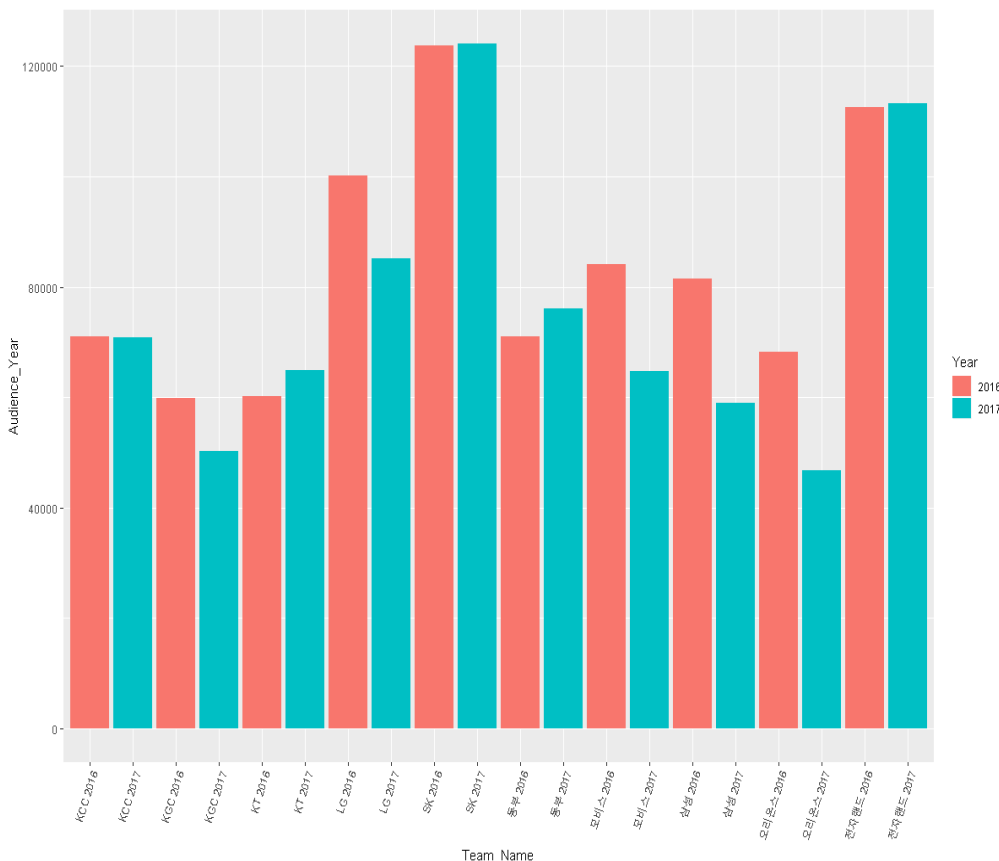
## 01 주제 선정 배경

# 주제 선정 배경

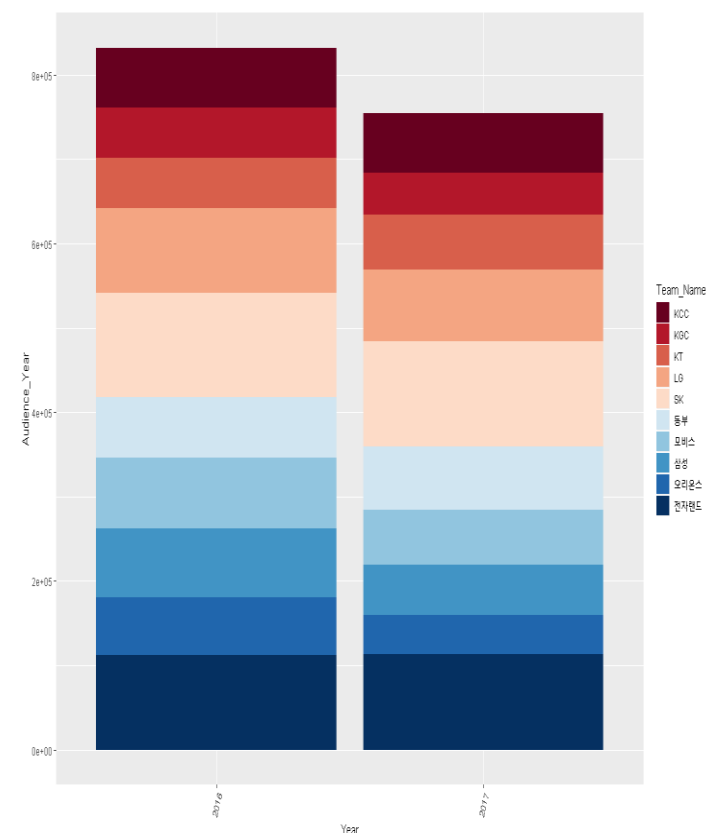
(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천

<구단 별 16,17 시즌 관객 수>



<kbl 16,17 시즌 총 관객 수>



하지만, 정작감소하는 KBL의 관중 수

# 주제 선정 배경

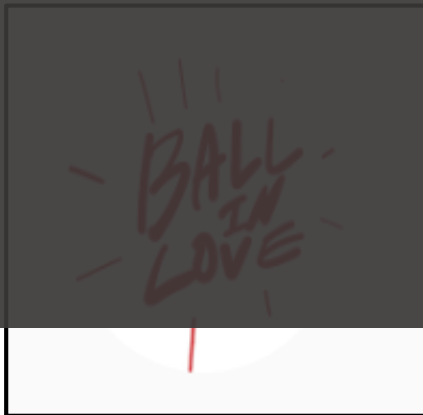
(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천

국민들의  
“농구 관심”



어떻게 하면 **KBL 흥행 성공**으로  
연결시킬 수 있을까 ?





01 주제 선정  
배경

# 주제 선정 배경

KBL **예비** **관중** 유형을 3가지로 구분해보자 !

(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천

1)



농구에 전혀  
無관심한 타입

2)



농구는 좋아하지만 **KBL**  
을 전혀 모르는 타입

3)



농구는 좋아하지만 **NBA**  
를 더 좋아하는 타입



# 주제 선정 배경

## 01 주제 선정 배경

(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천

1)



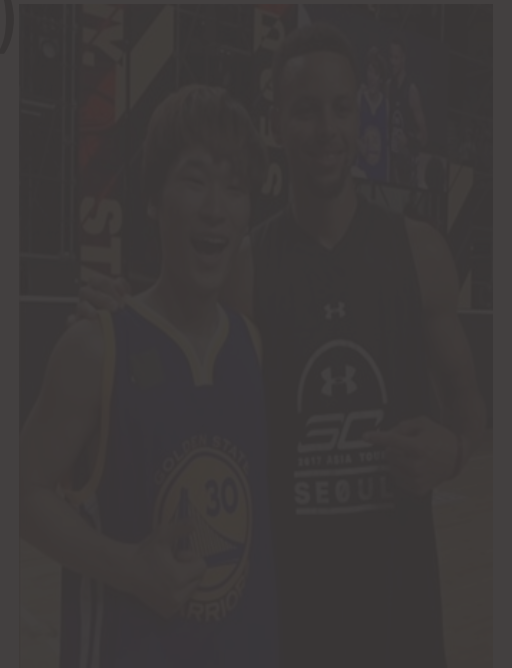
농구에 전혀  
無 관심한 타입

2)



농구는 좋아하지만 **KBL**  
을 전혀 모르는 타입

3)



농구는 좋아하지만 NBA  
를 더 좋아하는 타입

## 01 주제 선정 배경

# 주제 선정 배경

(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천

1)



농구에 전혀  
無 관심한 타입

2)



농구는 좋아하지만 **KBL**  
을 전혀 모르는 타입

3)

간단한 설문지를 통해  
취향에 맞는 선수를  
추천 해준다면 어떨까?

농구는 좋아하지만 NBA  
를 더 좋아하는 타입

## 01 주제 선정 배경

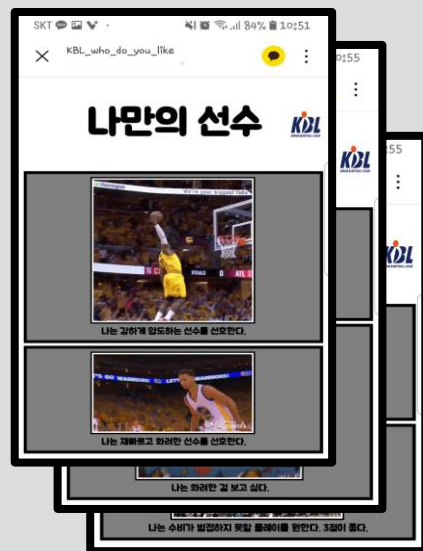
(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천

# 주제 선정 배경

## 선수 추천 알고리즘 소개

1



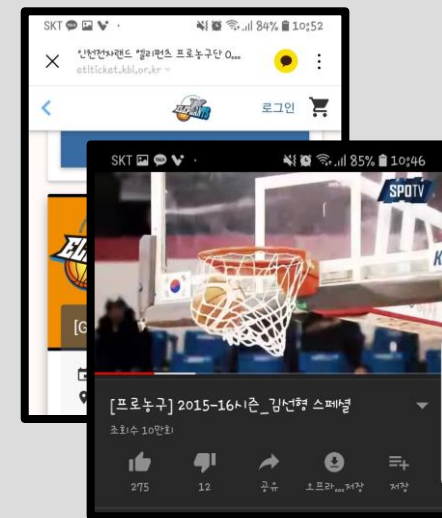
독자적 알고리즘을 이용해  
생성한 질문지 제시

2



개인에게 가장  
적합한 선수 추천

3



해당 선수의 경기 예매처 및  
하이라이트 영상 링크 제공

## 01 주제 선정 배경

# 주제 선정 배경

(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천



3)



농구는 좋아하지만  NBA  
를 더 좋아하는 타입



# 주제 선정 배경

## 01 주제 선정 배경

(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천



3)



농구는 좋아하지만 NBA  
를 더 좋아하는 타입



## 01 주제 선정 배경

(1)  
KBL 자체 선수  
추천

(2)  
NBA 유사 선수  
추천

# 주제 선정 배경

1)

2)

KBL?

좋아하는 NBA 선수들 유형과

■ ■ ■ ■ ■ ■ ■ ■  
유사한 유형의 선수를 추천해준다면 ?

농구에 전혀  
無 관심한 타입

농구는 좋아하지만 KBL  
을 전혀 모르는 타입

3)



농구는 좋아하지만  NBA  
를 더 좋아하는 타입

# 주제 선정 배경

## 01 주제 선정 배경

- (1)  
KBL 자체 선수  
추천
- (2)  
NBA 유사 선수  
추천



농구는 좋아하지만 NBA  
를 더 좋아하는 타입

---

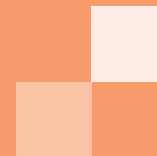
선수 유형 추천 시스템

---

# 2

## KBL 자체 선수 추천

- 1) 데이터 전처리
- 2) Factor Analysis
- 3) Hierarchical clustering



# Step1. 변수삭제 및 파생변수 생성

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

fb(속공), idf(부정수비)

```
> table(player$fb)      > table(player$idf)

 0              0
15086          15086
```

모든 값들이 동일

↓  
변수삭제!

자유투, 2점슛, 3점슛, 덩크

fg	fg_a	fg_ratio
20	100	20%
15	50	30%

$$\frac{\text{성공횟수}}{\text{전체 시도 횟수}}$$

↓  
각각 ratio 변수 생성!

# Step1. 변수삭제 및 파생변수 생성

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

<http://bookyoon.dothome.co.kr/g5/glossary.php>

KBL레퍼런스

선수스탯 팀스탯 스탯랭킹 스탯설명 자유게시판 🔍 세부검색 ▾

Player Efficiency Rating (PER)

True Shooting Percentage(TS%), Effective Field Goal Percentage(eFG%)

AST%, TOV%, BLK%, REB%, STL%

100포제션당 해당 스탯이 얼마나 나오는가에 대한 비율 스탯.  
출장시간이 달라 스탯이 다른 선수들을 동일 조건에 놓고 비교해보기 위한 스탯이다.  
공식

AST% :  $100 * AST / (((MP / (Tm MP / 5)) * Tm FG) - FG)$

BLK% :  $100 * (BLK * (Tm MP / 5)) / (MP * (Opp FGA - Opp 3PA))$

TOV% :  $100 * TOV / (FGA + 0.44 * FTA + TOV)$

REB% :  $100 * (REB * (Tm MP / 5)) / (MP * (Tm REB + Opp REB))$

STL% :  $100 * (STL * (Tm MP / 5)) / (MP * Opp Poss)$

공헌도

제공된 데이터와  
KBL레퍼런스 변수명을 맞추고



Group\_by를 통해  
Opp, Tm 변수 생성  
(팀단위, 게임당 수치)



KBL 레퍼런스에서 제공하는  
공식에 맞추어 파생변수 생성





- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

# Step1. 변수삭제 및 파생변수 생성

## 생성된 파생변수의 의미

변수명	설명
TS%	True Shooting% - 선수의 슈팅 효율성 지수
EFG%	Effective Field Goal % - 2점슛보다 3점슛에 높은 가중치를 부여한 효율적 슈팅 지수
AST%	해당 선수가 코트에 있는동안 만들어진 필드골이 해당 선수에 의해 어시스트 될 확률 추정치
BLK%	해당 선수가 코트에 있는 동안 상대팀이 시도한 2점슛이 해당 선수에 의해 블락될 확률 추정치
TOV%	플레이 100번 당 턴오버를 할 횟수에 대한 추정치
REB%	해당 선수가 코트에 있는 동안 발생한 리바운드가 해당 선수에 의해 리바운드 될 확률 추정치



출장시간이 다른 선수들을 동일 조건에 놓고 비교하기 위한 **파생변수** 생성완료!

## Step2. Data Reshape

(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

	play_min	play_sec	fg	fg_a	ft	ft_a	threep	threep_a	dk	dk_a
1	7	53	0	0	0	0	0	0	0	0
2	14	45	3	6	0	1	1	1	1	1
3	10	50	3	6	0	0	1	1	0	0
4	35	48	2	5	1	2	2	5	0	0
5	24	15	1	4	0	0	1	1	0	0
6	0	0	0	0	0	0	0	0	0	0
7	31	58	13	25	4	6	0	2	1	1
8	24	33	0	0	0	0	2	4	0	0
9	3	14	0	0	0	0	0	0	0	0
10	13	37	0	1	0	0	0	2	0	0
11	30	36	6	11	2	2	1	4	2	2
12	28	29	1	3	2	2	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0
14	16	35	0	0	1	2	1	3	0	0
15	22	51	0	1	0	0	1	2	0	0
16	0	0	0	0	0	0	0	0	0	0
17	13	40	0	0	0	0	0	1	0	0
18	0	0	0	0	0	0	0	0	0	0
19	17	9	2	4	0	0	0	1	0	0
20	7	36	1	2	2	2	0	0	0	0



선수마다 출전시간이 다른데 비교할 수 있을까?

## Step2. Data Reshape

(1)  
데이터 전처리

### 1. 최소 경기 수 설정

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

player_no	game_no	fg	fg_a
1	1	10	30
1	2	15	20
1	3	5	10
2	1	10	20

- player\_no : 선수코드
- game\_no : 경기번호
- fg : 2점슛성공
- fg\_a : 2점슛시도



player_no	avg_fg	avg_fg_a
1	10	20
2	10	20

한 번 출전한 선수와  
세 번 출전한 선수의 성공률이 같음!

투입경기 수가 적은 선수를 포함할 시 **편향된 결과** 도출 위험!

## Step2. Data Reshape

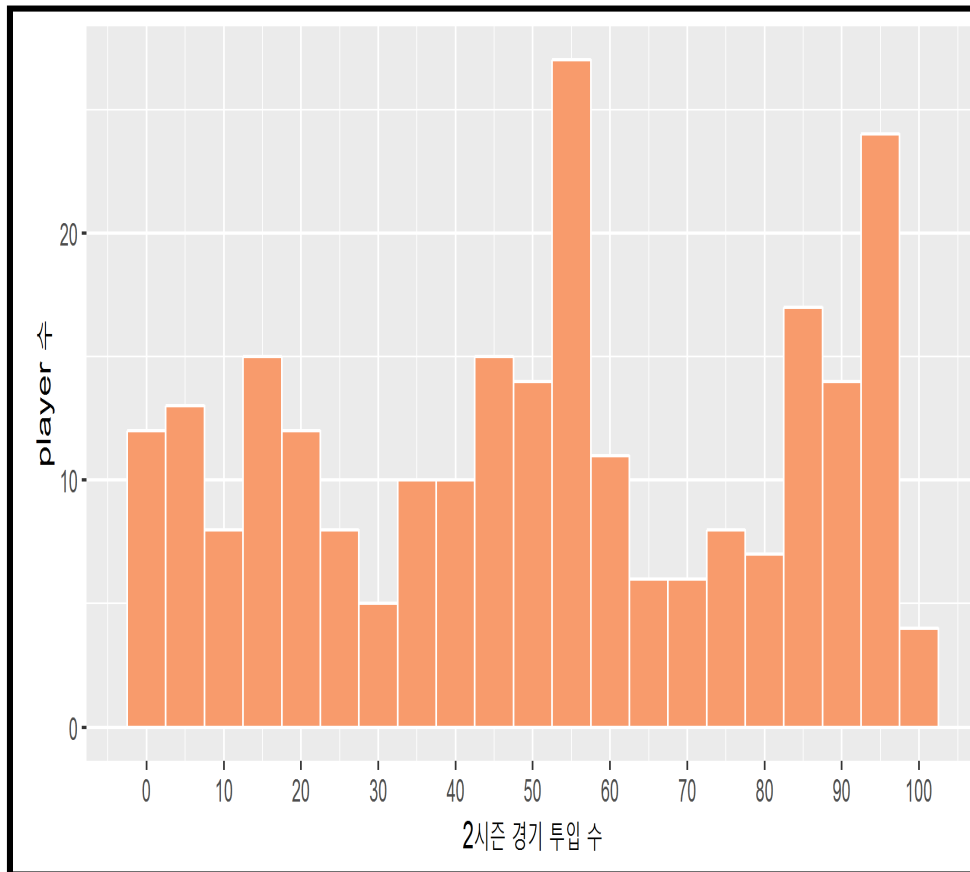
### 1. 최소 경기 수 설정

(1)  
데이터 전처리

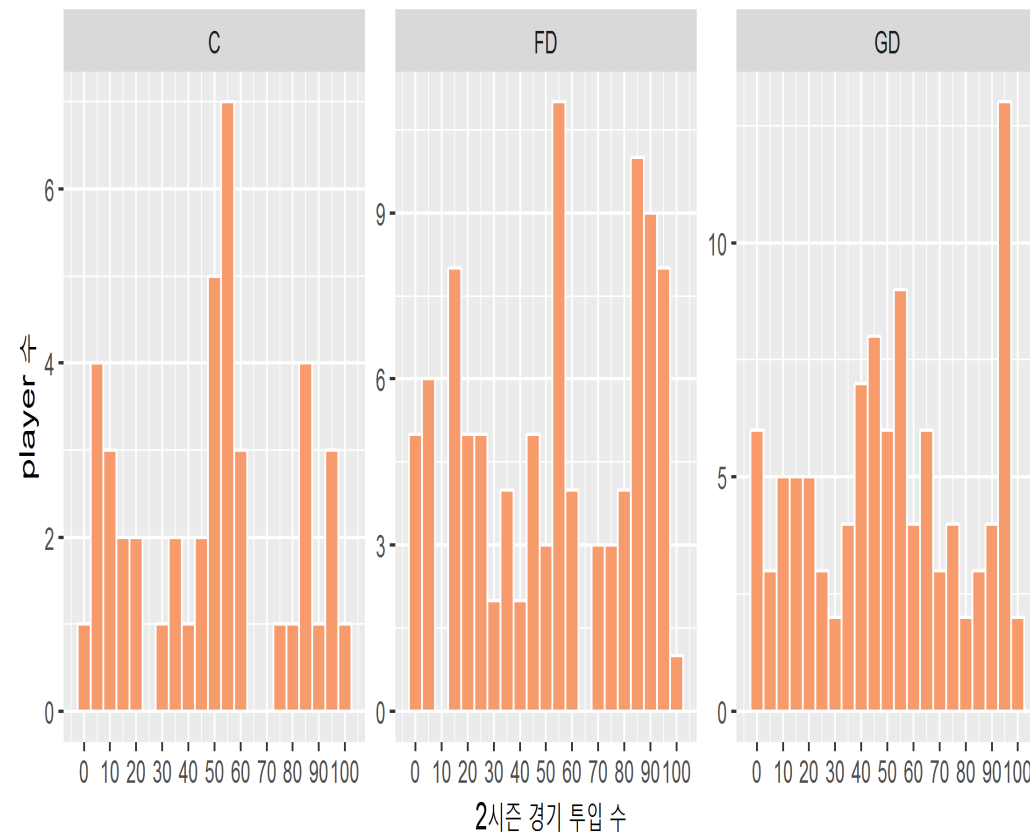
(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

<전체 선수 별 투입 경기 수>



<포지션 별 투입 경기 수>



2시즌 총 투입 경기가 **10경기** 미만인 데이터 삭제!

## Step2. Data Reshape

(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

### 2. 최소 경기 시간 설정

player_no	play_time	fg	fg_a	fg_ratio
A	100	30	60	50%
B	10	1	1	100%

두 선수의 슈팅 성공률은 100%인 B가 더 높다

하지만, B선수는 1개의 슈팅 중 1개를 성공

즉, 선수 마다 플레이 시간이 다르므로 절대적인 수치 비교 불가



따라서, 한 경기 내 플레이 시간이 100초 이하 데이터 삭제!



## Step2. Data Reshape

(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

### 3. 동일 시간 단위로 변환

player_no	play_time	fg	fg_a
A	100	30	60
B	50	20	40



player_no	play_time	fg	fg_a
A	100	30/100	60/100
B	50	20/50	40/50

동일 시간 단위로 normalizing!



## Step2. Data Reshape

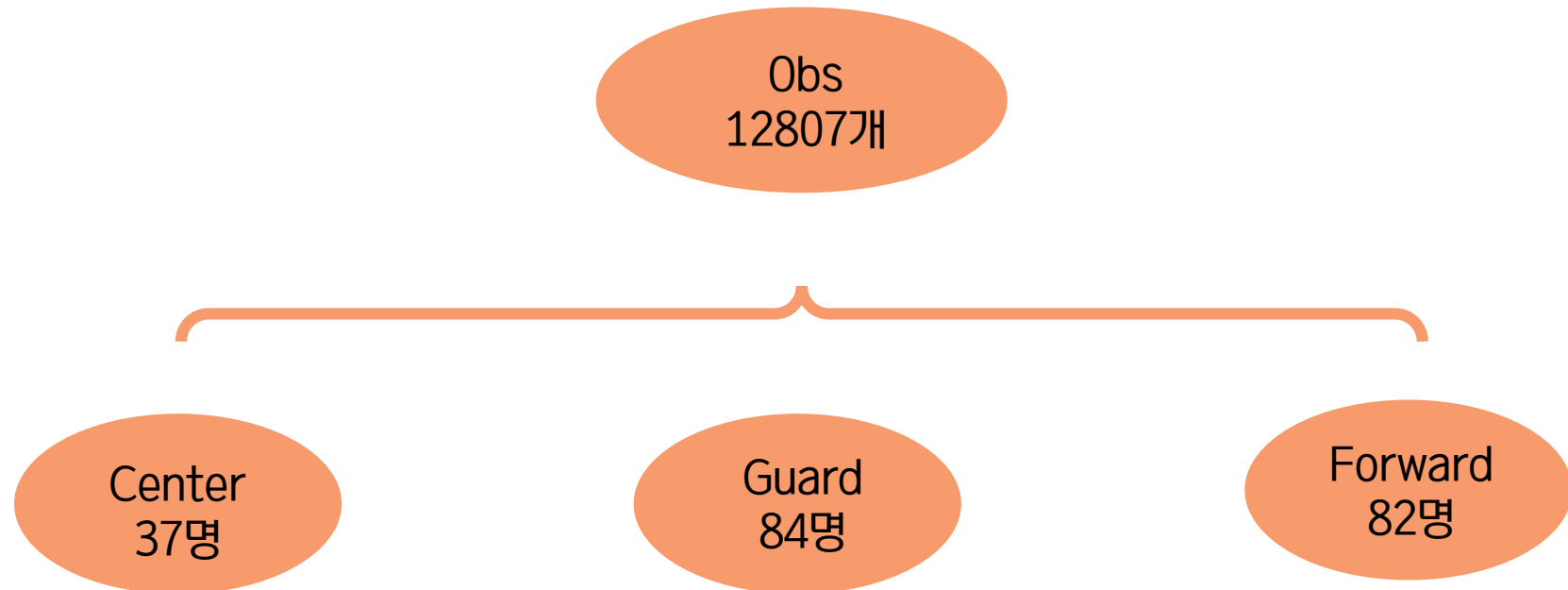
(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

최종 목적인 선수 추천을 하기 위해서  
**경기 별 선수 데이터 -> 선수별로** 변경

Data set을 **포지션 별로 분리함**  
(C, FD, GD 세 데이터 셋 생성)



(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

# 전처리 이후 데이터셋

	player_no	fg	fg_a	ft	ft_a	threep	threep_a	dk	dk_a	pp	pp_a
양동근	210074	-0.178	-0.140	-0.491	-0.618	0.651	0.451	-0.295	-0.302	-0.270	-0.403
김효범	210080	-0.542	-0.443	-0.959	-1.090	0.706	0.713	-0.085	-0.114	-0.719	-0.785
주희정	215012	-0.859	-1.028	-1.307	-1.276	-0.394	-0.595	-0.295	-0.302	-0.653	-0.814
조성민	220194	-0.589	-0.619	0.684	0.412	0.911	0.595	-0.295	-0.302	-0.537	-0.625
박지현	230056	-0.928	-0.987	-0.539	-0.638	0.272	-0.296	-0.295	-0.302	-0.801	-0.779
이시준	235075	-0.832	-0.939	0.187	0.239	0.260	-0.258	-0.295	-0.302	-0.707	-0.889
이정석	240064	-0.907	-0.997	-1.307	-1.432	-0.331	-0.854	-0.295	-0.302	-0.711	-0.736
이현민	250067	-0.610	-0.664	-0.199	-0.392	-0.917	-1.196	-0.295	-0.302	-0.542	-0.474
박구영	290101	-0.930	-1.206	-0.849	-0.869	-0.021	0.474	-0.295	-0.302	-0.776	-1.014
김태술	290108	-0.032	0.057	0.707	0.784	-0.957	-1.214	-0.295	-0.302	-0.195	-0.121
정영삼	290114	-0.178	-0.082	1.066	0.895	0.654	0.650	-0.295	-0.302	-0.429	-0.416
정병국	290115	0.778	0.731	-0.109	-0.286	0.360	-0.522	-0.295	-0.302	-0.682	-0.685
신명호	290116	-0.794	-0.851	-0.865	-0.223	-1.639	-0.921	-0.295	-0.302	-0.575	-0.514
이팔재	290122	-0.395	0.217	-0.523	-0.253	-0.347	-0.223	-0.295	-0.302	-0.934	-0.878
정재홍	290234	-0.426	-0.608	-0.462	-0.571	0.526	0.120	-0.295	-0.302	-0.413	-0.500
강병현	290239	-0.852	-0.947	-0.804	-0.720	-0.585	0.183	-0.295	-0.302	-0.644	-0.726
양우섭	290243	-0.106	-0.059	-0.878	-0.904	-0.019	-0.271	-0.295	-0.302	-0.159	-0.128
김강선	290283	-0.375	-0.421	0.603	0.380	0.635	-0.055	-0.295	-0.302	-0.147	-0.122
박성진	290285	0.094	0.044	-0.032	0.062	1.355	1.729	-0.295	-0.302	-0.090	-0.339
전태풍	290287	0.115	0.162	1.101	0.841	-0.163	-0.444	-0.295	-0.302	0.141	0.100
김종근	290292	-0.935	-0.818	-0.841	-0.923	-1.122	-1.264	-0.295	-0.302	-0.745	-0.595
최윤호	290293	-0.873	-1.155	-0.570	-0.586	1.628	1.012	-0.295	-0.302	-0.765	-1.030

OBS :  
각 선수

Variable :  
동일 시간 단위로  
Scaling된 변수 +  
새로 만든 파생변수

Dataset 구분 :  
포지션별로 3 set

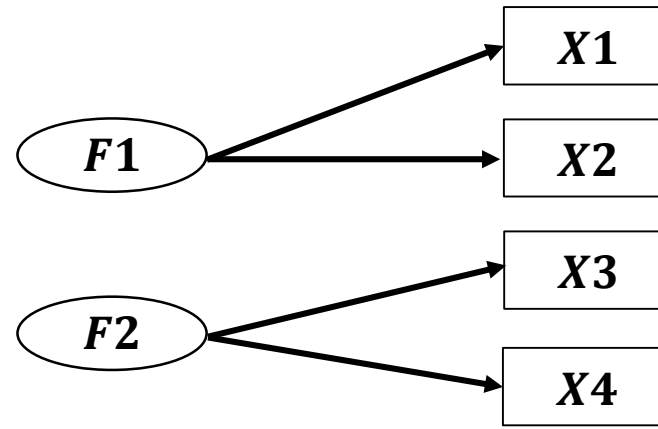
# Factor Analysis

Factor Analysis(요인 분석) 이란?

(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering



기존 변수( $X1 \sim X4$ )의  
**“선형결합”**으로  
새로운 변수( $F1 \sim F2$ ) 생성

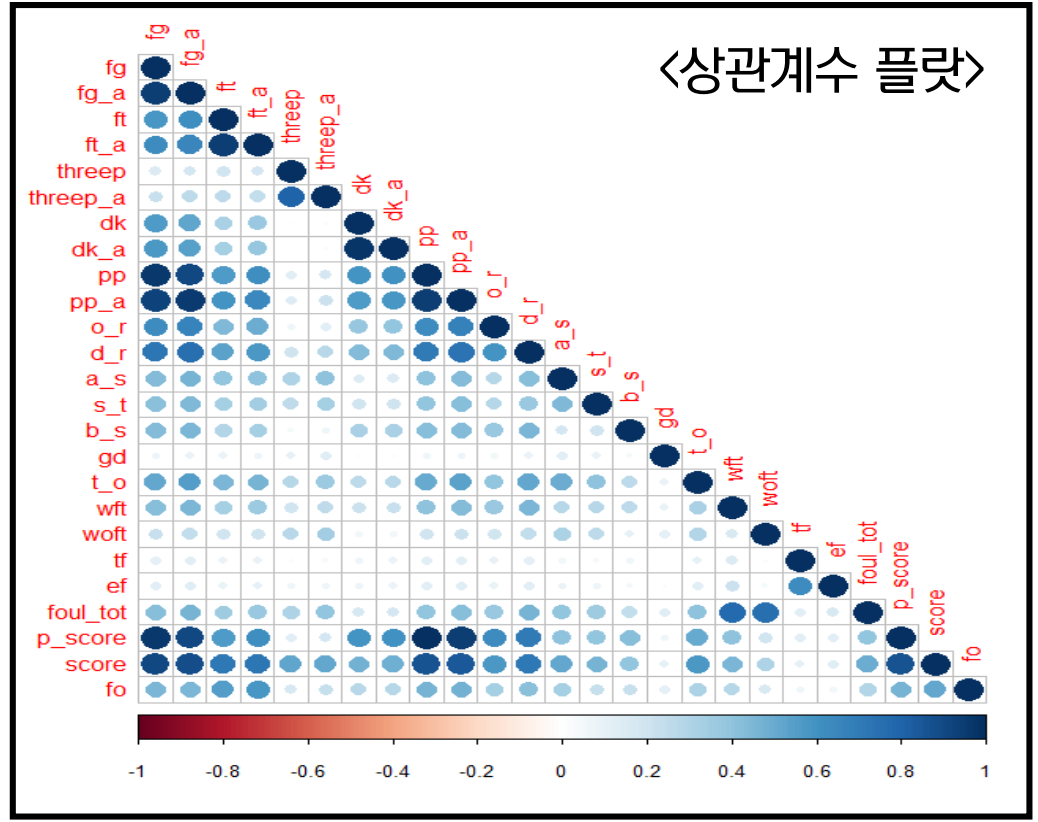
Factor Analysis를 이용해 변수를 통합해 재정의 가능!



단, 변수 간 높은 **“상관관계”**가 존재해야 사용 가능하다!

# Factor Analysis

- (1) 데이터 전처리
- (2) Factor Analysis
- (3) Hierarchical clustering



변수들 간 상관관계가  
충분히 높으므로  
**factor analysis**에 적합!

Factor Analysis를 이용한 활용 예시

2점슛 성공, 3점슛 성공 → 슈팅형 스타일

굿디펜스, 스틸, 블락 → 수비형 스타일



# Factor Analysis

- (1)  
데이터 전처리
- (2)  
Factor Analysis
- (3)  
Hierarchical clustering

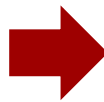
1차적으로 Position 별로 Obs를 구분하여 따로 진행  
(Center, Guard, Forward)



각 Postion 별로 Factor Analysis를 통해  
비슷한 패턴의 Variable들을 축약시킴 (F7까지)



Varimax method로 적절한 축  
Rotation을 진행한다.



한 눈에 파악할 수 있도록  
시각화를 해볼까?



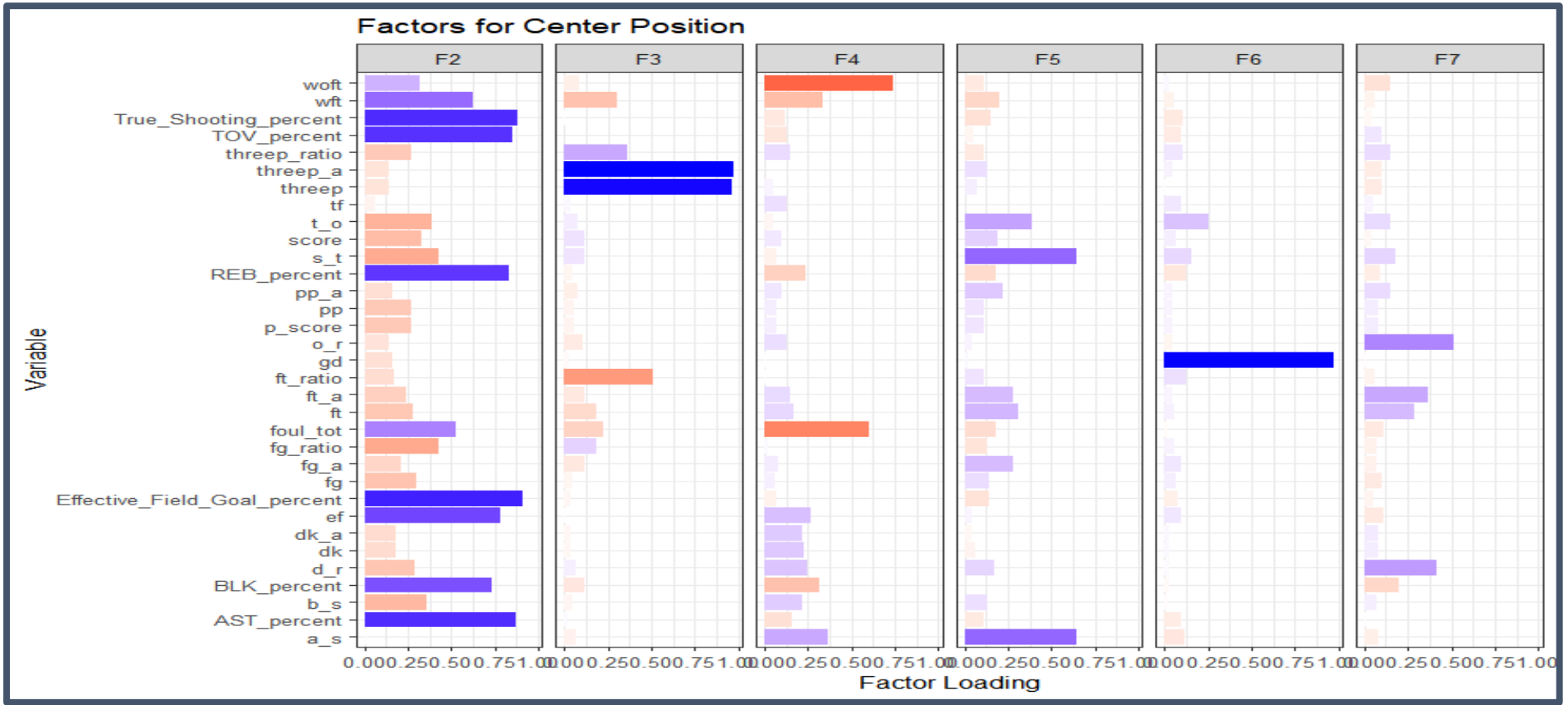
	F2	F3	F4	F5	F6	F7
fg	-0.189	0.271	-0.113	0.202	0.033	-0.013
fg_a	-0.181	0.255	-0.055	0.241	0.042	-0.021
ft	-0.248	-0.069	-0.056	0.097	0.130	0.030
ft_a	-0.225	-0.020	-0.035	0.101	0.131	0.076
threep	-0.237	-0.943	-0.162	0.033	0.000	-0.100
threep_a	-0.187	-0.776	-0.075	0.052	0.022	-0.124
dk	-0.043	0.066	-0.024	-0.146	0.002	0.223
dk_a	-0.047	0.068	-0.008	-0.120	0.032	0.209
pp	-0.202	0.283	-0.111	0.180	0.024	0.027
pp_a	-0.192	0.279	-0.088	0.218	0.036	0.023
o_r	-0.199	0.451	-0.053	0.090	-0.121	0.291
d_r	-0.269	0.225	-0.065	0.178	-0.013	0.348
a_s	-0.184	-0.089	-0.210	0.502	0.130	0.105
s_t	-0.114	0.058	0.043	0.644	0.047	0.072
b_s	-0.191	0.264	-0.043	0.127	-0.077	0.788
gd	0.148	-0.037	-0.029	0.105	0.249	-0.077
t_o	-0.152	-0.093	-0.082	0.454	0.165	-0.032
wft	0.026	0.293	0.567	-0.007	-0.011	-0.063
woft	0.271	-0.025	0.715	-0.058	-0.057	-0.021
tf	-0.212	0.185	-0.118	-0.074	0.427	0.111
ef	-0.126	0.006	0.013	0.061	0.887	-0.022
foul_tot	0.193	0.148	0.973	-0.048	-0.058	-0.086
p_score	-0.202	0.283	-0.111	0.180	0.024	0.027
score	-0.307	-0.167	-0.171	0.203	0.056	-0.047
True_Shooting_percent	0.953	0.091	0.099	-0.083	-0.025	-0.068
Effective_Field_Goal_percent	0.950	0.105	0.073	-0.089	-0.005	-0.089
AST_percent	0.911	0.075	0.186	-0.080	0.015	-0.025
BLK_percent	0.828	-0.062	0.305	-0.036	-0.061	0.025
TOV_percent	0.889	0.107	0.145	-0.086	0.045	-0.053
REB_percent	0.888	0.104	0.174	-0.136	-0.070	-0.107
fg_ratio	-0.145	0.205	-0.526	0.026	0.008	-0.132
ft_ratio	-0.305	-0.223	0.115	-0.051	-0.035	0.007
threep_ratio	-0.180	-0.345	-0.313	-0.064	-0.076	-0.014

실제 설명변수들과 각 새로 정립된 Factors간의 loadings  
(Pattern coefficients)

02 KBL 자체  
선수 추천

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

# Center Position Visualization



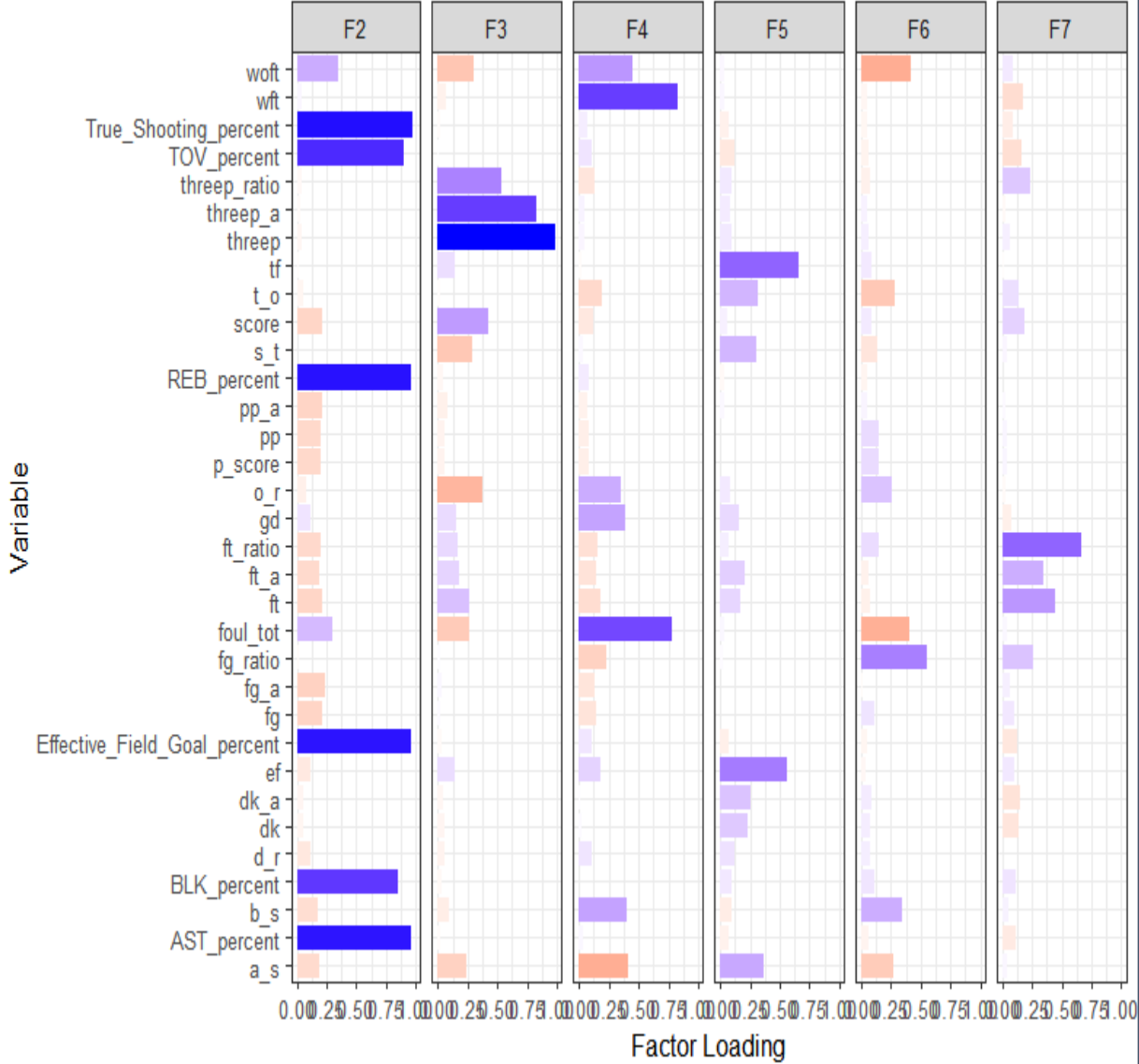
실제 설명변수들과 각 Factor 간의 loadings을 시각화해 봄으로써  
Factor에 축약 되어있는 정보가 무엇인지 판단



다른 두 포지션도 시각화 해볼까?

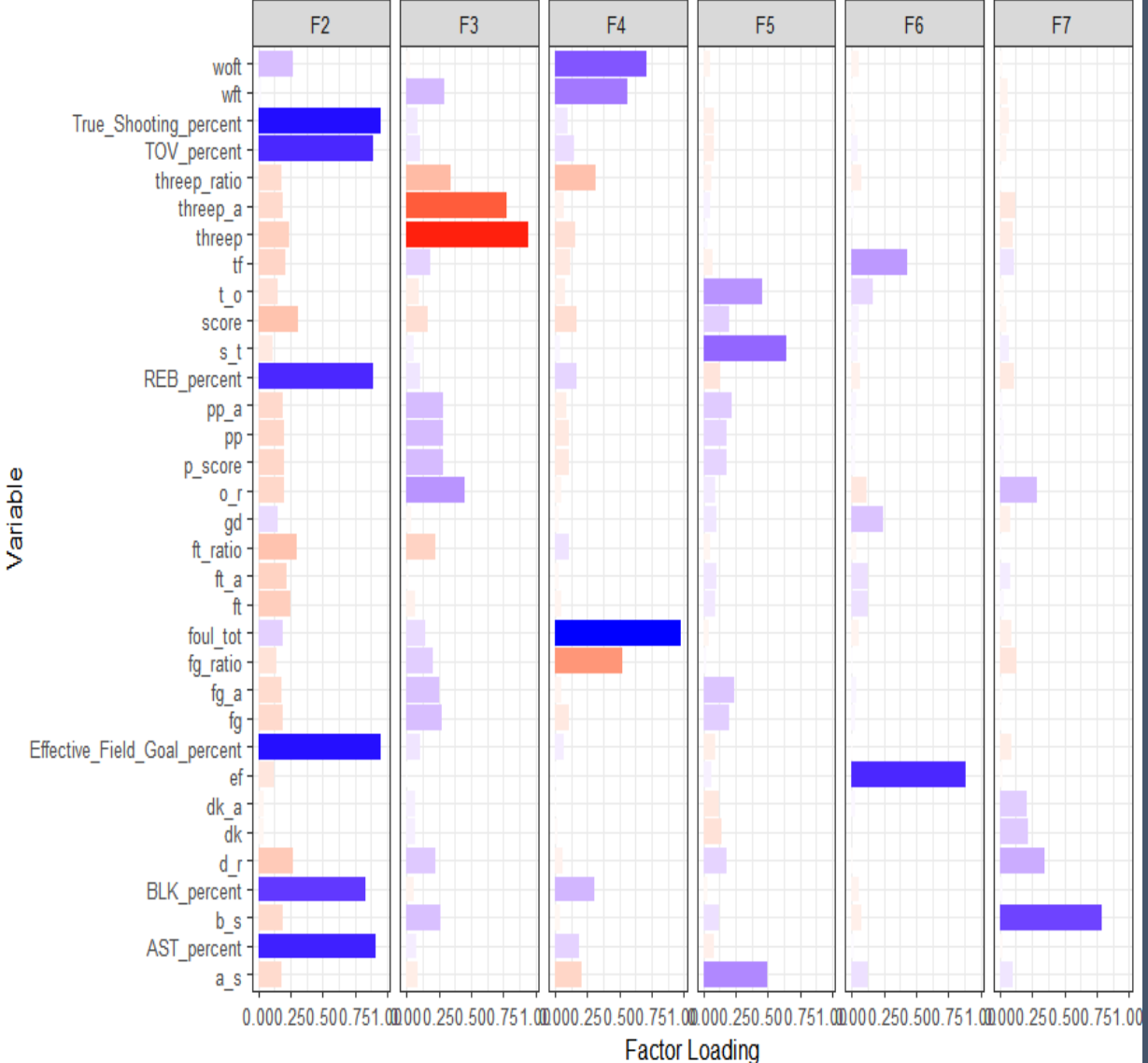
# Guard Position

Factors for Guard Position



# Forward Position

Factors for Forward Position



- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

# 시각화 결과를 바탕으로 각 Factor를 Labeling!!!

Why? 이후 User가 질문지를 답변할 때 하나의 Factor가 하나의 선택지가 되기  
때문에 Label을 명시하여 User가 이해하기 쉽도록 한다!

Pos :  
Center

Factor	Variable
factor1	TS%, EFG%, BLK%, AST%
factor2	굿디펜스, 스틸
factor3	턴오버, 스틸, 어시스트, 블록
factor4	공격리바운드, 수비리바운드
factor5	3점성공률, 자유투 성공률
factor6	덩크시도, 덩크성공



New variable
다재 다능형 (All-Around-Player)
수비 전문형
서포터형 (조력자 역할)
리바운드의 제왕
스트레치형 빅맨 (슛 쏘는 센터)
덩크왕

- TS% : True Shooting %
- EFG% : Effective Field Goal %
- BLK% : 블락할 확률 추정치
- AST%: 어시스트할 확률 추정치

# 시각화 결과를 바탕으로 각 Factor를 Labeling!!!

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

Pos :  
Guard

factor	Variable
factor1	TS%, EFG%, BLK%, AST%
factor2	3점슛 시도, 3점슛 성공, 3점슛 성공률
factor3	파울(자유투 유,무), 한경기 총 파울, 굿디펜스, 블록
factor4	어시스트, 스틸
factor5	파울(-), 야투 성공률, 블록
factor6	자유투 시도, 자유투 성공률



New variable
다재다능형 (All-Around-Player)
3점 스페셜리스트
수비 전문형
패스,스틸 전문형
센스 만점 지능형
파울 유도형

- TS% : True Shooting %
- EFG% : Effective Field Goal %
- BLK% : 블락할 확률 추정치
- AST% : 어시스트할 확률 추정치

( - )는 해당 값의 음수 값이므로 반대를 의미

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

Pos :  
Forward

# 시각화 결과를 바탕으로 각 Factor를 Labeling!!!

factor	Variable
factor1	TS%, EFG%, BLK%, AST%
factor2	페인트존 슛, 공격리바운드, 3점슛 시도(-), 3점슛 성공률(-)
factor3	득점, 스틸, 어시스트
factor4	굿 디펜스, 파울
factor5	파울(자유투 유,무), 한경기 총 파울
factor6	덩크, 리바운드, 블록



New variable
다재다능형 (All-Around-Player)
과감한 돌파형
센스 만점 플레이
지능형 수비 플레이
과격하고 거친형
골밑 공략형

- TS% : True Shooting %
  - EFG% : Effective Field Goal %
  - BLK% : 블락할 확률 추정치
  - AST% : 어시스트할 확률 추정치
- ( - )는 해당 값의 음수 값이므로 반대를 의미



# Factor Analysis 이후 Data set

(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

	player_no	F1	F2	F3	F4	F5	F6	F7
김주성	215053	0.284	-0.588	-0.708	-0.729	0.400	1.773	0.377
오용준	230061	-1.179	0.671	-1.070	1.240	-1.351	-1.130	-1.066
김동욱	235070	-0.329	-0.452	-1.368	-0.891	0.192	-0.171	-0.652
한정원	240074	-1.173	1.932	-0.882	2.200	-1.075	-1.445	-0.605
함지훈	290100	0.055	-0.537	-0.036	-1.037	0.332	-0.328	0.417
양희종	290106	-0.524	-0.201	-0.577	-0.139	0.040	-0.118	-0.155
박상모	290118	0.039	-0.333	-0.051	-0.212	0.156	0.054	-0.102
김영환	290119	-0.275	-0.402	-0.876	-0.861	-0.010	-0.125	-0.510
윤호영	290218	-0.184	-0.353	-0.097	-0.529	0.229	-0.415	0.664
천대현	290219	-1.024	0.661	-0.888	1.044	-0.886	-0.292	-1.210
정취랄	290225	-0.689	1.679	0.850	2.087	-1.127	-1.311	-0.498
김민수	290227	-0.082	-0.433	-0.171	-0.367	-0.313	-0.574	0.426
기승호	290231	-0.080	-0.197	-0.035	-0.337	-0.106	-0.459	-0.481
이지운	290232	-0.975	0.617	-0.840	0.334	-1.097	-0.498	-0.861
애런 헤인즈	290261	2.106	-1.343	1.612	-1.542	2.177	1.524	2.200
문태영	290280	0.241	-0.565	-0.022	-0.825	0.400	1.048	-0.062
허일영	290284	-0.383	-0.303	-0.536	-0.533	-0.398	-0.811	-0.608
김우겸	290295	-0.391	0.980	0.893	1.632	-0.702	0.007	-0.681
제스퍼 존슨	290335	-0.130	-0.521	-1.615	-0.264	0.808	-0.256	-0.573
이민재	290352	-1.593	2.497	0.242	3.408	-1.617	-1.415	-0.512

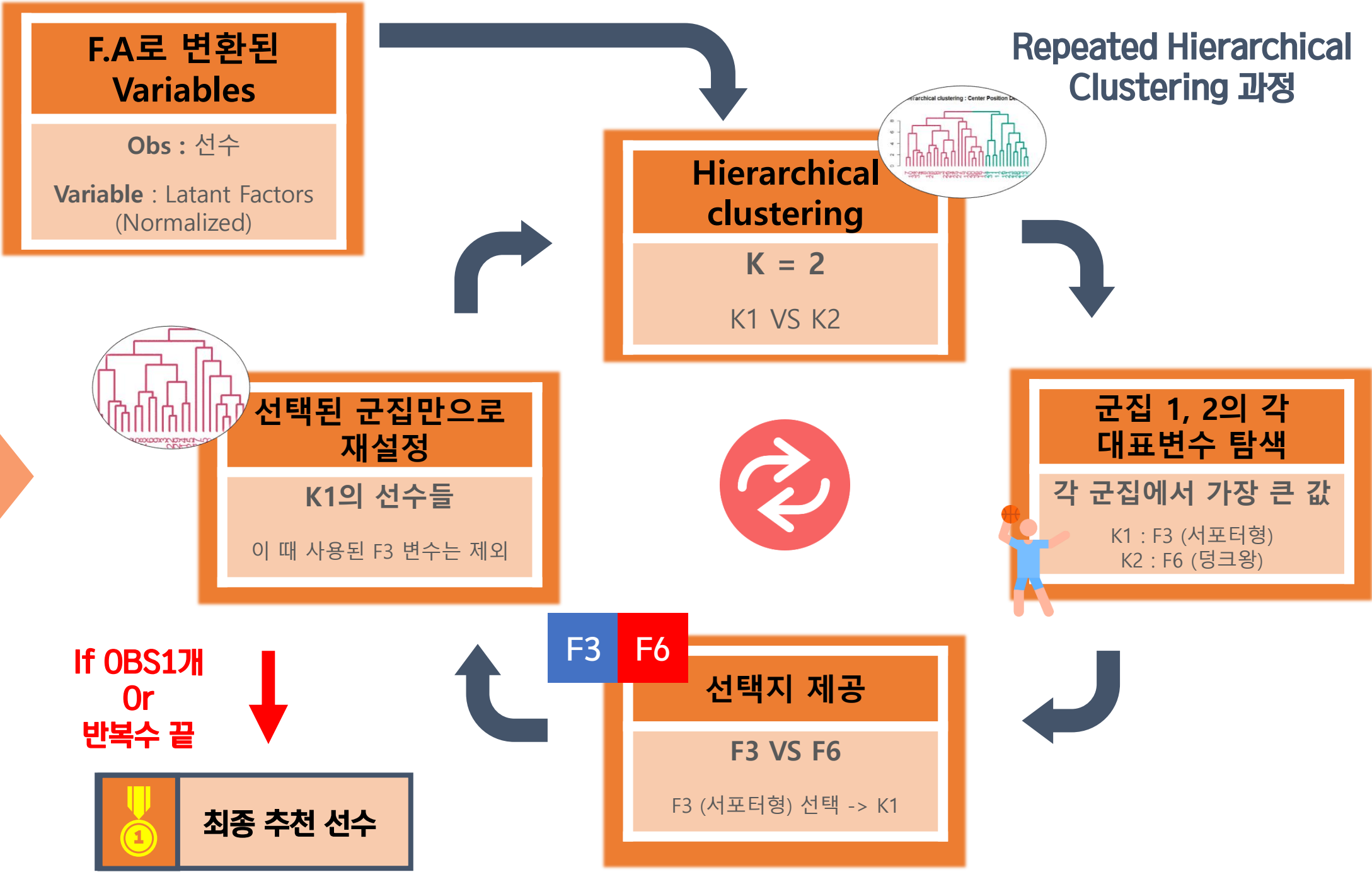
OBS :  
각 선수

Variable :  
재정립한 Factor

Dataset 구분 :  
포지션별로 3 set

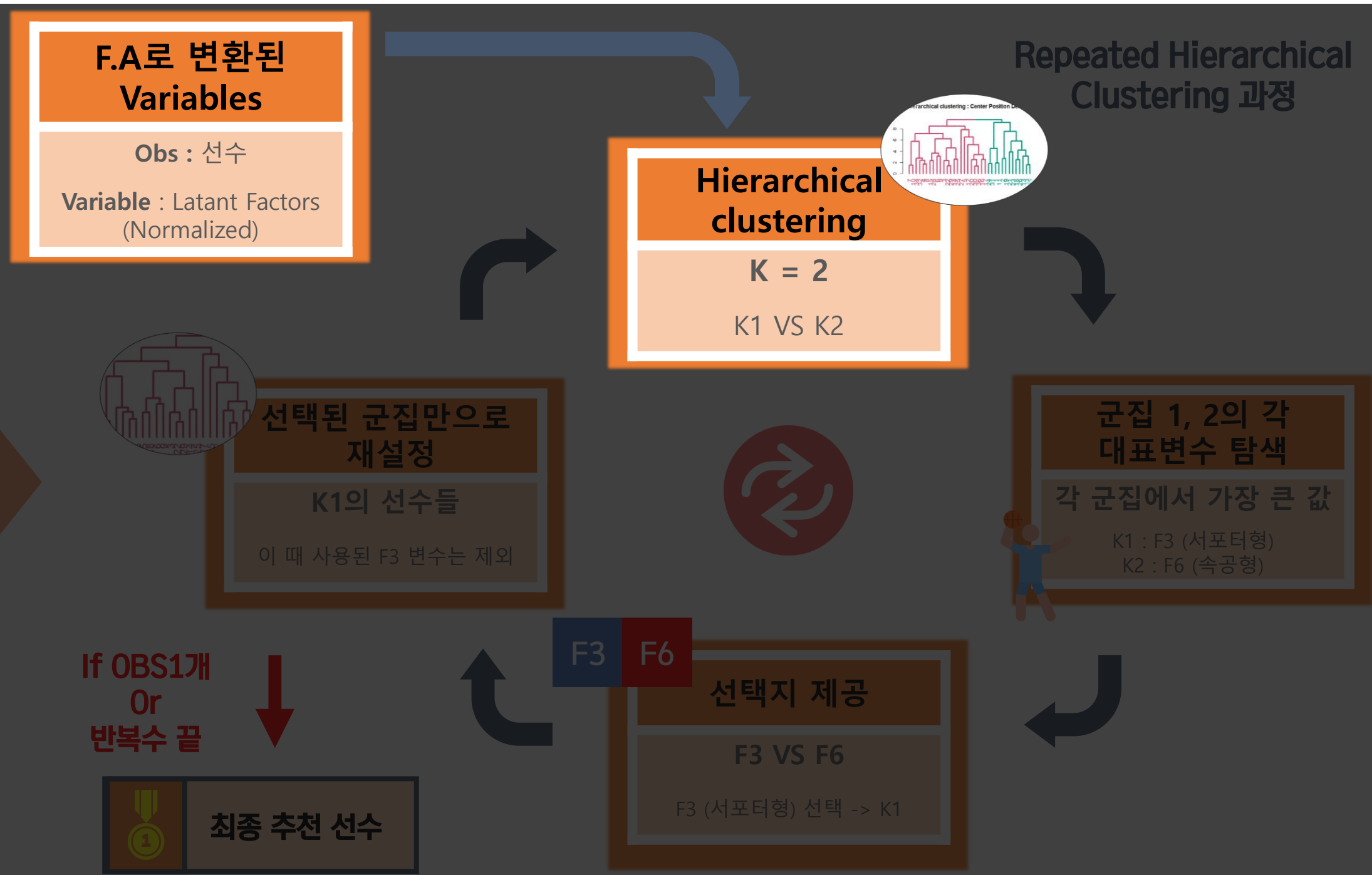
02 KBL 자체  
선수 추천

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering



02 KBL 자체  
선수 추천

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering



# Hierarchical Clustering (계층군집분석) 이란?

(1)  
데이터 전처리

(2)  
Factor  
Analysis

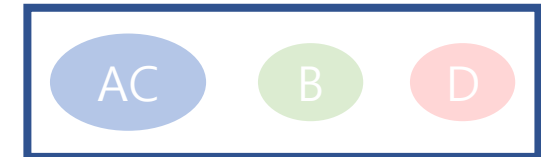
(3)  
Hierarchical  
clustering

1. 변수들간의 **Distance Matrix**를 구한다.  
(모두 수치형이므로 Euclidian 거리 사용)

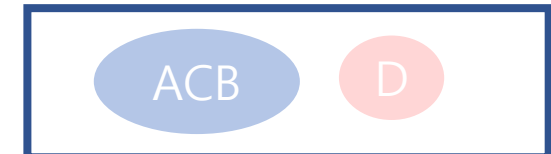
	A	B	C	D
A	0	0.9	0.6	1.7
B	0.9	0	0.7	2.9
C	0.6	0.7	0	1.8
D	1.7	2.9	1.8	0



2. 거리가 가까운 변수들끼리 하나의 '군집' 으로 묶는다.



3. 서로 유사한 군집끼리 하나의 군집으로 합친다.



**군집간의 유사도는 어떻게 알 수 있을까?**

: *Ward.d2 Method* =

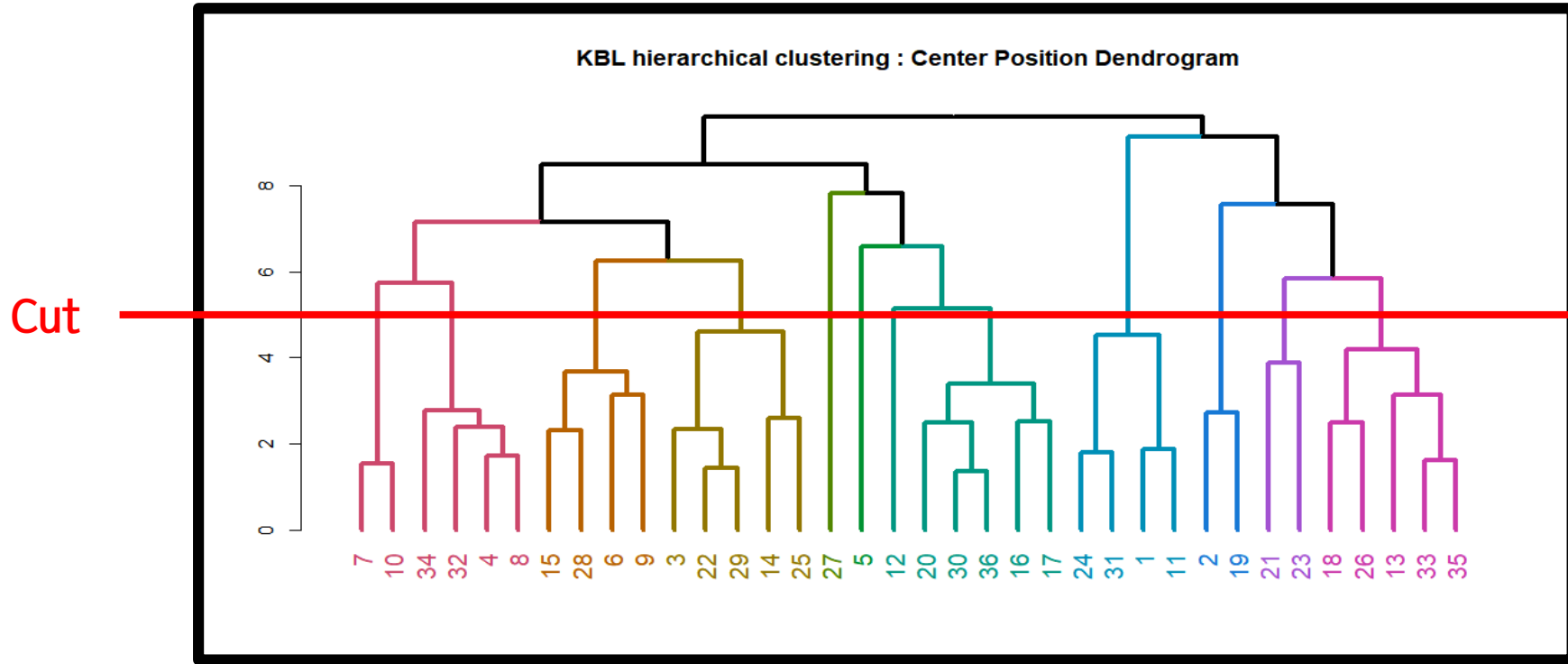
각 군집내 값들의 분산이 최소화 되는 경우를 유사한 것으로 판정.  
(비슷한 값들끼리 묶일 경우 해당 집단의 분산은 작아지기 때문)

# Hierarchical Clustering (계층군집분석) 이란?

(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

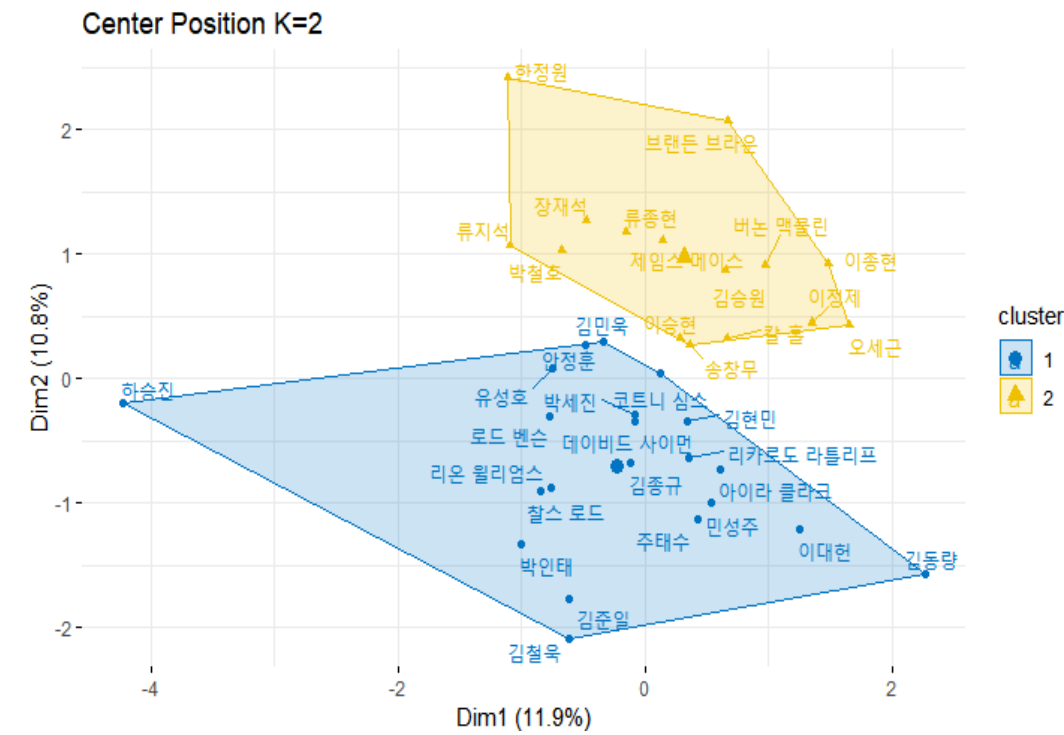
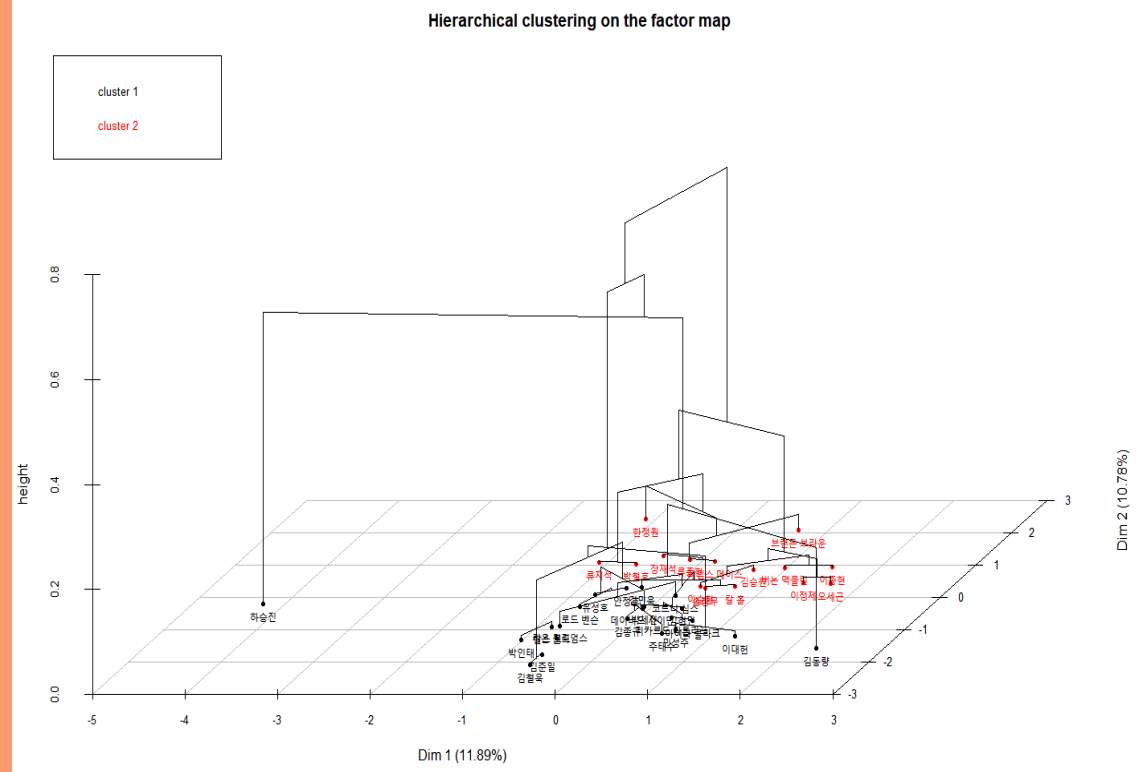


특징 :

시작 시 군집개수  $k$ 를 지정하지 않고  
Dendrogram 을 보며 Cut을 조정하면서 군집을 몇 개로 나눌 지 선택할 수 있음.

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

# Step1. K=2 Hierarchical Clustering ( 예시 : Center 포지션 )



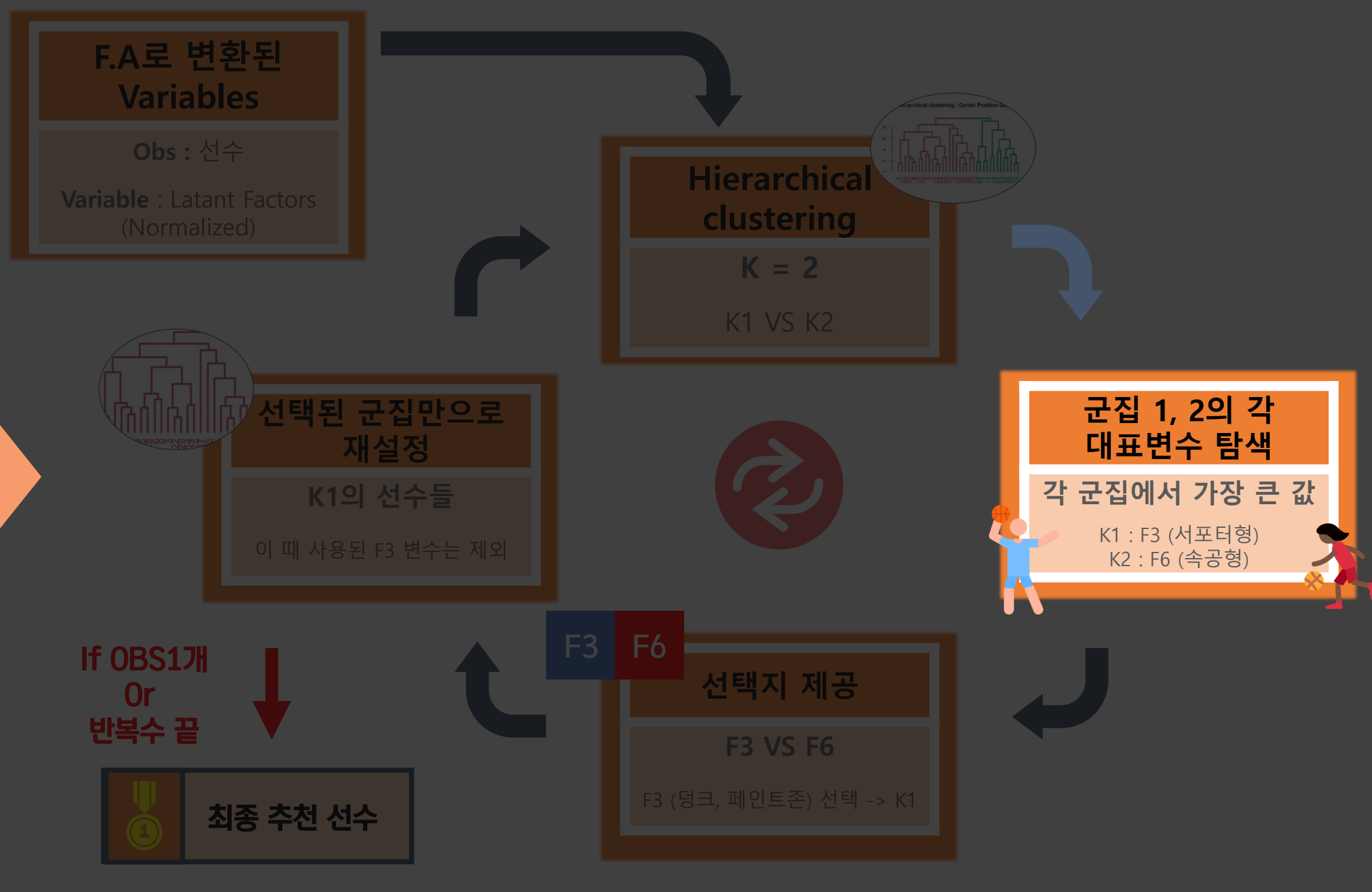
H\_clust의 결과를 K=2로 cut 조정후 PCA 축으로 시각화

좌측의 그림을 위에서 바라보았을 때 군집화된 모습

앞서 설명 한 계층군집분석의 방식대로 먼저 두 개의 군집으로 나눈다.

02 KBL 자체  
선수 추천

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering





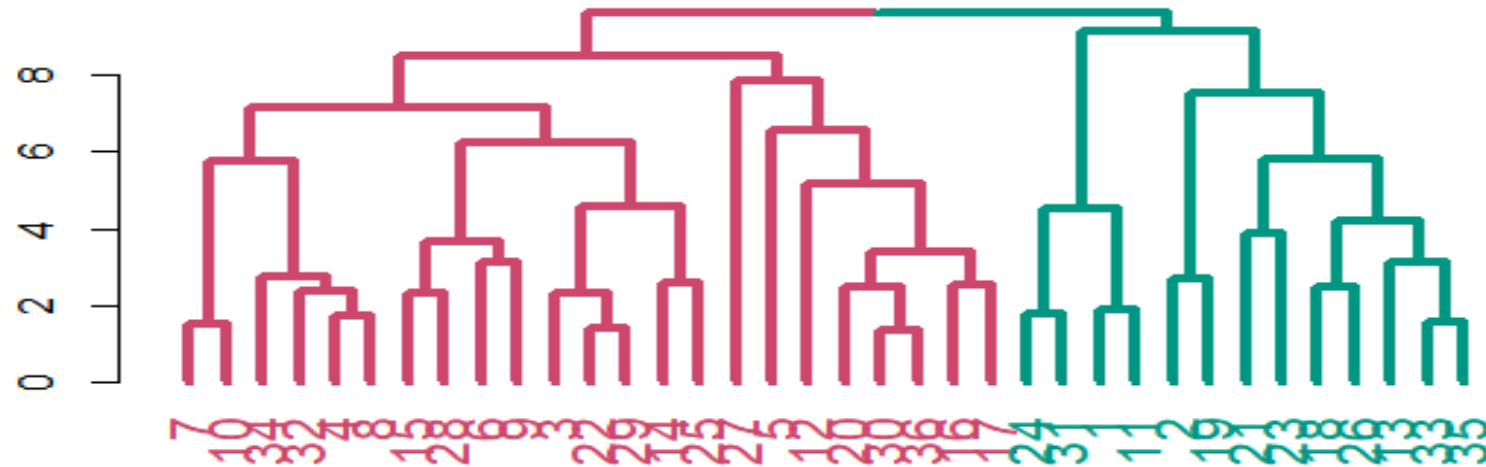
## Step2. 군집 1, 2의 대표변수 탐색

(1)  
데이터 전처리

(2)  
Factor  
Analysis

(3)  
Hierarchical  
clustering

KBL hierarchical clustering : Center Position Dendrogram



Center 포지션에 대해 K=2로 군집을 나누었을 때 Dendrogram

이 때 각 obs가 가지는 변수 F1, F2, ..., F8는 모두 Normalized 되어있는 상태다.  $\sim N(0,1)$   
즉, 모든 값들의 Scale이 같기 때문에 군집 내에서 평균적으로 가장 큰 값을 가지는 변수가  
해당 군집을 대표한다고 볼 수 있다. 예를 들어

군집1 : F1(만능형), F2 (수비 전문형), ..., F8 각 평균 中 가장 큰 값을 가지는 변수 → 대표변수 F3

군집2 : F1(만능형), F2 (수비 전문형), ..., F8 각 평균 中 가장 큰 값을 가지는 변수 → 대표변수 F6

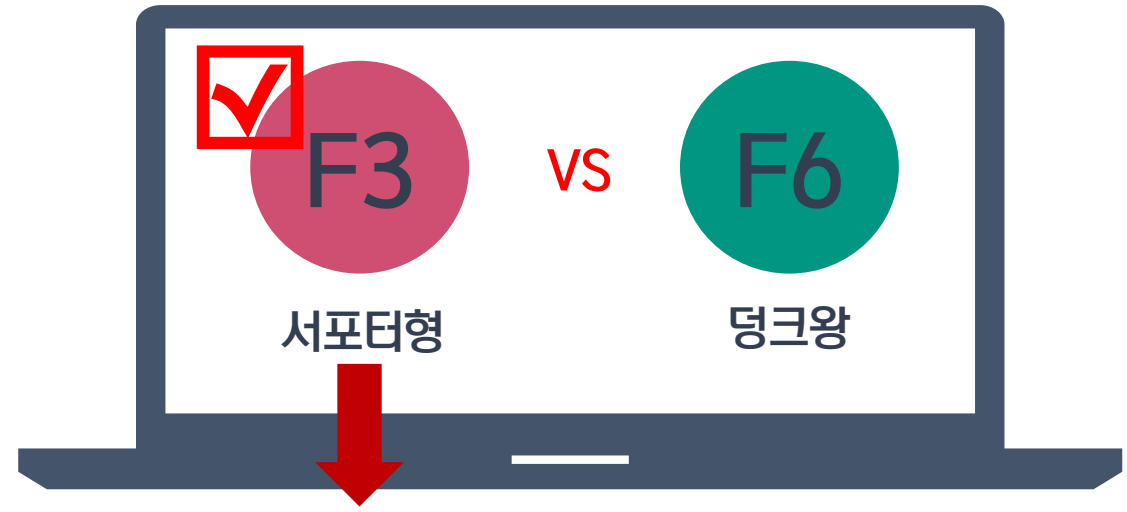
02 KBL 자체  
선수 추천

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering



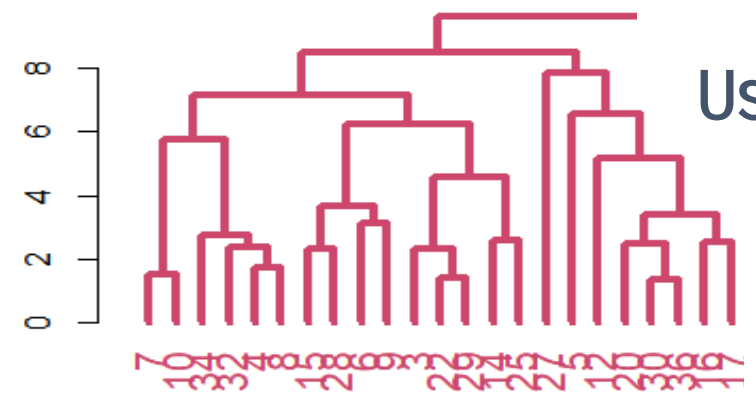
- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering

# Step3. 두 대표변수로 선택지 제공 (To user)



User가 F3, 즉 패스보다  
덩크나 페인트존 플레이가  
좋은 선수를 선호한다고 체크!

KBL hierarchical clustering : Center Position Dendrogram

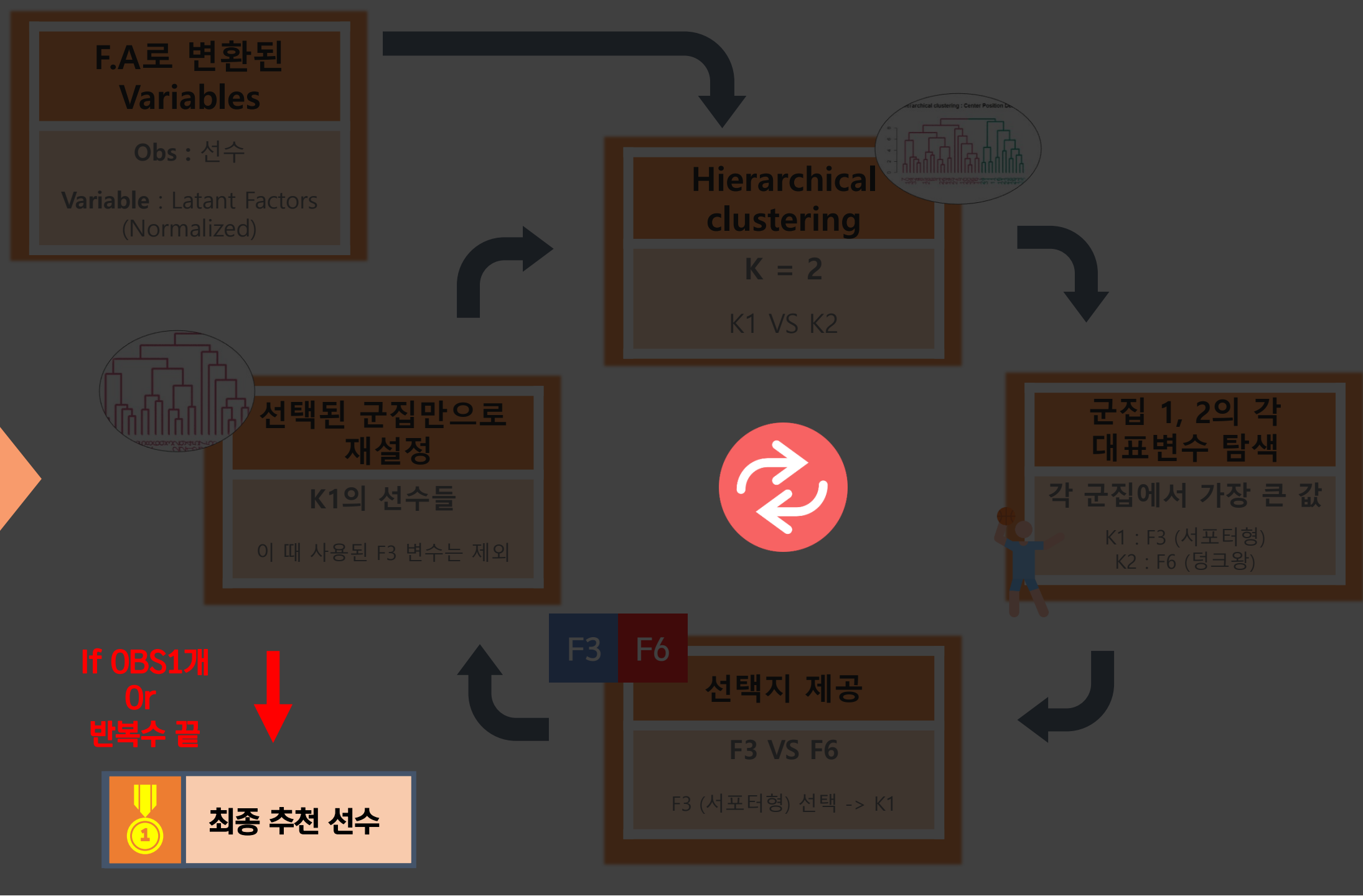


User에 의해 선택된 군집만을 대상으로  
다시 K=2 클러스터링 진행

! 이때 선택된 F3은 이미 User에 의해 선택된 변수이므로 다음  
클러스터링 과정에서 제외한다. New군집에서의 변수는 : F1, F2, F4 ,...

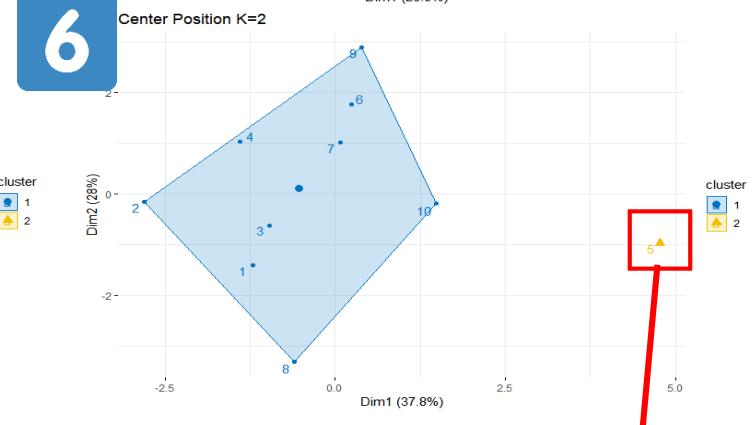
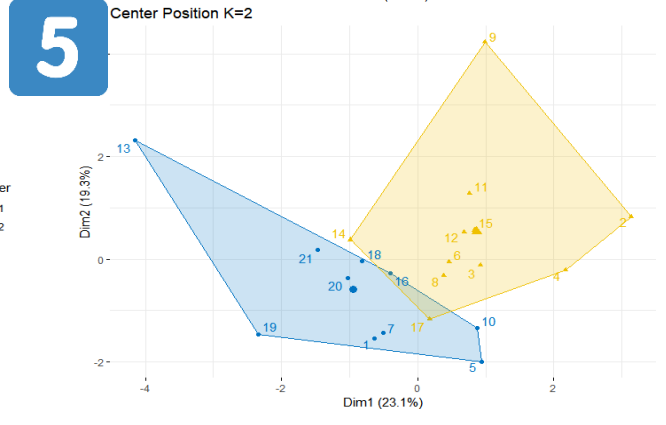
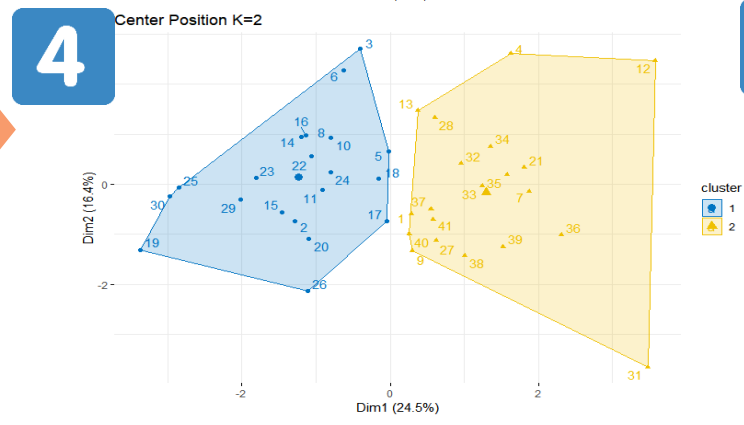
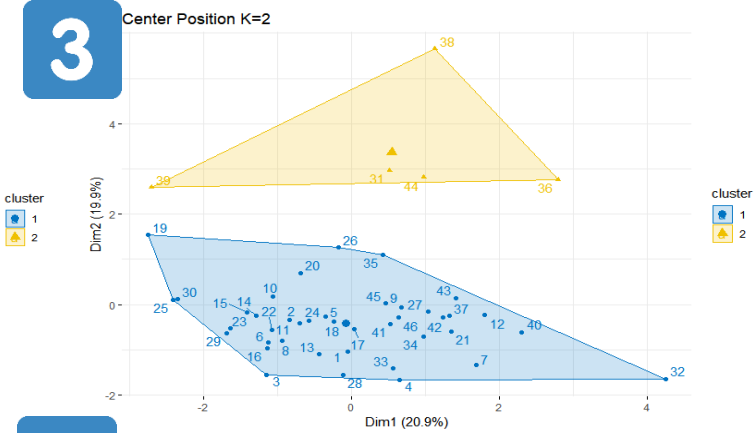
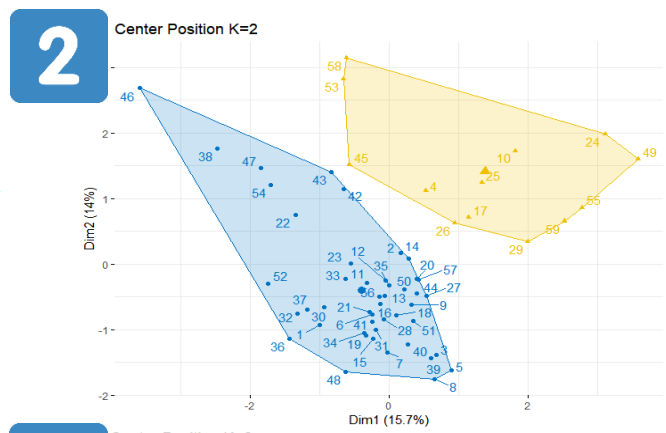
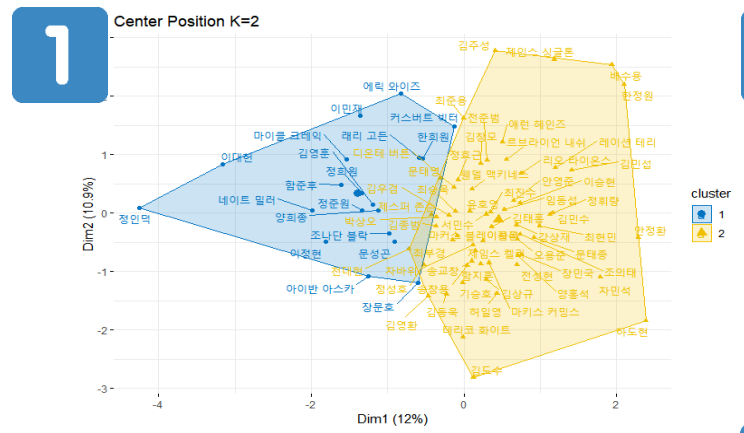
02 KBL 자체  
선수 추천

- (1)  
데이터 전처리
- (2)  
Factor  
Analysis
- (3)  
Hierarchical  
clustering



# Step4. 지금까지의 과정을 반복

- (1) 데이터 전처리
- (2) Factor Analysis
- (3) Hierarchical clustering



[ FD 포지션을 대상으로 시뮬레이션한 예시결과를 PCA 축으로 시각화 ]  
질문의 답변이 총 6번 이루어진 결과 추천 선수가 도출되었다

1차적으로 포지션별로 선수들을 나눈 상태에서 진행되기 때문에  
선수가 추천 되기까지 5~6번 정도의 Clustering이 평균적으로 이루어짐을 확인.

추천된 선수

# 최종 시뮬레이션 Function 결과

- (1) 데이터 전처리
- (2) Factor Analysis
- (3) Hierarchical clustering

```
> Recommend_center(kbl_fa_c_final,7)
[1] "F6" "F1"
Please Choose K for 1 or 2 : 1
Criteria      Value Your_Choice
1      F6 1.1650258          F6
2      F1 0.3551603
[1] "F2" "F4"
Please Choose K for 1 or 2 : 1
Criteria      Value Your_Choice
1      F2 0.4823236          F2
2      F4 1.6477127
[1] "F5" "F4"
Please Choose K for 1 or 2 : 2
Criteria      Value Your_Choice
1      F5 0.3939682          F4
2      F4 0.6127613
[1] "F5" "F1"
Please Choose K for 1 or 2 : 2
Criteria      Value Your_Choice
1      F5 -0.02826497        F1
2      F1 -0.55317971
```

- 1. F6(골밑 공략형) VS F1(다재 다능형) 선택지  
-> F6 선택
- 2. F2 VS F4 선택지  
-> F2 선택
- 3. F5 VS F4 선택지  
-> F4 선택
- 4. F5 VS F1 선택지  
-> F1 선택

```
$Your_player
[1] "김기훈"

$Your_player_stat
  player_no      F0      F1      F2      F3      F4      F5      F6
1   290794 -0.04806799 -0.5531797 0.6814567 -0.6319893 0.651895 -0.6845326 0.7231663

$Your_preference
[1] "F6" "F2" "F4" "F1"
```

최종 추천 선수

선수의 Factor 값 (Stat)

User의 선호도 순위  
(Factor 선택 순차적 기록)

선수 유형 추천 시스템

# 3 유사 선수 추천

1. 데이터 전처리
2. Contents Based Filtering





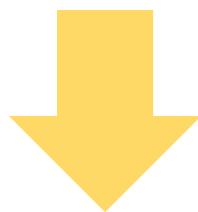
3



## 유사 선수 추천 개요



한국 팬들에게 인지도가 높은 1980년대~현역  선수들을  
추출하여  선수들과의 Euclidean Distance 계산

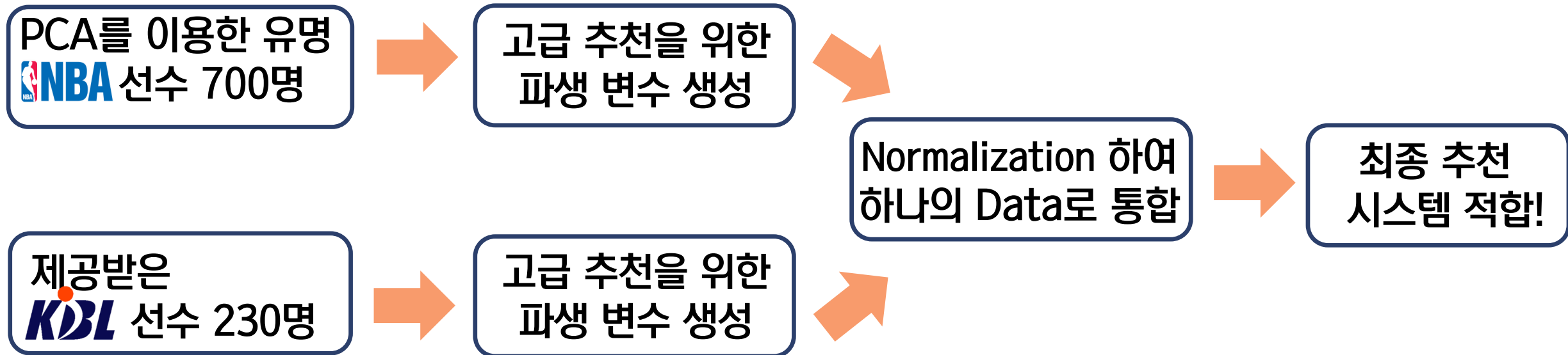


가장 가까운 거리의 **3명을 추천** 함으로써  
플레이 성향이 비슷한  선수를 추천해줄 수 있다!

### 3 NBA 유사 선수 추천 개요



#### < 알고리즘 순서도 >



NBA 데이터 출처

: <https://www.kaggle.com/drgilermo/nba-players-stats>

# 데이터 전처리

Principal Component Analysis (주성분 분석) 이란?

(1)  
Data  
Preprocessing

(2)  
Contents  
Based  
Filtering



다변량 데이터의 차원을 축소시키는 효과적 방법 !

Feature Extraction

“변수 추출”

모든 변수를  
조합하여 데이터를 잘  
표현할 중요 성분을  
추출하기



$X_1$   
 $X_2$   
 $X_3$   
 $X_4$

$C_1$   
 $C_2$   
 $C_3$

중요 성분을 가진  
새로운 변수를  
추출!

PCA를 이용하여 NBA 선수 all-time-legend 700명을 추출해보자 !

# 데이터 전처리

\*PC (principal component) : 주성분

PCA를 통한 **All time legend NBA** 선수 추출하기

- PC성분 추출

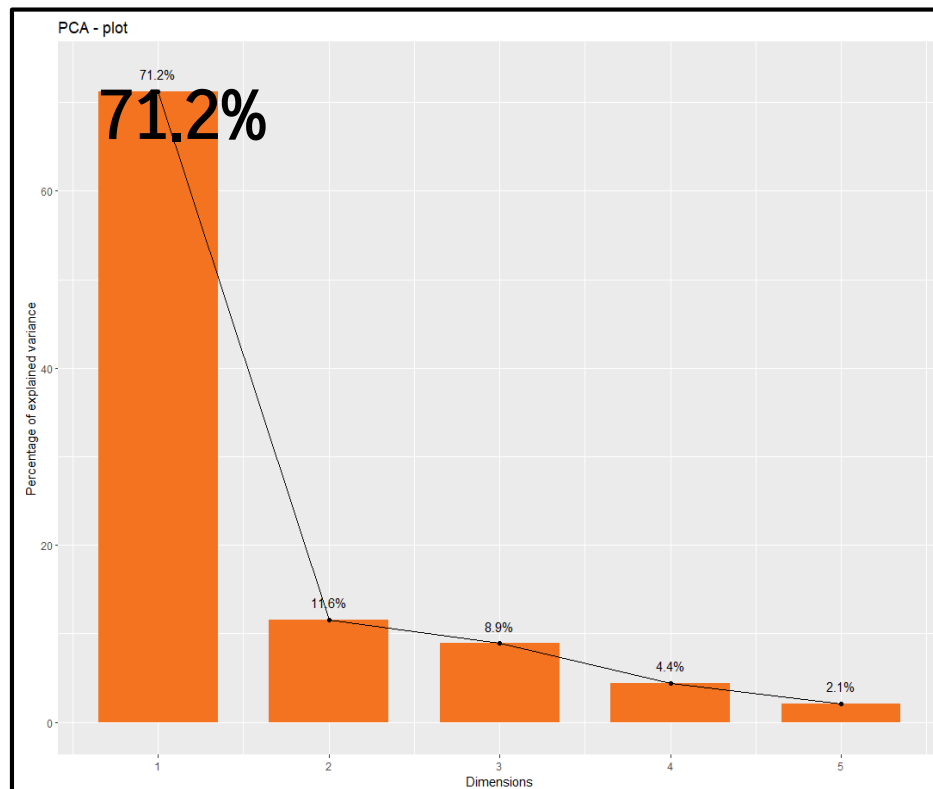


```
nba_PCA <- prcomp(nba[, c(MP, PPG, APG, RPG, SPG,  
                           BPG, `FG%`, `3P%`, `2P%`, `F%`,  
                           DRPG, ORPG, TPG)],  
                  center = TRUE,  
                  scale. = TRUE)
```

경기당 평균 득점, 어시스트,  
리바운드, 스틸, 블록, 야투율,  
3점성공율, 턴오버

10개의 PC\*로 재정립!

첫번째 PC의 설명력 : **71.2%**



(1)  
Data  
Preprocessing

(2)  
Contents  
Based  
Filtering

# 데이터 전처리

PCA를 통한 **All time legend NBA** 선수 추출하기

- 유명 NBA 선수 추출

(1)  
Data  
Preprocessing

R

```
> cor(nba_C_PCA$scores[, "Comp.1"], nba_C$PER)  
[1] 0.8105873
```

PC1와 PER(선수 생산성 지수)

과의 상관관계가 매우 높음 !

(2)  
Contents  
Based  
Filtering

따라서 legend NBA를 추출할 때 PER만을 고려하는 것보다  
PC1을 함께 고려하는 것이 더 정확할 것으로 판단

R

```
by_PER<- nba_C %>% arrange(desc(PER)) %>%  
  transmute(PY = str_c(Player, Year, sep = "_"))  
  
by_PC1 <- nba_C %>% arrange(desc(PC1)) %>%  
  transmute(PY = str_c(Player, Year, sep = "_"))  
  
Join <-inner_join(a,b, by = c("PY" = "PY")) %>%  
  separate(PY,c('player', 'season'), sep = "_") %>%  
  distinct(player, .keep_all = TRUE) %>% slice(1:140)
```

PER과 PC1이 높은 선수를 기준으로

‘1980년대 ~ 현재’

NBA 선수들을 주어진 KBL 데이터 포지션 비율  
에 맞춰서 추출 !

ex) 센터 : 가드 : 포워드 = 2 : 4 : 4

# 데이터 전처리

PCA를 통한 **All time legend NBA** 선수 추출하기

- 최종 선택 변수 선별

◀ NBA 데이터셋과 KBL데이터 셋을 하나로 통합하는  
과정에서 **겹치지 않는 변수는 제거**해 주었다. ▶

변수명	설명	변수명	설명
Fg	2점슛 성공수	O_r	공격 리바운드수
Fg_a	2점슛 시도수	D_r	수비 리바운드수
Ft	자유투 성공수	A_s	어시스트 수
Ft_a	자유투 시도수	S_t	스틸 수
3p	3점슛 성공수	B_s	블락 수
3p_a	3점슛 시도수	T_o	턴오버 수

NBA\_KBL 통합 데이터 셋 변수들

# 데이터 전처리

PCA를 통한 All time legend NBA 선수 추출하기

- 최종 선택 변수 선별

변수명	설명
TS%	True Shooting% - 선수의 슈팅 효율성 지수
EFG%	Effective Field Goal % - 2점슛보다 3점슛에 높은 가중치를 부여한 효율적 슈팅 지수
AST%	해당 선수가 코트에 있는동안 만들어진 필드골이 해당 선수에 의해 어시스트 될 확률 추정치
BLK%	해당 선수가 코트에 있는 동안 상대팀이 시도한 2점슛이 해당 선수에 의해 블락될 확률 추정치
TOV%	플레이 100번 당 턴오버를 할 횟수에 대한 추정치
REB%	해당 선수가 코트에 있는 동안 발생한 리바운드가 해당 선수에 의해 리바운드 될 확률 추정치

NBA\_KBL 통합 데이터 셋 변수들



# 데이터 전처리

PCA를 통한 **All time legend NBA** 선수 추출하기

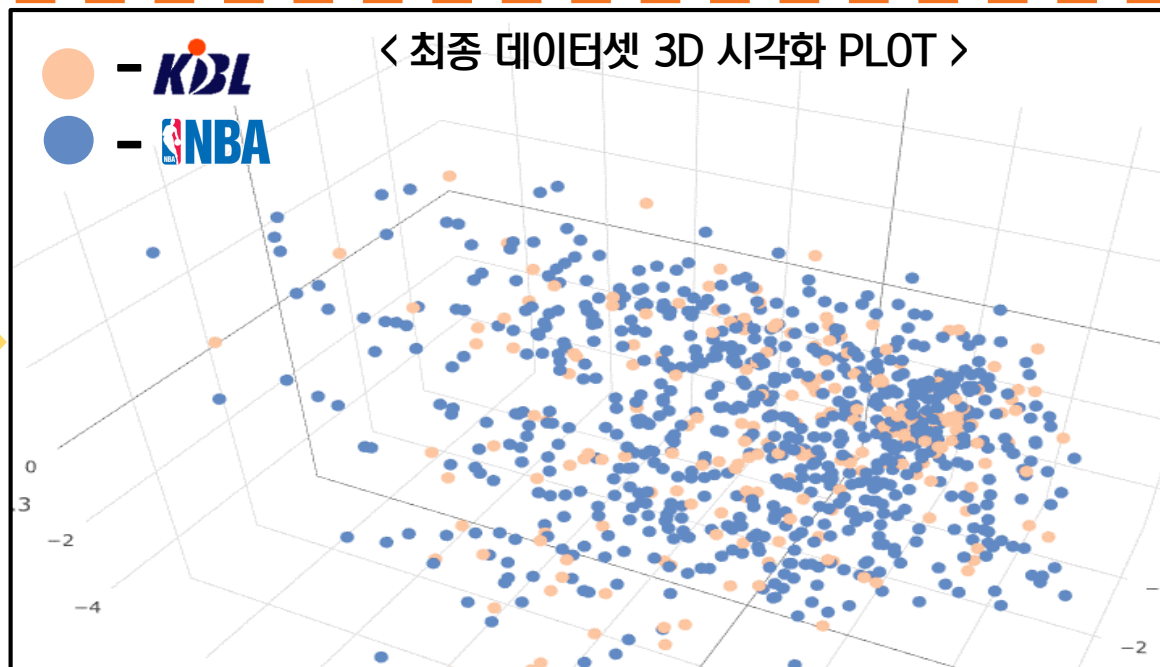
- PC성분 추출



```
KBL_norm <-  
left_join(  
  KBL[,names(NBA_mean[-23])] %>%  
  ungroup %>%  
  group_by(player_no, pos) %>%  
  summarise_if(is.numeric, mean) %>%  
  ungroup %>%  
  mutate(player_no = as.character(player_no)) %>%  
  mutate_if(is.numeric, scale) ,  
  KBL_name %>%  
  select(player_no, kname) %>%  
  unique %>%  
  mutate(player_no = as.character(player_no))) %>%  
  select(kname,2:23)
```



```
NBA_norm <-  
left_join(  
  NBA[,names(NBA_mean[-23])] %>%  
  ungroup %>%  
  group_by(player_no, pos) %>%  
  summarise_if(is.numeric, mean) %>%  
  ungroup %>%  
  mutate(player_no = as.character(player_no)) %>%  
  mutate_if(is.numeric, scale) ,  
  NBA_name %>%  
  select(player_no, kname) %>%  
  unique %>%  
  mutate(player_no = as.character(player_no))) %>%  
  select(ename,2:23)
```

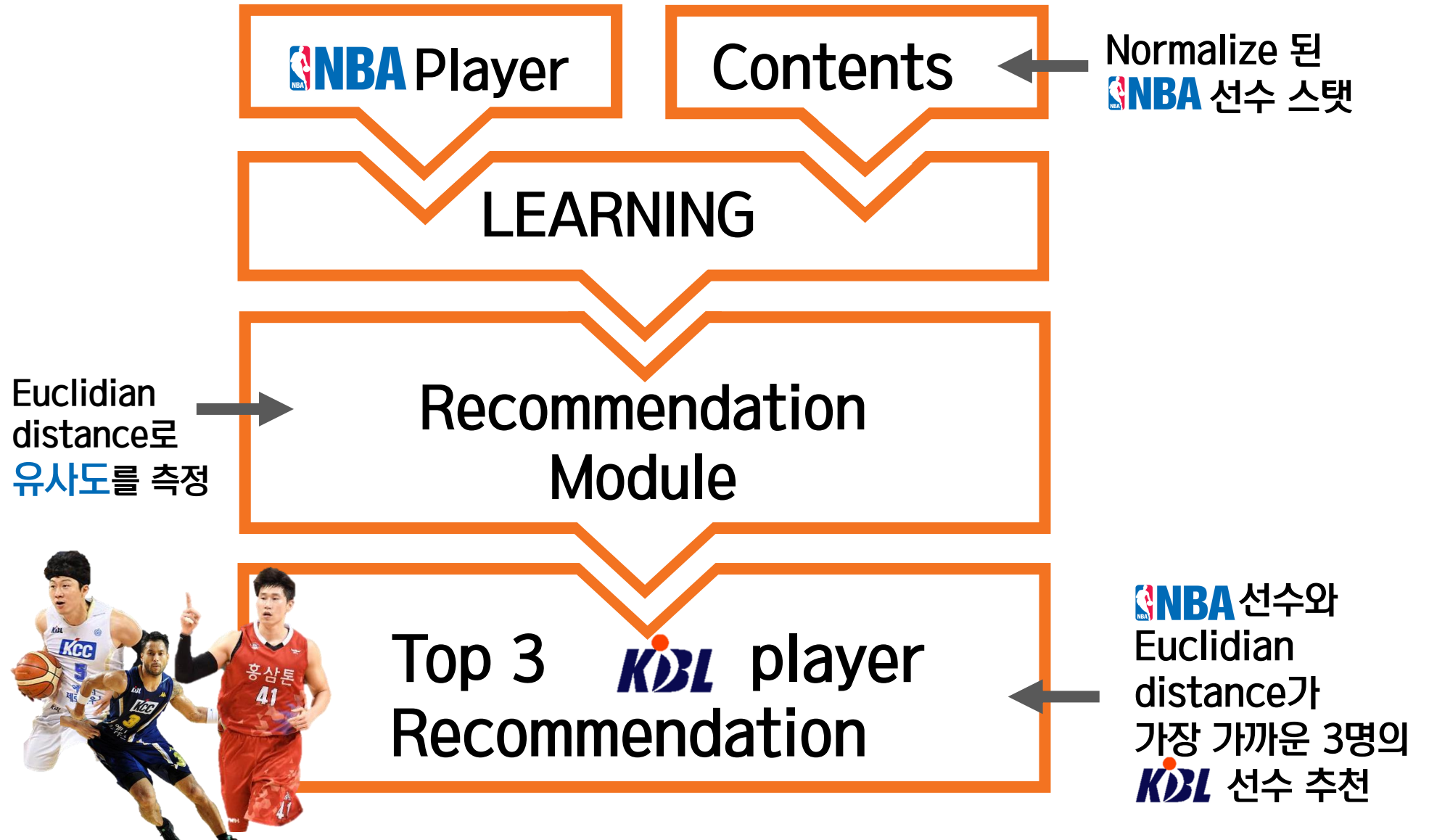


KBL 데이터와 NBA 데이터를  
Normalize 한 이후 최종 데이터셋 통합

# Contents Based Filtering

(1)  
Data  
Preprocessing

(2)  
Contents  
Based  
Filtering



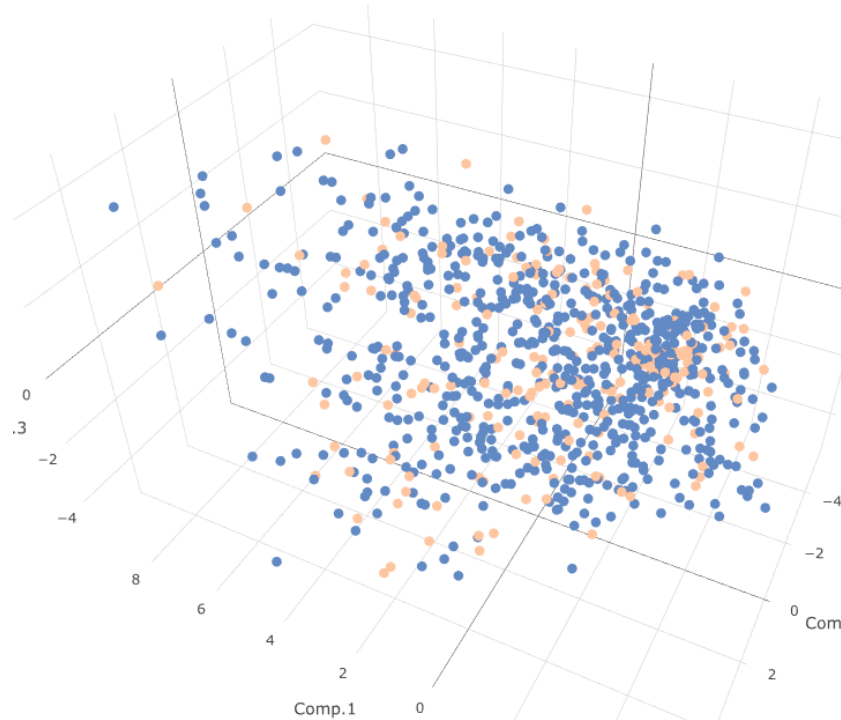
# Contents Based Filtering



NBA Player

Contents

LEARNING



전처리 과정을 통해 얻은  
Normalize된 선수들의 기록 데이터를  
이용하여

“ LEARNING ”

과정을 거침

# Contents Based Filtering

Contents (선수들의 경기 기록)을 Base 로 하여

입력된  선수들의 contents와

 선수들의 contents 사이의 **Euclidian Distance**를 계산

Recommendation  
Module

계산된 **Euclidian Distance**를 바탕으로 **유사도** 를 측정한 후  
이에 맞는  선수들을

Recommend!!!!

# Contents Based Filtering

(1)  
Data  
Preprocessing

(2)  
Contents  
Based  
Filtering

Recommendation  
Module 을 통해

입력된  NBA 선수와 가장 유사도가 높은

즉, 경기 기록측면에서 봤을 때,

플레이 스타일이 가장 비슷하다고 할 수 있는  선수 3명을 추천



Top 3  player  
Recommendation

# Contents Based Filtering

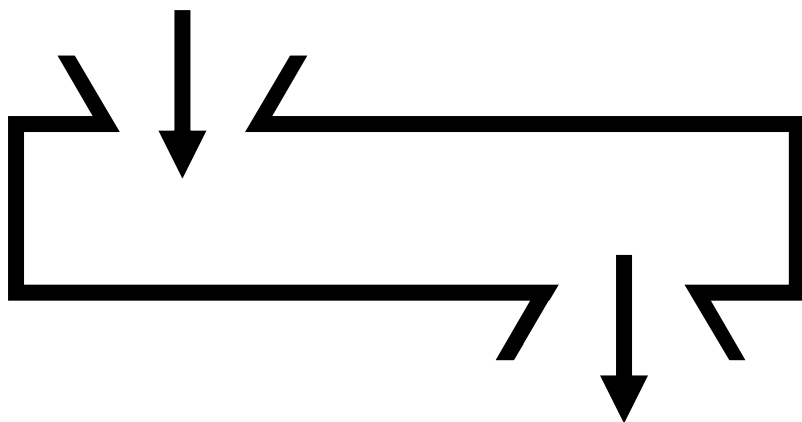
## < R 을 통한 실제 구현 >

위의 설명한

Contents based filtering의 아이디어를

직접 Function을 작성하여 구현

 **NBA** Player's name



Three  **KBL** Player's name

```
NBA_recommend_G <- function(nba_player_name){  
  number <- which(dat_norm_G$player_name == nba_player_name)  
  GD <-  
    dist_all_G[number,1:86] %>%  
    sort %>%  
    names() %>%  
    as.data.frame %>%  
    slice(1:3)  
  
  index <- GD$. %>% unfactor  
  dat_norm_G[index,'player_name']  
}
```

```
NBA_recommend_F <- function(nba_player_name){  
  number <- which(dat_norm_F$player_name == nba_player_name)  
  FD <-  
    dist_all_G[number,1:81] %>%  
    sort %>%  
    names() %>%  
    as.data.frame %>%  
    slice(1:3)  
  
  index <- FD$. %>% unfactor  
  dat_norm_F[index,'player_name']  
}
```

```
NBA_recommend_C <- function(nba_player_name){  
  number <- which(dat_norm_C$player_name == nba_player_name)  
  C <- dist_all_G[number,1:37] %>%  
    sort %>%  
    names() %>%  
    as.data.frame %>%  
    slice(1:3)  
  
  index <- C$. %>% unfactor  
  dat_norm_C[index,'player_name']  
}
```

# Contents Based Filtering

< R 을 통한 구현 >

(1)  
Data  
Preprocessing

(2)  
Contents  
Based  
Filtering

 Stephen Curry





```
R> NBA_recommend_G("Stephen Curry")
# A tibble: 3 x 1
  player_name
  <chr>
1 두경민
2 큐제이 피터슨
3 양동근
```

두경민



실제로

Function 에  NBA 의  
 Stephen Curry 선수의 이름을  
넣으면

Contents Based Filtering

에 의해, 가장 유사한 선수로 

두경민 선수를

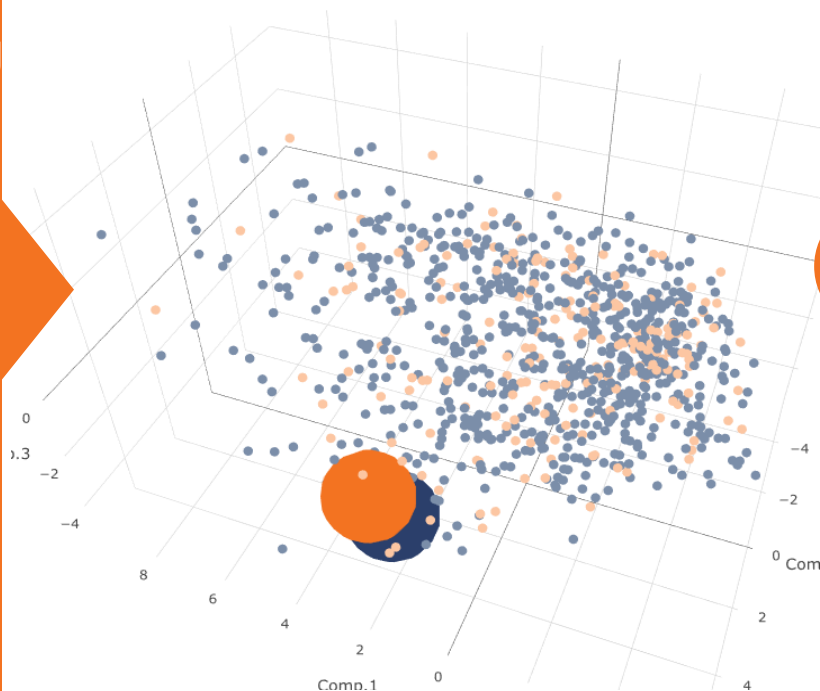
Recommend 해줌

# Contents Based Filtering

(1)  
Data  
Preprocessing

(2)  
Contents  
Based  
Filtering

< R 을 통한 구현 >



두 선수의 **Euclidian Distance** 를

Plot을 통해 확인해보면

매우 가까이 있음

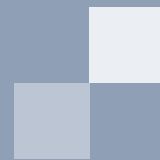


 **Stephen Curry** 선수와  
유사한 플레이 성향을 가진  
**KBL** 선수로  
**두경민** 선수를  
추천할 수 있다.



선수 유형 추천 시스템

# 4 활용방안 및 시뮬레이션



# 개요

(1)  
개요

(2)  
구체적 방안

(3)  
기대효과

(4)  
시뮬레이션

## 목표



농구 및 KBL에 대한  
자연스러운 흥미 도모

## 방안



개인별 최적 선수 추천  
알고리즘 구현

# 사용 도구

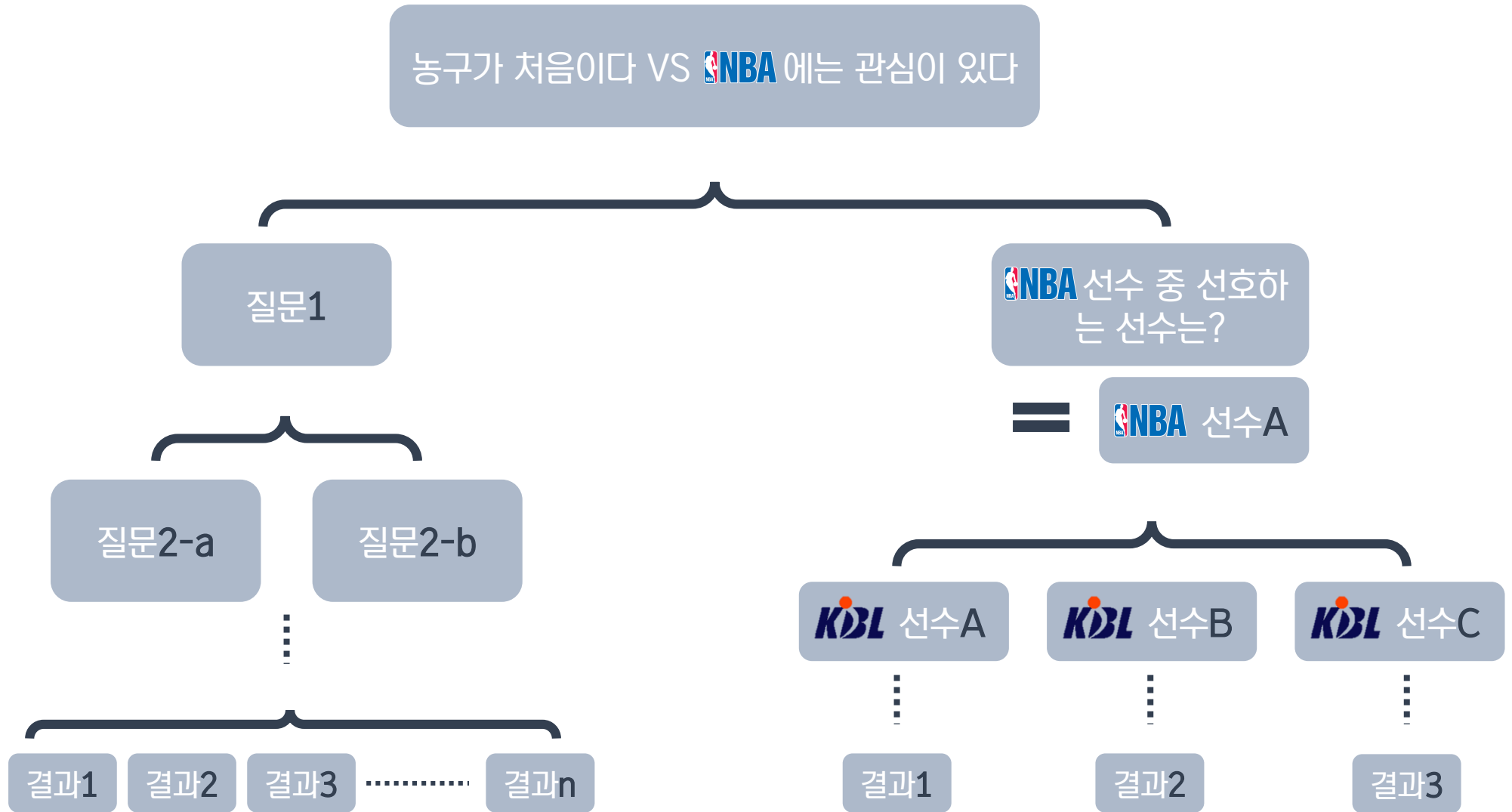
웹 이용, 잠재고객에게  
알고리즘 제공



접근성 및 제작용이성 측면에서  
모바일 어플리케이션 상회

	웹	모바일 어플
핸드폰 기종에 상관 없이 한 번 의 제작으로 작동가능한가?	0	X
작동에 설치가 필요 없는가?	0	X
PC에서 작동하는가?	0	X
수정이 용이한가?	0	X

# 기초 틀



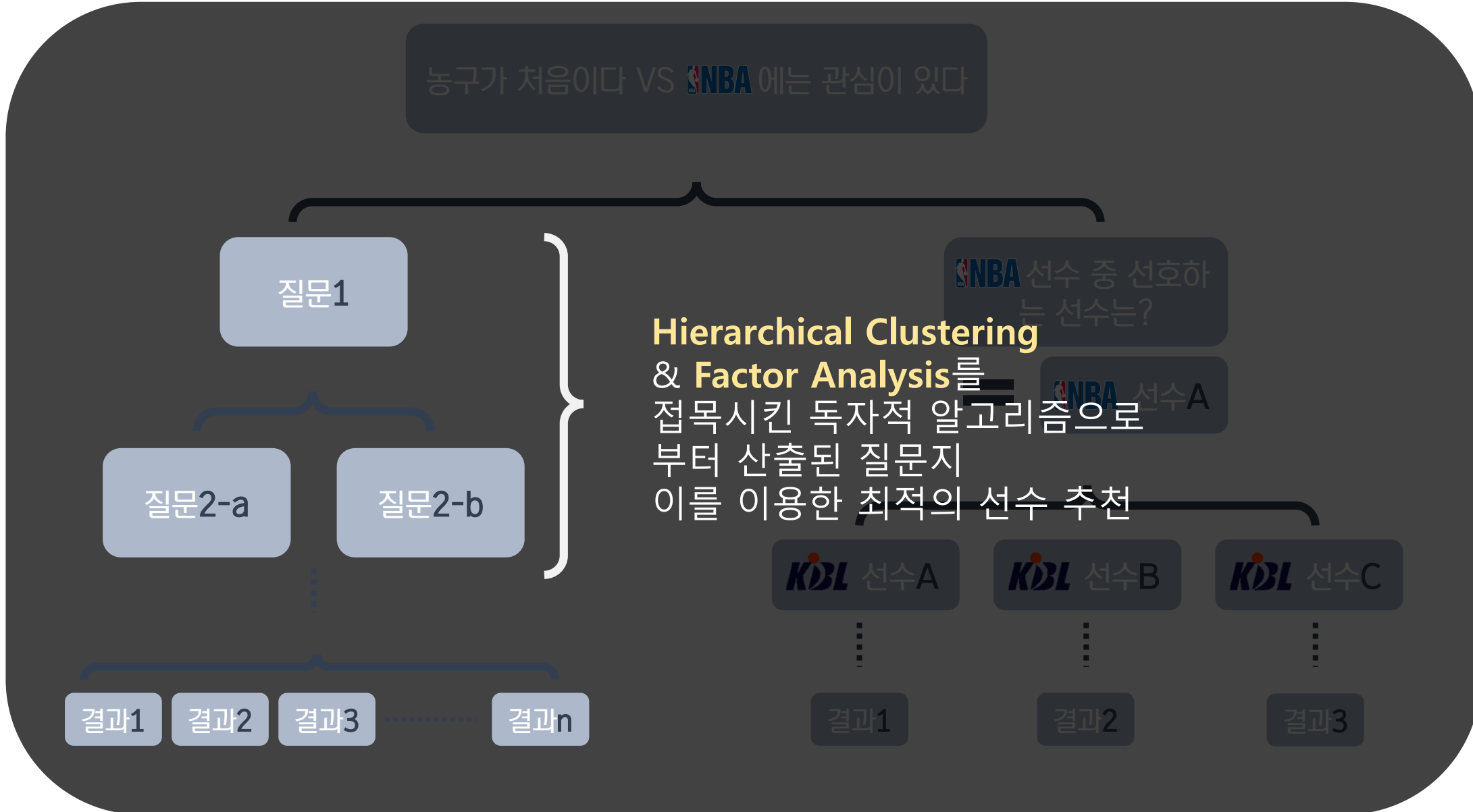
# 기초 틀

(1)  
개요

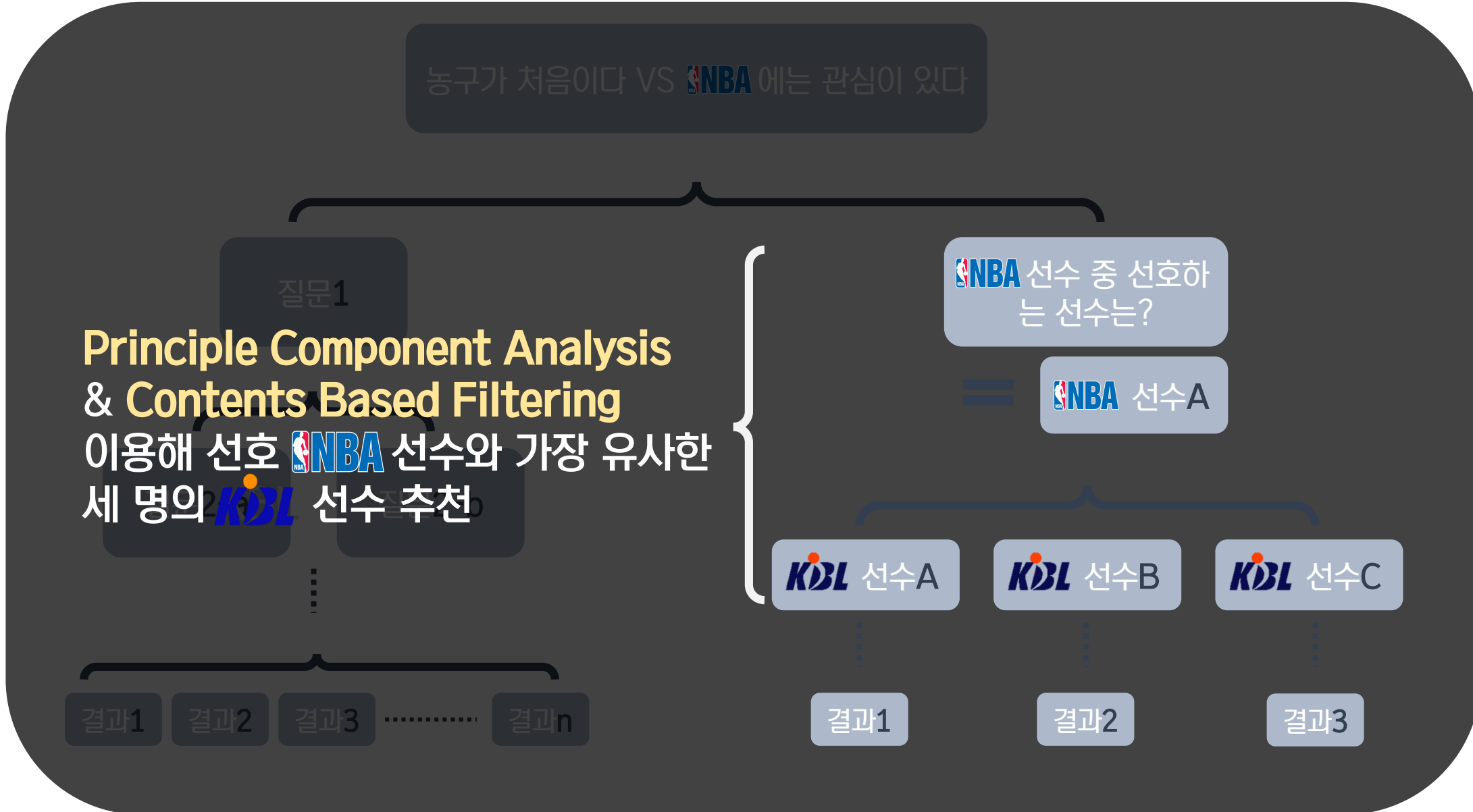
(2)  
구체적 방안

(3)  
기대효과

(4)  
시뮬레이션



# 기초 틀



# 기초 틀

(1)  
개요

(2)  
구체적 방안

(3)  
기대효과

(4)  
시뮬레이션

농구가 처음이다 VS NBA에는 관심이 있다

모든 질문지에는 농구경기 하이라이트 포함

질문1

+

최종 결과에서는 KBL 하이라이트 포함

질문2-a

질문2-b

=

잠재고객의 흥미 및 관심 유도

결과1

결과2

결과3

.....

결과n

결과1



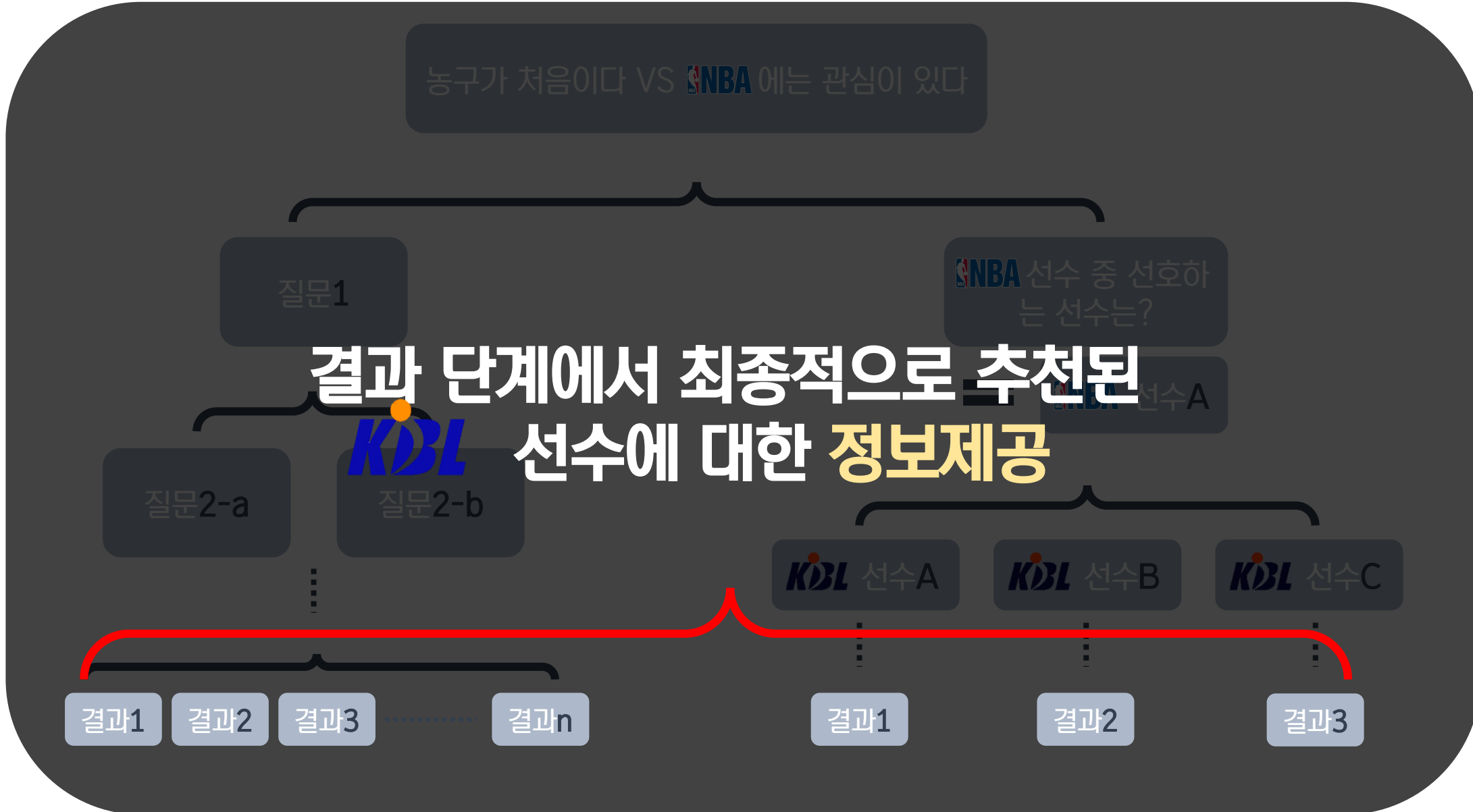
= NBA 선수A



KBL 선수A

C

# 기초 틀





## 결과 예시

(1)  
개요

(2)  
구체적 방안

(3)  
기대효과

(4)  
시뮬레이션



FORWARD  
NO.11  
허일영

생년월일 : 1985년 08월 05일  
신장 / 체중 : 195CM / 91KG  
혈액형 : O  
출신 학교 : 경남중 / 동아고 / 건국대  
발사 이즈 : 295MM  
포지션 : 포워드 (FORWARD)  
등번호 : 11  
드래프트 : 2009년 드래프트 1라운드 2순위

핀치히터  
3점 1위  
거미손

DEF	69	AST	84
ATK	88	3PT	99
SPD	75	BLK	64

당신의 선수를 실제로 보고싶다면...

다른 경기 영상이 보고 싶다면...

## 결과 예시

- (1) 개요
- (2) 구체적 방안
- (3) 기대효과
- (4) 시뮬레이션



FORWARD  
NO.11  
허일영

생년월일 : 1985년 08월 05일  
신장 / 체중 : 195CM / 91KG  
혈액형 : O  
출신 학교 : 경남중 / 동아고 / 건국대  
발사 이즈 : 295MM  
포지션 : 포워드 (FORWARD)  
번호 : 11  
드래프트 : 2009년 드래프트 1라운드 2순위

선수 하이라이트



선수 스탯

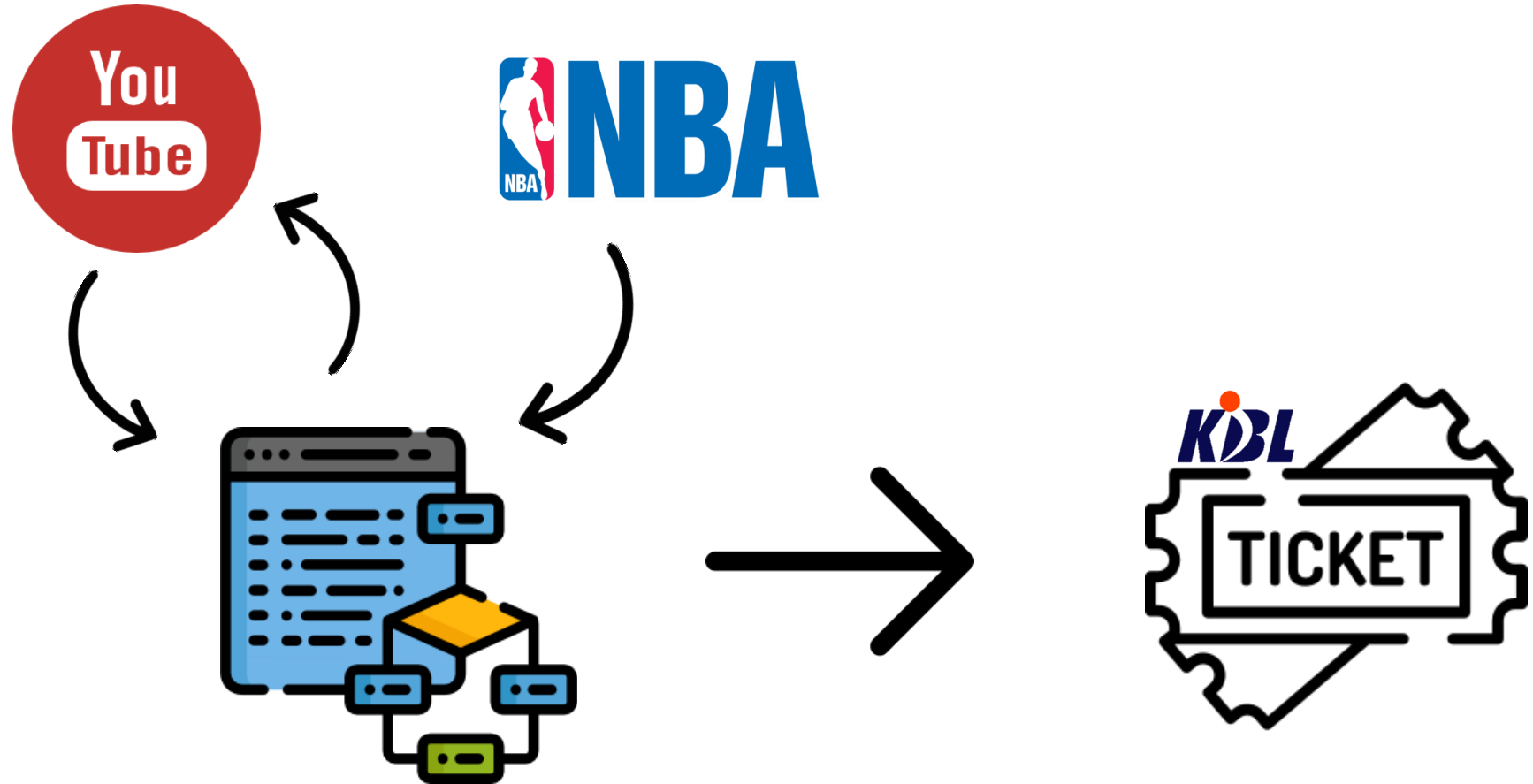
핀치히터	DEF 69	AST 84
3점 1위	ATK 88	3PT 99
거미손	SPD 75	BLK 64

당신의 선수를 실제로 보고싶다면...

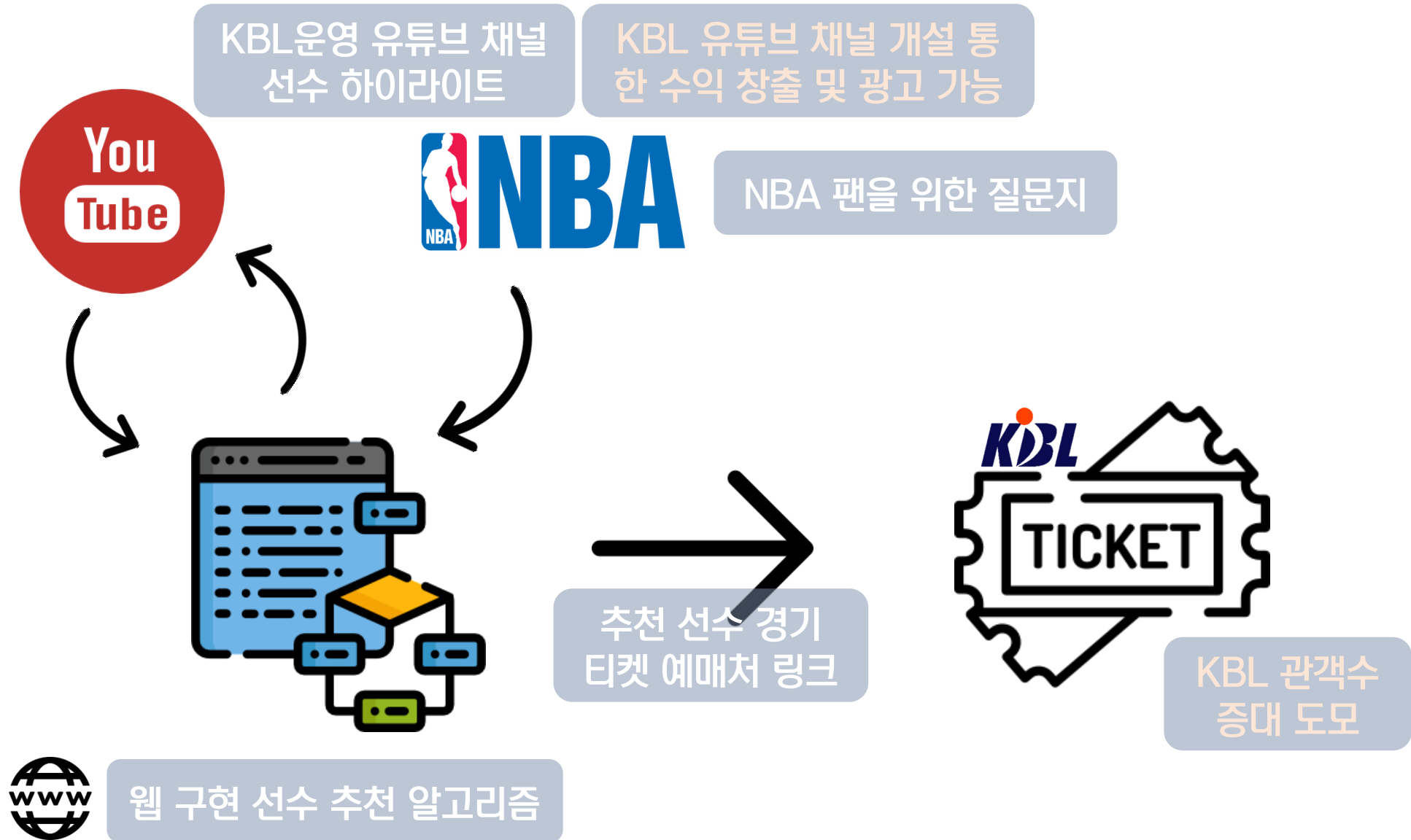
다른 경기 영상이 보고 싶다면...

해당 선수 하이라이트 유튜브 링크  
해당 선수 경기 티켓 예매 링크

## 기대 효과



## 기대 효과



## 기대 효과

(1)  
개요

(2)  
구체적 방안

(3)  
기대효과

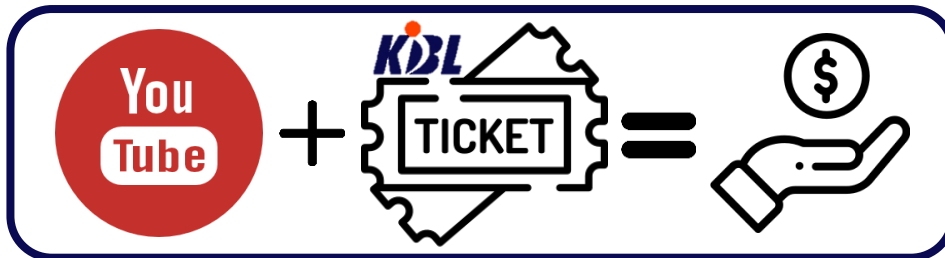
(4)  
시뮬레이션



방대한 팬을 보유한 유튜브, NBA 와  
KBL의 **교량** 역할을 수행함으로써 KBL  
로의 **지속적인 신규 팬 유입** 기대 가능



**KBL 공식 유튜브**를 운영하고 인기를 창  
출할 활로를 열고, 이를 활용해 트렌드에  
맞는 **다양한 홍보** 가능



웹 선수 추천 알고리즘 하나를 통해 KBL은  
**KBL 공식 유튜브**와 **KBL 경기 티켓**이라는  
두 가지 **수익 창출 Channel**을 활성화하게 됨



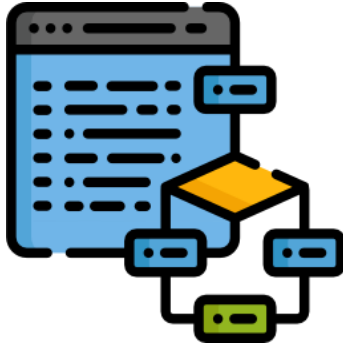
(1)  
개요

(2)  
구체적 방안

(3)  
기대효과

(4)  
시뮬레이션

# 시뮬레이션



[https://ddeolddeorumi.github.io/who\\_do\\_you\\_like\\_basket/](https://ddeolddeorumi.github.io/who_do_you_like_basket/)



Github Web Hosting을 이용한 간단한 구현

질문1 -> 질문2 -> 결과 -> 티켓 및 영상  
링크의 간단한 구성

선수 유형 추천 시스템

# 5 분석 의의 및 한계

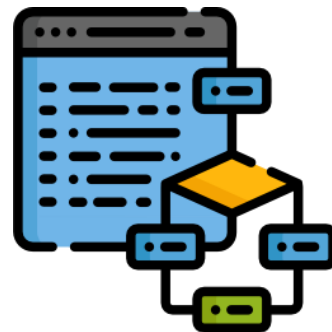


# 분석 의의

(1)  
분석 의의

(2)  
분석 한계

## 분석의의



요인 분석 및 계층 클러스터링 이용  
독자적 선수 추천 방식 고안



농구 선호층 및 비선호층 모두에게 간단한 질문지  
통한 최적 선수 추천, 나아가 KBL 관심 유도



단일 방법이 아닌 PCA, FA 등 활용, 데이터 또한  
NBA 데이터를 추가 이용해 추천 다양성 확보



# 분석 한계점

## 한계점



2시즌 데이터만을 이용함에 따라 좀  
더 정교한 분석을 하지 못했다는 점

감사합니다



Team 명륜동