회귀분석을 활용한
축구선수 이적료 예측모형

응용통계학과

201552043 전태양

목차

1. 서론

1) 분석 개요와 분석 목적

매년 축구 이적시장은 1년에 두차례 개방된다. 이적 시장은 타팀에서 선수를 영입하여 팀의 약점을 보강하거나 불필요한 선수를 매각해 구단의 예산 낭비를 방지하는 등 팀을 재정비하는 중요한 기간이다.

팀이 해당 기간을 성공적으로 마무리했는지에 대한 여부는 팀에 도움이 되는 좋은 선수를 영입했는지, 팀에 빈약한 포지션을 잘 보강했는지, 사전에 정해 놓은 방출 대상을 적절하게 방출했는지, 핵심 선수에 대한 계약을 잘 마쳤는지 등에 대한 내용을 가지고 판가름한다. 이때, 영입과 방출에 대해 고려되는 공통적인 요소로는 해당 선수에 대한 이적료 즉, 선수의 시장가치에 맞게 적절히 거래했는지에 대한 것이 큰 비중을 차지한다.

[그림 1] 올해 EPL 여름 이적시장 이적료 최다 지출 팀

재정이 탄탄한 빅클럽이라면 더 나은 성적을 위해 천문학적 금액을 쏟아서라도 선수 개인의 기록이라고 불리는 스탯(Stat)이 뛰어나거나 나이가 어리면서 앞으로의 성장가능성이 높은 유망주를 영입할 것이고, 빅클럽에 비해 상대적으로 투자 예산이 적은 중소형 클럽들은 실력은 좋은데 인지도가 낮거나, 전성기가 끝난 선수들을 적은 금액으로 영입하는 등 각자 클럽들 나름대로 기준을 정해 효율적이고 성공적인 이적시장을 일궈내고자 할 것이다.

이렇듯 본 분석에서는 선수를 매각하거나 영입하려는 구단은 선수의 어떤 정보를 기준으로 선수가치를 책정하고 이적을 확정하는지, 선수의 시장가치에 영향을 끼치는 주요

요인이 무엇인지 살펴보고 그에따라 다른 선수들의 시장가치를 예측해보고자 한다.

2. 데이터 설명

- 유럽 5대 리그에 소속된 축구 선수들의 정보와 능력치, 이적시장 가격이 포함된 데이터

- 총 8932명의 선수들의 정보가 담겨 있고 11개의 설명변수(축구선수들의 정보)와 1개의 반응변수(Value)로 구성되어 있다.
- 해당 데이터에 결측 값이 없었으며 중복 선수 역시 없었다.

<데이터 형태>

Id	Name	Age	Continent	Contract_until	...	Stat_potential	Stat_skill_moves	Value
0	L.Messi	31	South America	2021	...	94	4	110500000
3	De Gea	27	Europe	2020	...	93	1	72000000
7	L.Suarez	31	South America	2021	...	91	3	80000000
8	Sergio	32	Europe	2020	...	91	3	51000000

[표 1] 데이터 형식

- 선수 고유의 정보를 나타내는 ID와 Name을 제외한 변수 10개의 자료형은 실수형 6개, 문자형 4개로 구성되어 있으며, 변수의 종류는 범주형 변수 6개, 연속형 변수 4개로 구성되어 있다.

<변수 설명>

변수 명	변수 타입	설명
Id	실수형	선수 고유의 아이디
Name	문자형	이름
Age	실수형(연속형)	나이
Continent	문자형(범주형)	선수들의 국적이 포함되어 있는 대륙
Contract_until	문자형(범주형)	선수의 계약기간이 끝나는 시점
Position	문자형(범주형)	선호하는 포지션 ex) 공격수, 골키퍼
Prefer_foot	문자형(범주형)	선수가 선호하는 발 ex)오른발, 왼발
Reputation	실수형(범주형)	선수의 인지도, 높을수록 유명
Stat_overall	실수형(연속형)	선수의 현재 능력치
Stat_potential	실수형(연속형)	선수가 경험 밑 노력으로 발전할 수 있는

		정도
Stat_skill_moves	실수형(범주형)	선수의 개인기 능력치
Value	실수형(연속형)	FIFA가 선정한 선수의 이적 시장 가격(단위: 유로)

[표 2] 데이터 형식

<연속형 변수 범위>

변수 명	범위
Age	16 ~ 40
Stat_overall	47 ~ 94
Stat_potential	48 ~ 94
Value	10000 ~ 110500000

[표 3] 연속형 변수 scale

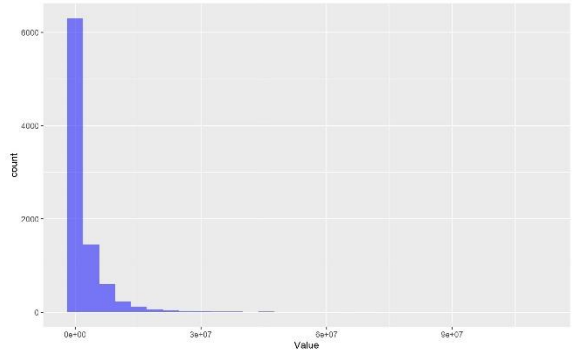
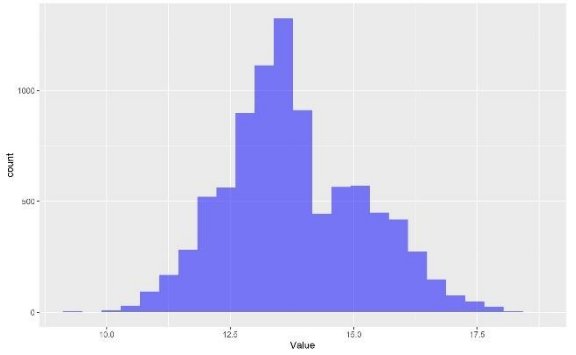
- 선수 고유명인 ID를 제외한 연속형 변수 4개의 Scale을 살펴보면 반응 변수인 Value의 Scale이 압도적으로 컸으며 분포 또한 넓게 퍼져 있었다.

3. EDA

3-1. 변수 탐색

<Value>

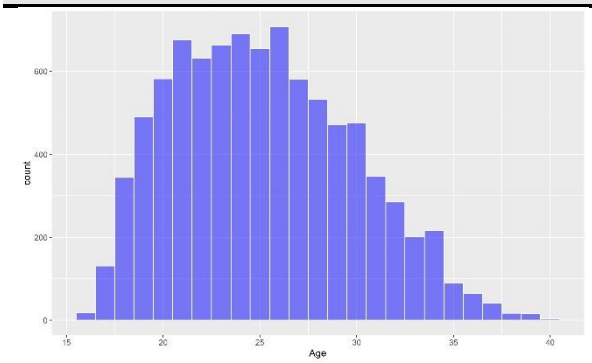
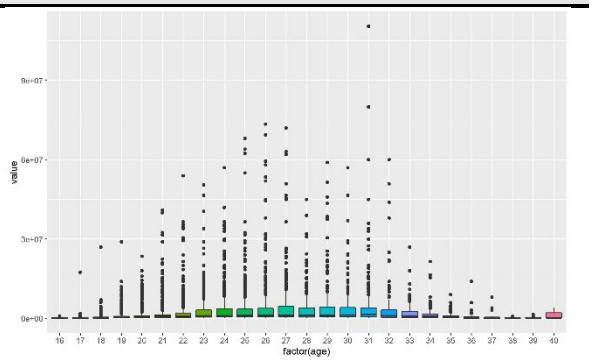
- 반응변수인 Value의 분포를 살펴보면 대다수의 분포가 왼쪽에 치우쳐져 있음을 확인할 수 있다.
- 데이터의 편차가 크기 때문에 log변환을 고려해볼 필요가 있다.

시장가치 분포	Log(시장가치) 분포
	

[그림 2] EDA1

<Age>

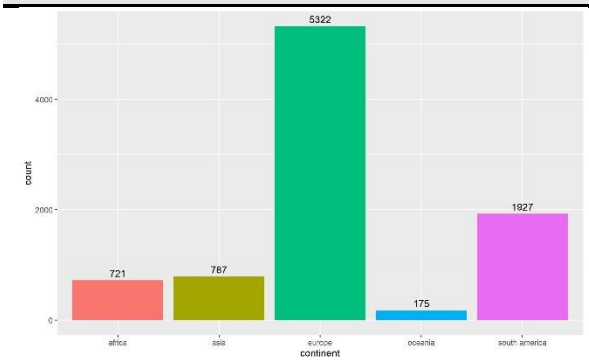
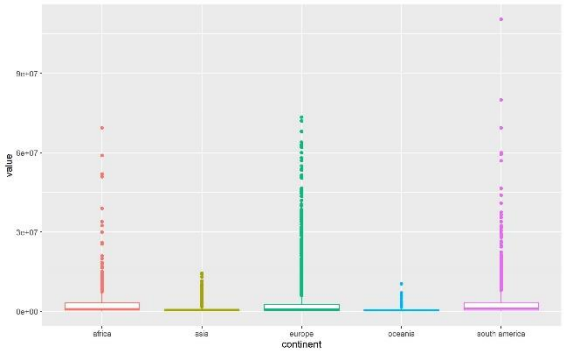
- 가장 나이가 적은 선수의 나이는 16세로 총 18명이다.
- 가장 나이가 많은 선수의 나이는 40세로 총 3명이다.
- 대체적으로 21세부터 32세까지의 시장가치가 높게 분포되어 있는 평균이 25.2, 표준편차가 4.64인 정규분포 형태를 띄며, 동나이대여도 시장가치의 격차가 크게 벌어지는 것을 확인할 수 있다.

나이별 선수 분포	나이별 시장가치 분포
	

[그림 3] EDA2

<Continent>

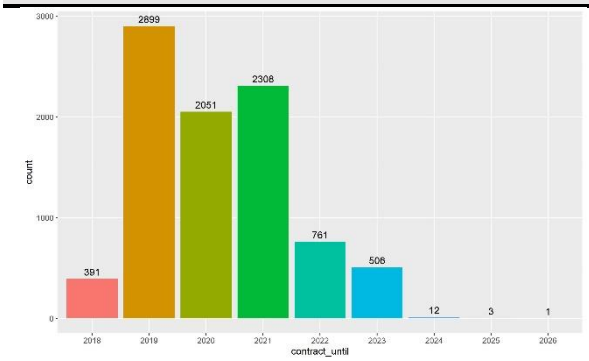
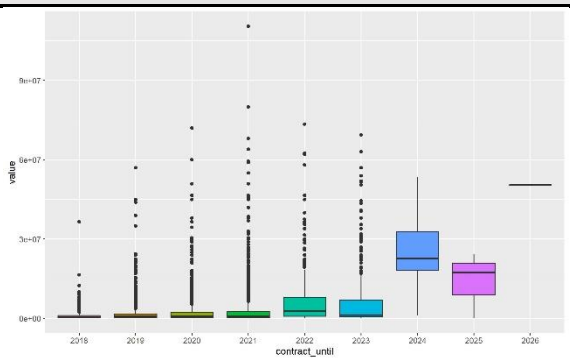
- 아프리카, 아시아, 유럽, 오세아니아, 남아메리카 대륙 5개의 범주로 구성된 범주형 변수다.
- 유럽 리그 기반의 선수들이다 보니 대부분의 선수가 유럽에 국적을 둔 선수들이었다.
- 아시아, 오세아니아를 제외하곤 시장가치 분포의 차이가 크게 벌어졌다.

대륙별 선수 수	대륙별 시장가치 분포
	

[그림 4] EDA3

<Contract Until>

- 2018 ~ 2026년까지 계약기간이 끝나는 시점을 나타내는 범주형 변수이다.
- 2023년 이후 계약자는 단 15명인 것으로 나타났다.
- 계약 만료까지 1년(2018년 기준) 남은 선수가 가장 많았으며 장기계약자들의 시장가치가 대체적으로 높게 나타났다.

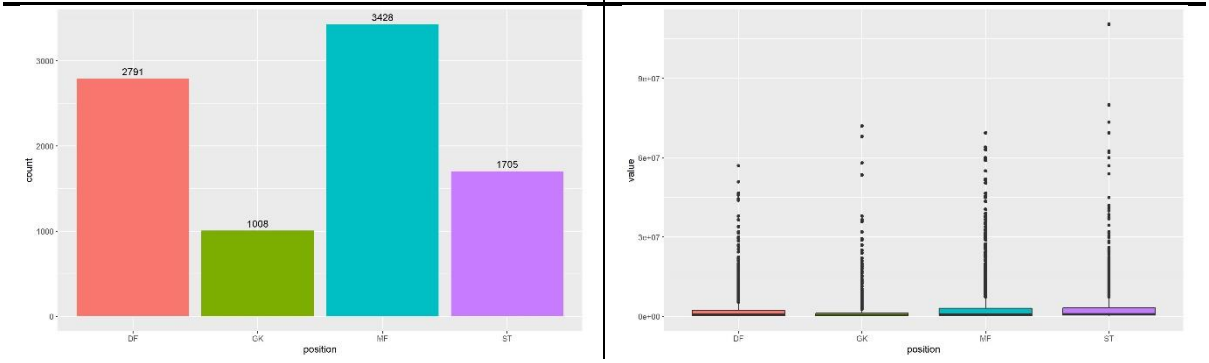
계약기간별 선수 수	계약기간별 시장가치 분포
	

[그림 5] EDA4

<Position>

- DF, GK, MF, ST 총 4개의 범주로 구성된 범주형 변수이다.
- 수비수와 미드필더의 선수 수가 공격수와 골키퍼의 수보다 약 2배 가량 많았지만, 포지션간에 시장가치 분포의 형태는 크게 나지 않았다.

포지션별 선수 수	포지션별 시장가치 분포

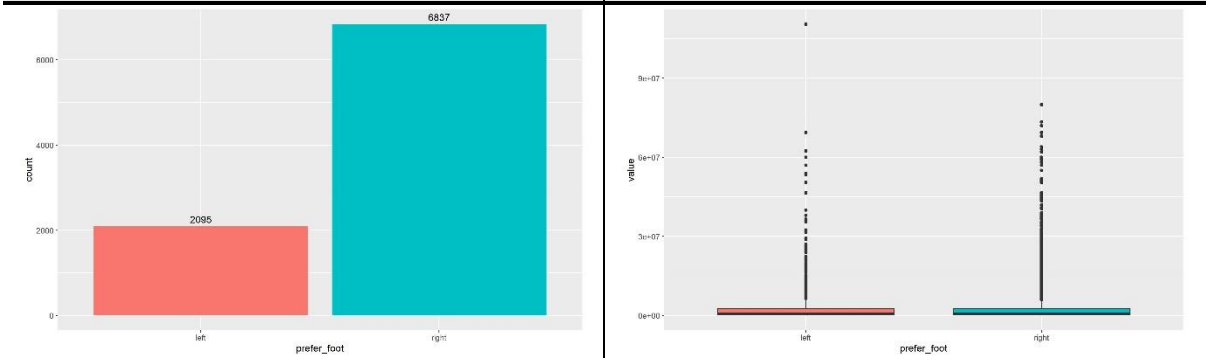


[그림 6] EDA5

<Prefer Foot>

- 주로 사용하는 발 오른발, 왼발 2개의 범주로 구성된 범주형 변수이다.
- 오른발을 사용하는 선수가 왼발을 사용하는 선수에 비해 3배 가량 많았지만, 시장가치 분포는 비슷한 형태로 나타났다.

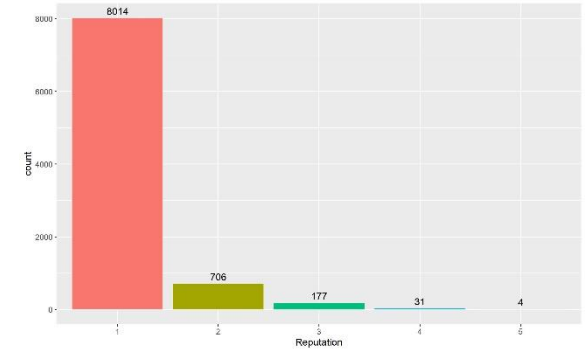
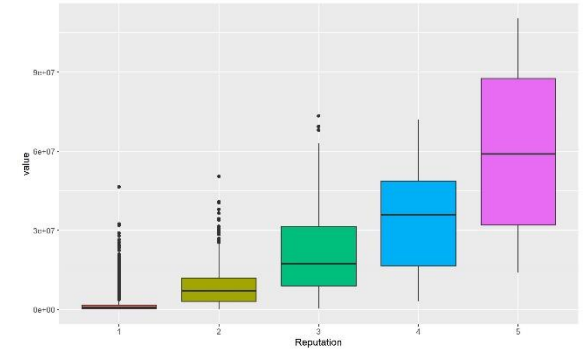
주로 사용하는 발에 따른 선수 수	시장가치 분포



[그림 7] EDA6

<Reputation>

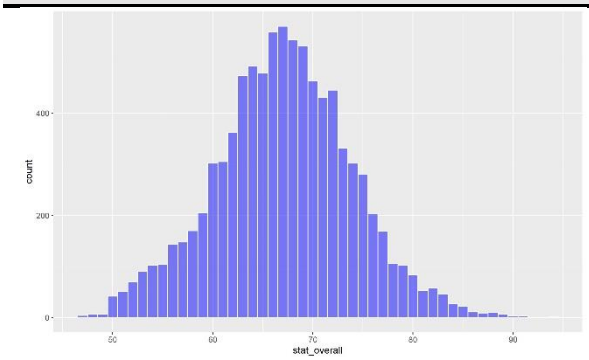
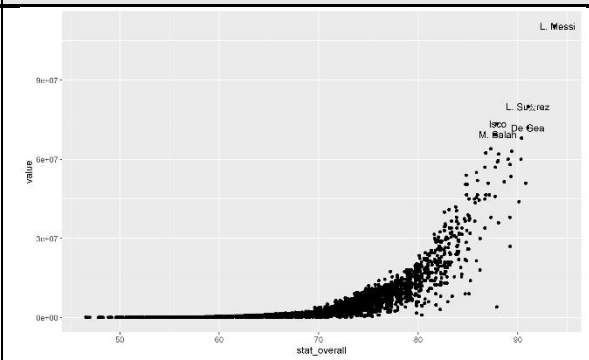
- 선수의 인지도를 5등급으로 분리한 범주형 변수이다.
- 선수의 인지도가 4, 5등급인 선수는 35명으로 인지도가 낮은 선수들이 대다수를 차지했으며, 인지도가 높을수록 시장가치의 평균도 높아지는 경향이 나타났다.

인지도 별 선수 수	인지도별 시장가치 분포
	

[그림 8] EDA7

<Stat Overall>

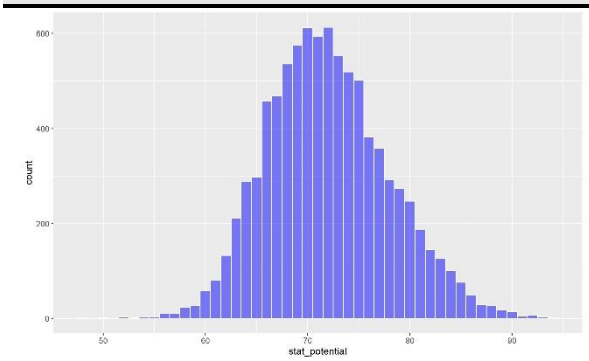
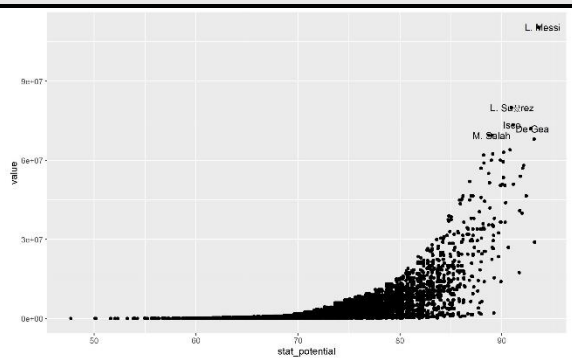
- 선수의 스탯(Stat)을 수치화한 연속형 변수이다.
- 선수 능력치가 가장 높은 선수는 L.Messi이며 94이다.
- 선수 능력치가 가장 낮은 선수들은 총 4명으로 47이다.
- 선수 능력치 분포는 평균은 67.091, 표준편차는 6.85인 정규분포 형태를 띄며, 능력치가 70이상인 구역부터 선수간 시장가치 격차가 커지는 것을 볼 수 있다.
- 선수 능력치가 94로 가장 높은 Messi의 시장가치와 다른 선수들간 시장가치 격차가 유독 큰 차이를 보인다.

선수 능력치 별 선수 분포	선수 능력치 별 시장가치 분포
	

[그림 9] EDA8

<Stat Potential>

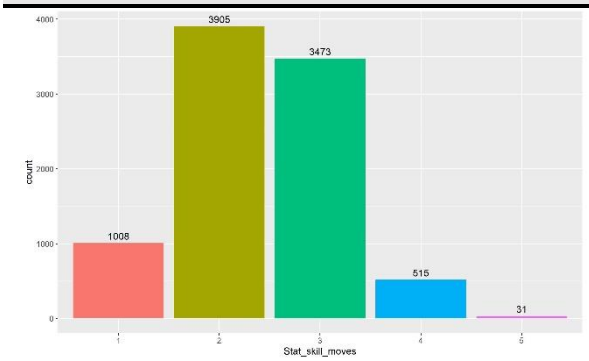
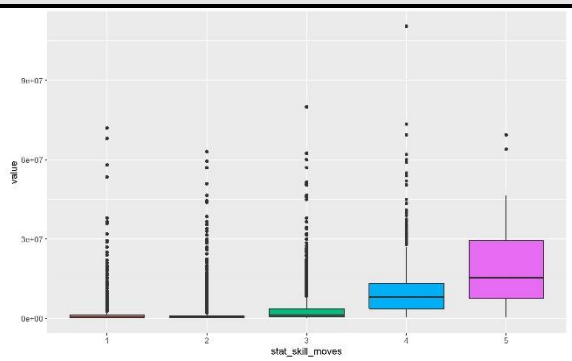
- 선수의 성장 가능한 능력치를 수치화한 연속형 변수이다.
- 선수의 잠재력 수치가 가장 높은 선수는 L.Messi이며 94이다.
- 선수의 잠재력 수치가 가장 낮은 선수는 Y.Uchimura이며 48이다.
- 선수 잠재력 수치 분포는 평균이 71.997, 표준편차는 5.988인 정규분포 형태를 띄며, 선수 능력치와 마찬가지로, 잠재력이 70이상부터 선수간 시장가치 격차가 커지고 있다.
- 전반적으로 선수의 능력치와 비슷한 분포를 보이는 것을 알 수 있다.

선수 잠재력 수치별 분포	선수 잠재력에 따른 시장가치 분포
	

[그림 10] EDA9

<Stat Skill Moves>

- 선수의 개인기 능력치를 1부터 5까지 등급별로 분리한 범주형 변수이다.
- 개인기 능력치가 2 ~ 3등급인 선수들이 대다수로 나타났다.
- 개인기 능력치가 1~3등급인 선수들과 4~5등급인 선수들의 시장가치는 차이가 있음을 알 수 있으며, 대다수를 차지한 2 ~ 3등급의 시장가치 분포는 비슷한 것을 알 수 있다.

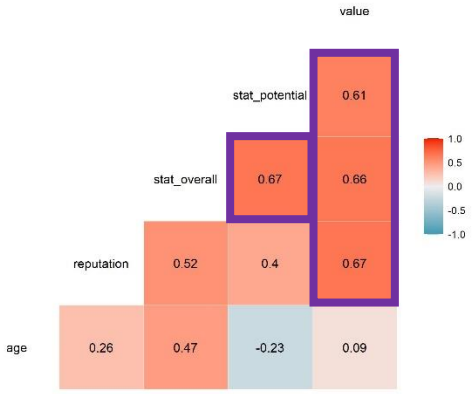
선수 개인기 능력치 분포	선수 개인기 수준별 시장가치 분포
	

[그림 11] EDA12

변수 탐색 요약

1. 전반적으로 value의 scale 격차가 크다 보니, 설명변수가 같은 수준이라도 value의 격차가 큰 구간들이 나타났다.
2. 나이, 특정 대륙, 계약기간, 선수 능력치, 잠재력, 개인기 변수에서 시장가치 분포에 차이가 나타났다.

상관계수



[그림 12] 상관계수

- 선수의 인지도, 능력치, 잠재력이 반응변수 value와 양의 상관관계로 나타났다.
- 앞서 변수 탐색에서 봤던 것처럼 능력치와 잠재력이 강한 양의 상관관계로 나타나고 있으며, 인지도와 능력치도 양의 상관관계로 나타났다.

4. 회귀분석

먼저 데이터셋을 학습시킬 Train데이터와 예측에 쓰일 Test데이터로 분리했다. 이때, 각 선수들의 Position 비율을 기준으로 층화 추출하였다.

4-1. 자료 분리 : Train(80%)와 Test(20%)데이터로 분리

Train data	8932개의 데이터 중 7047개의 row
Test data	7047개를 제외한 1785개 row

[표 4] 분리한 데이터 셋

4-2. 회귀모형 적합

Formula : Value ~ . -ID, -NAME(고유 번호와 이름을 제외한 모든 변수)
유의한 변수 : age, contract_until, reputation, overall, potential
Adj R^2 : 0.6492
AIC : 235566.2
BIC : 235676.2
P-value : < 2.2e-16

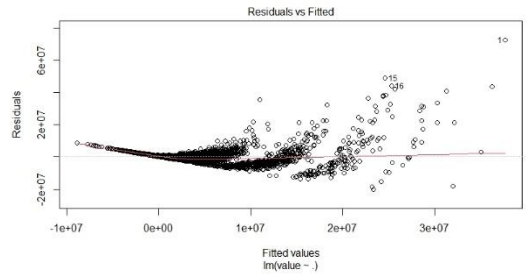
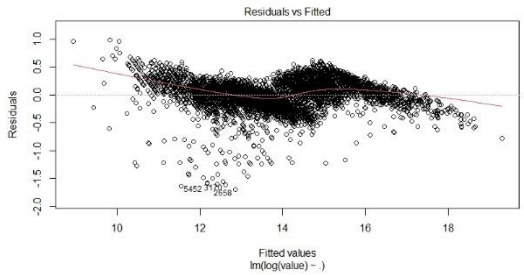
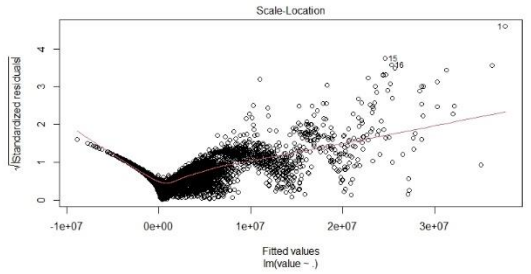
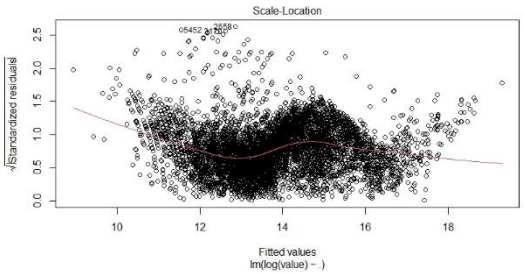
[표 5] 회귀분석 결과

1) P-value가 <2.2e-16으로 통계적으로 유의하며 약 65%의 설명력을 가진다.
그러나 EDA에서 살펴본 것처럼 반응변수(value)의 scale이 매우 커서 오차의 min값(-20304453)과 max(72853546)의 값이 상당히 큰 것을 확인할 수 있었고, 따라서 이상치가 존재하거나 등분산성을 만족하지 못할 것으로 판단했다.

2) potential의 회귀계수가 -56771로 value와 음의 상관관계를 이룬다는 회귀 결과가 나왔다. 그러나 EDA에서 potential이 양의 상관관계임을 확인했기 때문에 설명변수들이 서로 연관되어 있는 다중 공선성을 의심해 볼 필요가 있다.

등분산성 확인

1)을 기반으로 모형의 등분산성을 확인해본 결과 분산이 점점 커지는 것을 확인할 수 있었다. 따라서 value를 log변환하여 scale을 줄여주는 작업을 진행하였다.

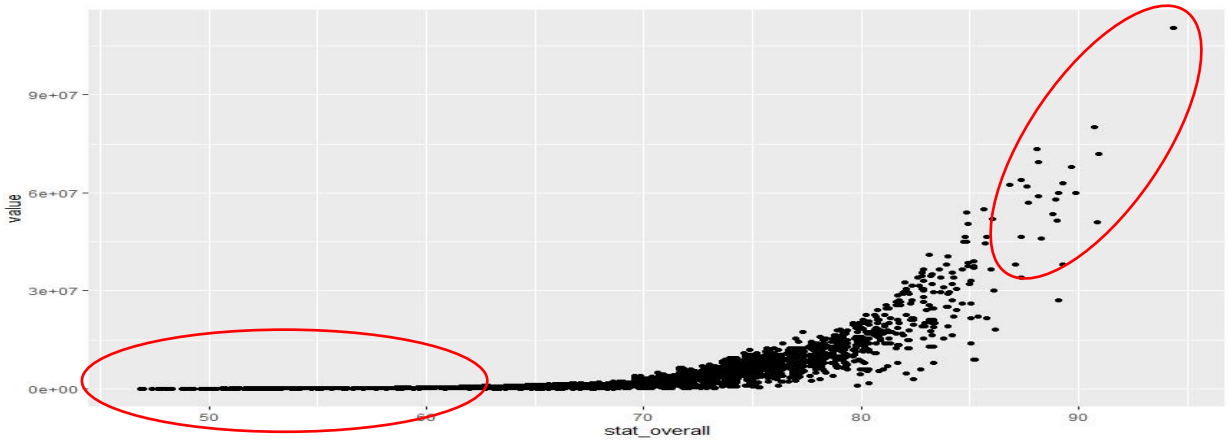
로그 변환 전	로그 변환 후
	
	

[표 6] 로그변환 결과

Value의 변환을 통해 비교적 분산이 안정화된 것을 확인할 수 있다. 다만, 변환 후에도 여전히 이상치가 존재하는 것을 확인할 수 있다.

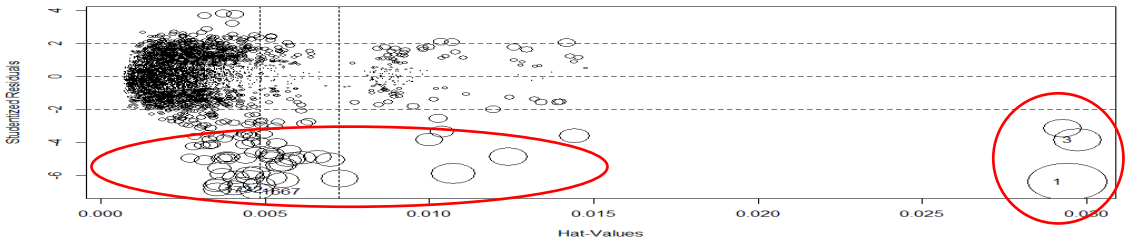
이상치 탐지

Log 변환으로 scale을 줄인 이후에도 수많은 이상치들이 검출되었다. 검출된 이상치들은 대부분 빨간 도형에서 발생한 것을 확인하였다. 이상치들의 대부분이 분포의 끝자락이므로 우측은 IQR Rule(q3 + 1.5 * IQR)을 활용하여 Value가 6150000이상, 좌측은 IQR Rule(q1 – 1.5 * IQR)을 활용하면 음수(-3050000)가 나오기 때문에 10000이하로 임의로 설정하여 해당 범위를 벗어나는 경우 이상치로 간주하였다. 이렇게 하면 추후 예측 결과는 나빠질 수 있으나, 이상치로 인한 영향력을 줄일 수 있기에 범위를 벗어난 954명의 선수를 데이터에서 제거하였다.



[그림 13] 이상치 탐색 결과1

이상치들을 제거한 이후에 회귀분석을 재차 진행한 결과 아래 그림과 같이 여전히 몇 몇의 이상치들이 존재했다.



[그림 14] 이상치 탐색 결과 2

위와 같은 이상치들은 또 다른 측정지표인 Cook's Distance를 활용하여 영향력을 판단했다. Cook's Distance를 사용하여 이상치를 판단할 때는 일반적으로 Cook's D가 Cooks' D 평균의 3배를 넘어갈 경우 이상치로 간주했다.

Cook's Distance $> 3 \times$ Mean(Cook's Distance)

[표 6] Cook's Distance를 활용한 이상치 범위

2번의 이상치 탐색을 거쳐 1147명의 선수 정보를 제거하였고 변수 선택을 진행하였다.

4-3. 변수 선택

우리는 지금까지 9개의 변수를 가지고 회귀모형을 설정하였다. 회귀분석에서는 많은 변수를 활용하여 예측변수로 사용할 수 있는데 더 많은 변수를 추가한다고 해서 반드시 좋은 모형이 나온다는 것은 아니다. 복잡한 모형보다는 비교적 단순한 모형이 더 설명력이 높게 나올 수도 있고, 변수 선택을 통해 적은 설명변수를 사용하면서 앞서 양의 상관관계임에도 Potential의 회귀 계수가 음수가 나온 것과 같은 다중 공선성 문제를 해결할 수도 있다.

변수선택을 하는 기준으로는 AIC, BIC, adjR^2가 있는데 AIC와 BIC는 공통적으로 모형에 변수를 추가할수록 페널티를 부여하여 모형의 품질을 평가하며, adjR^2는 추가된 변수가 모형의 설명력에 도움이 될 경우 증가한다.

모형 선택 및 단계적 회귀	선택된 변수
AIC에 의한 전진선택(M1)	Stat overall + continent + age + position + stat potential + stat skill moves + continent + reputation
AIC에 의한 후진소거	AIC에 의한 전진선택과 동일
BIC에 의한 전진선택(M2)	Stat overall + age + position + stat potential + stat skill moves + reputation
단계적 회귀에 의한 선택 (M3)	age + position + stat overall

[표 7] 변수 선택

변수 선택 결과 3가지의 모형이 채택되었고 3개의 모형의 AIC, BIC, adjR^2를 비교했다.

AIC	BIC	AdjR^2
M1 : -3201.85	M1 : -3101.564	M1 : 0.9684
M2 : -3186.96	M2 : -3120.099	M2 : 0.9683
M3 : -3071.07	M3 : -3024.271	M3 : 0.9677

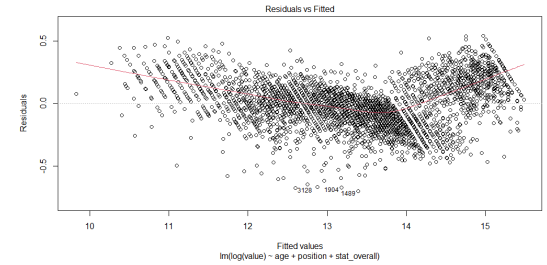
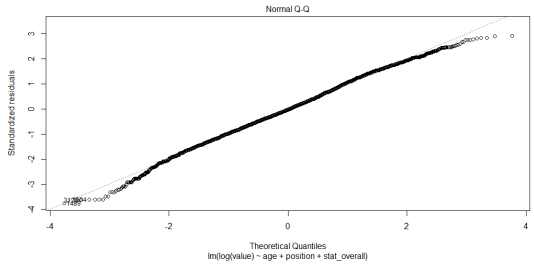
[표 8] 3가지 모형 결과 값

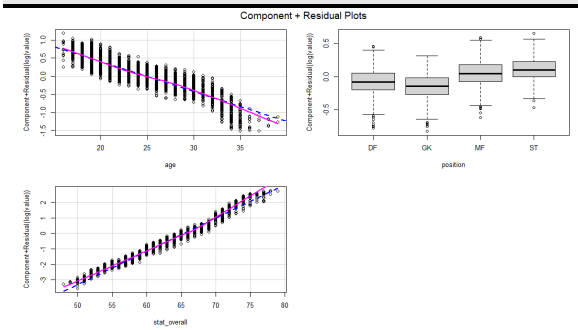
3가지 모형의 기준이 큰 차이가 없어 다중 공선성을 확인하는 지표인 VIF를 확인했고, VIF가 5이상인 변수가 없는 M3를 최종 모형으로 설정했다.

4-4. 최종 모형

Log(Value) ~ -0.0825*Age + -0.0711*PositionGK ... + 0.214*Stat Overall

최종 모형 가정 만족 여부 확인

등분산성 가정	정규성의 가정
	

선형관계의 가정	VIF		
	Age	Position	Overall
	1.47	1.03	1.48

[표 9] 최종 모형 가정 만족 여부

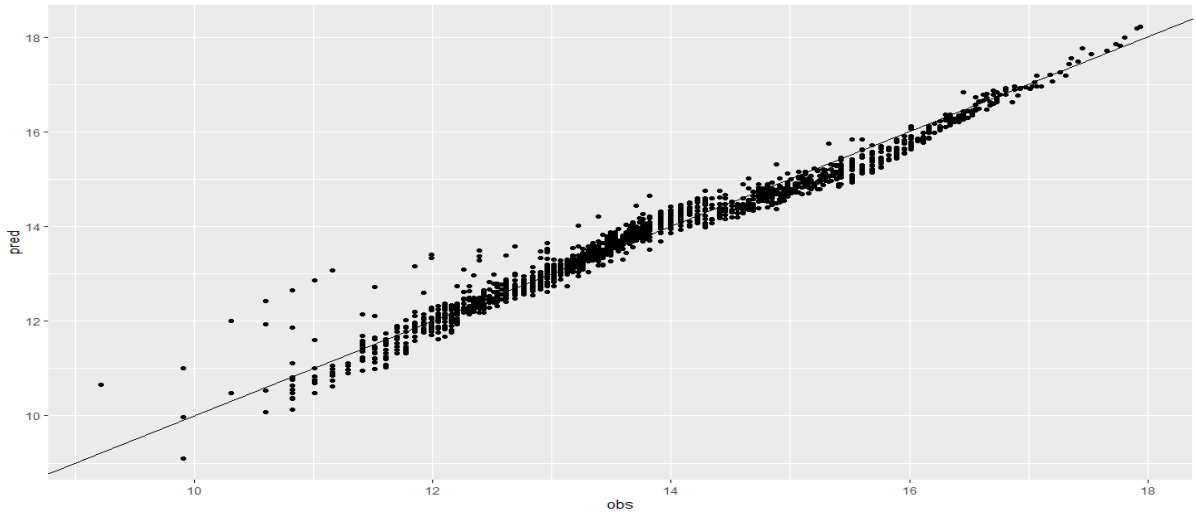
최종 회귀 모형의 가정들을 살펴본 결과, 모두 오차항의 가정을 만족하였으며, 설명변수와 반응변수의 선형관계도 만족하였다.
다중 공선성 역시 크게 문제가 발생하지 않았다.

5. 최종 모형으로 Test 데이터 예측

Train data의 최종 모형을 활용하여 선수의 나이와 포지션, 능력치로 분석 목적이었던 Test data의 선수 시장가치를 예측했다.

RMSE	Rsquared	MAE
0.255	0.9673	0.1801

[표 10] 예측 결과



[그림 15] 예측 결과2

[그림 16] 예측 결과3

선수별 흐름은 비슷하나 실제값보다 높게 예측하거나 낮게 예측하는 경향을 보인다.

6. 결론

분석을 진행한 결과 선수의 나이와 포지션, 능력치가 선수의 시장가치에 영향을 끼치는 것으로 나타났다.

그에 따라 변수 선택에 따른 최종 모형을 선정하여 분석 목적이었던 선수들의 시장가치를 예측했다.

선수 별 시장가치 흐름은 비슷해 보이지만 실제보다 낮게 예측하거나 높게 예측하는 구간이 보인다는 아쉬움이 남았다. Train 데이터의 반응변수를 좀 더 전처리를 진행하거나, 새로운 설명 변수를 추가, 이상치로 인식한 구간을 완화시키는 등의 추가 작업이 필요할 것으로 보인다.

부록

데이터 출처 : 데이콘(dacon) 해외축구선수 이적료 예측

https://dacon.io/competitions/open/235538/overview/description

code

```
library(tidyverse)
library(ggplot2)
library(GGally)
library(caret)
library(leaps)
library(car)
library(MASS)


## 데이터 불러오기
data <- read.csv('D:/Portfolio/통계자료분석/FIFA_train.csv')
data

## 데이터 요약
str(data)

## 데이터 통계
summary(data)

## data 중복 확인 : 중복이 없음
data %>% duplicated() %>% table()

# 데이터에서 값이 0 인 비중
data %>% summarise(zero_rate = mean(value == 0))

## 데이터 결측값 확인 0 개인 걸 확인
sum(is.na(data))

## 계약기간 변수 변환
result <- vector('character', length=nrow(data))

for(i in 1:nrow(data)){
  ifelse(str_detect(data$contract_until[[i]],'-'),
         result[i] <- paste0('20',substr(data$contract_until[[i]],8,9)),
         result[i] <- data$contract_until[[i]])
  result[i] <- result[i]
}

data$contract_until <- result
```

```
######### EDA###############
#### value
data %>% ggplot(aes(x = value)) + geom_histogram(bins = 25, fill = 'blue', alpha = 0.5) +
  xlab('Value') + ggsave('value histogram.jpg',dpi = 300)

data %>% ggplot(aes(x = log(value))) + geom_histogram(bins = 25, fill = 'blue', alpha = 0.5) +
  xlab('Value') #+ ggsave('log value histogram.jpg',dpi = 300)

#### age
data %>% filter(age == max(data$age)) %>% select('name')
data %>% filter(age == min(data$age)) %>% select('name','age')

mean(data$age)
sd(data$age)
data %>% ggplot(aes(x = age)) + geom_histogram(fill = 'blue', colour = 'white', alpha = 0.5, bins = 25) +
  xlab('Age') + ggsave('age histogram.jpg', dpi = 300)

data %>% ggplot(aes(x = factor(age), y = value, fill = factor(age))) + geom_boxplot() +
  theme(legend.position = 'none') +
  ggsave('age value boxplot.jpg', dpi = 300)

#### continent
data %>% ggplot(aes(x = continent, fill = continent))  + geom_bar() +
  geom_text(stat = 'count', aes(label = ..count..), vjust = -0.5) +
  theme(legend.position = 'none') +
  ggsave('continent barplot.jpg', dpi = 300)

data %>% ggplot(aes(x = value, fill = continent)) + geom_histogram()

data %>% ggplot(aes(x = continent, y = value, color = continent)) + geom_boxplot() +
  theme(legend.position = 'none') +
  ggsave('continent boxplot.jpg', dpi = 300)

#### contract_until
data %>% ggplot(aes(x = contract_until, fill = contract_until)) + geom_bar() +
  geom_text(stat = 'count', aes(label = ..count..), vjust = -0.5) +
  theme(legend.position = 'none') + ggsave('contract barplot.jpg', dpi = 300)

data %>% ggplot(aes(x = contract_until, y = value, fill = contract_until)) + geom_boxplot() +
  theme(legend.position = 'none') +
  ggsave('contract boxplot.jpg', dpi = 300)

#### position
data %>% ggplot(aes(x = position, fill = position)) + geom_bar() +
  geom_text(stat = 'count', aes(label = ..count..), vjust = -0.5) +
  theme(legend.position = 'none') + ggsave('position barplot.jpg', dpi = 300)
```

```
data %>% ggplot(aes(x = position, y = value, fill = position)) + geom_boxplot() +
  theme(legend.position = 'none') +
  ggsave('position boxplot.jpg', dpi = 300)


### prefer foot
data %>% ggplot(aes(x = prefer_foot, fill = prefer_foot)) + geom_bar() +
  geom_text(stat = 'count', aes(label = ..count..), vjust = -0.5) +
  theme(legend.position = 'none') + ggsave('prefer_foot barplot.jpg', dpi = 300)

data %>% ggplot(aes(x = prefer_foot, y = value, fill = prefer_foot)) + geom_boxplot() +
  theme(legend.position = 'none') +
  ggsave('prefer_foot boxplot.jpg', dpi = 300)

### reputation
data %>% ggplot(aes(x = factor(reputation), fill = factor(reputation))) + geom_bar() +
  geom_text(stat = 'count', aes(label = ..count..), vjust = -0.5) + xlab('Reputation') +
  theme(legend.position = 'none') + ggsave('reputation barplot.jpg', dpi = 300)

data %>% ggplot(aes(x = factor(reputation), y = value, fill = factor(reputation))) + geom_boxplot() +
  theme(legend.position = 'none') + xlab('Reputation') +
  ggsave('reputation boxplot.jpg', dpi = 300)

### stat_overall
data %>% filter(stat_overall == max(stat_overall)) %>% select(name, stat_overall)
data %>% filter(stat_overall == min(stat_overall)) %>% select(name, stat_overall)
best_overall <- data %>% arrange(desc(value))
best_overall <- best_overall[1:5,]
best_overall
mean(data$stat_overall)
sd(data$stat_overall)

data %>% ggplot(aes(x = stat_overall)) +
  geom_histogram(bins = 48,fill = 'blue', colour = 'white', alpha = 0.5) + ggsave('stat_overall hist.jpg', dpi = 300)

data %>% ggplot(aes(x = stat_overall, y = value)) + geom_jitter() +
  geom_text(aes(label = name), data = best_overall) + ggsave('stat_overall scatter.jpg', dpi = 300)


### stat_potential
data %>% filter(stat_potential == max(stat_potential)) %>% select(name, stat_potential)
data %>% filter(stat_potential == min(stat_potential)) %>% select(name, stat_potential)

mean(data$stat_potential)
sd(data$stat_potential)

data %>% ggplot(aes(x = stat_potential)) +
```

```
  geom_histogram(bins = 47,fill = 'blue', colour = 'white', alpha = 0.5) + ggsave('stat_potential hist.jpg', dpi = 300)

data %>% ggplot(aes(x = stat_potential, y = value)) + geom_jitter() +
  geom_text(aes(label = name), data = best_overall) + ggsave('stat_potential.jpg', dpi = 300)

### stat_skill_moves


data %>% ggplot(aes(x = factor(stat_skill_moves),   fill = factor(stat_skill_moves))) + geom_bar() +
                    geom_text(stat = 'count', aes(label = ..count..), vjust = -0.5) +
                    theme(legend.position = 'none') + xlab('Stat_skill_moves') + ggsave('stat_skill_moves.jpg', dpi = 300)

data %>% ggplot(aes(x = factor(stat_skill_moves), y = value, fill = factor(stat_skill_moves))) + geom_boxplot() +
  theme(legend.position = 'none') + xlab('stat_skill_moves') +
  ggsave('skill boxplot.jpg', dpi = 300)

### pairplot
data %>% select(!id & !name & !stat_skill_moves) %>% ggcorr(label = TRUE, label_round = 2) +
  ggsave('data pairplot.jpg', dpi = 300)

### 회귀분석 ##########
### 계약기간 변수 변환 ex)2018 년 기준 계약기간이 2020 년 만료이면 2020 - 2018 = 2
result <- vector('character', length=nrow(data))

for(i in 1:nrow(data)){
  ifelse(str_detect(data$contract_until[[i]],'-'),
         result[i] <- paste0('20',substr(data$contract_until[[i]],8,9)),
         result[i] <- data$contract_until[[i]])
  result[i] <- as.integer(result[i]) - 2018
}
data$contract_until <- as.integer(result)



## train test split ##
x.id <- createDataPartition(data$position, p = 0.8, list = FALSE)
train <- data %>% slice(x.id)
test <- data %>% slice(-x.id)

train_data <- train %>% dplyr::select(!id & !names)
test_data <- test %>% dplyr::select(!id & !names)

## lr ##
fits <- lm(value ~ ., train_data)
summary(fits)
vif(fits)
AIC(fits)
```

```
BIC(fits)

plot(fits,1)
plot(fits,3)

# log(lr)
fits <- lm(log(value) ~ ., train_data)
summary(fits)
vif(fits)
AIC(fits)
BIC(fits)

plot(fits,1)
plot(fits,3)

## 이상치 탐색 ##
q3 <- quantile(train_data$value,0.75)
q1 <- quantile(train_data$value,0.25)
iqr <- q3 - q1
ad <- q1 - (1.5*iqr)
ap <- q3 + (1.5*iqr)
ad
ap

train_data2 <- train_data %>% filter(value < ap & value > 10000)

length(train_data$value) - length(train_data2$value)

fit1 <- lm(log(value) ~ ., train_data2)
summary(fit1)
plot(fit1,1)




fit2 <- lm(log(value) ~ age + position+stat_overall, train_data2)
summary(fit2)
plot(fit2,2)

influencePlot(fit2)

plot(fit2,4)

### cooks'd 이상치 탐색
cooksD <- cooks.distance(fit2)
influential <- cooksD[(cooksD > (3 * mean(cooksD, na.rm = TRUE)))]
influential
```

```
names_of_influential <- names(influential)
outliers <- train_data2[names_of_influential,]
train_without_outliers <- train_data2 %>% anti_join(outliers)
model2 <- lm(log(value) ~ position + age + stat_overall, data = train_without_outliers)
summary(model2)

plot(model2,2)
influencePlot(model2)
plot(model2,1)
plot(model2,3)
plot(model2,4)
vif(model2)
AIC(model2)
BIC(model2)

### 모델 예측 ###
pred_s <- predict(model2, newdata = test_data)
defaultSummary(data.frame(obs = log(test_data$value),

                          pred = pred_s))

data.frame(obs = test_data$value, pred = pred_s) %>%
  rownames_to_column(var = "name") %>%
  ggplot(aes(x = obs, y = pred)) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1) +
  geom_text(aes(label = name),

            nudge_x = 0.3, nudge_y = 0.2)

df <- data.frame(obs = log(test_data$value), pred = pred_s)
df$test_name <- test$name
df$idx <- seq(from = 1, to = length(df$obs))
rownames(df) = test$name


 ## 100 개의 데이터만 확인 ##
df[1:100,] %>% ggplot() + geom_line(aes(x = idx, y = obs, color = 'Real')) +
  geom_line(aes(x = idx, y = pred, color = 'Pred'))
```