2021-1학기 AI+X:R-PY컴퓨팅 기말보고서

학과 경영학부

학번 2018027303

이름 손정범

1. 연구 목적

필자는 축구의 광팬이다. 특히 해외축구 팀 중 첼시의 엄청난 팬이다. 지난 5월 30일 2020-2021 시즌의 대미를 장식하는 UEFA챔피언스리그 결승전이 진행되었다. 잉글랜드 프리미어리그 소속의 맨시티와 첼시가 맞대결을 펼쳤는데, 365bet과 같은 유럽의 유명한 배팅사이트, 공신력 있는 언론사 등에서 일제히 맨시티의 우위를 예상했다. 그도 그럴 것이 맨시티는 세계 최고의 명장인 펩 과르디올라 감독의 지휘 하에 이번 시즌 프리미어리그과 EFL컵에서 우승을 차지했으며, 챔피언스리그 8강과 4강에서 각각 강력한 우승후보로 점쳐졌던 팀들인 바이에른 뮌헨과 파리 생제르맹을 꺾고 결승에 진출했기 때문이다. 반면 첼시는 리그에서 겨우 4위를 기록했으며, 시즌 막바지에 흔들리는 모습을 보였다. 하지만 결과는 첼시의 1대0승리로 첼시가 9년만에 유럽챔피언 자리에 등극했다. 이것이 바로 축구이다. 몸값이 높고 실력이 출중한 스타플레이어들이 즐비하다고 반드시 승리를 장담할 수 없는 스포츠가 바로 축구이다. 15-16 레스터시티의 동화 같은 EPL우승, 그리스의 유로2004 등의 상징적인 언더독의 우승을 비롯해 04-05 이스탄불의 기적, 11-12 첼램덩크 등 다른 스포츠에 비해 수도 없이 많은 이변이 발생하는 종목이다.

하지만 경기를 측정할 수 있는 다양한 수단과 더불어 기술 발전의 영향으로 근래들어 축구를 데이터화하려는 움직임이 가속화되고 있다. 최근 여러 축구전문 통계사이트들을 통해 득점, 실점, 점유율, 패스성공률 같은 클래식한 스탯뿐만 아니라 키패스, 기대득점 등 점차 더 세분화된 팀 스탯 및 선수별 개인 스탯을 확인할 수 있다. 필자가 축구를 보기 시작한 이래로 가장 강력했던 팀은 2008년부터 2012년 무렵까지의 FC바르셀로나이다. 볼 소유와 점유율을 기반으로 한 '티키타카'라고 불리우는 패스게임으로 공간을 점유하며 압도적인 경기력을 보여주었던 강력한 팀이었다. 올타임 넘버원으로 거론되는 축구의 신 메시를 비롯해 세얼간이 사비, 이니에스타 그리고 부스케츠가 지키고 있는 중

원까지, 바르셀로나는 참가한 모든 대회에서 우승을 기록하여 당대 최고의 팀이었다. 바르셀로나의 축구철학은 볼을 빼앗기는 즉시 5초안에 다시 뺏어와 끊임없는 패스워크를 통해 점유를 하는 점유율에 기반을 하고 있다. 우리가 볼을 오래 가지고 있는만큼 우리의 득점 기회는 늘어나고 상대의 득점기회는 그만큼 적어지는 아이디어에서 출발한 것이다. 이것이 점유율이 중요한 이유이다. 축구는 전후반 45분, 총 90분 내에 상대보다 많은 득점을 기록해야 이길 수 있는 게임이다. 즉 제한된 90분이라는 시간 내에 공을 오래 가지고 있는 것이 유리할 수밖에 없다. 축구는 필드 위에 있는 22명의 선수의 역량도 중요하지만, 이들을 잘 조립하여 써먹는 감독의 역량에 따라서 결과가 크게 좌지우지되는 스포츠이다. 큰 틀에서 주도적인 전략을 설정할 것인지, 반응적인 전략을 설정하는 것이 전술 수립의 1단계이다. 과거에 비해 볼 점유의 중요성이 줄었다고 할지라도 여전히 점유율은 매우 중요한 팀스탯 중 하나이다.

그러나 높은 점유율이 반드시 승리라는 결과물을 가져다주는 것은 아니다. 점유율 대신 최근에는 xG라는 수치가 뜨겁게 주목을 받고 있다. xG는 expected Goals의 약자로 골 기댓값 혹은 골 기대확률이라고 번역할 수 있다. 이러한 xG값을 활용하는 방법은 실제 득점수치와 xG값을 비교하는 것이다. 실제 득점이 xG값보다 높다면, 득점이 예상되지 않는 어려운 상황에서도 높은 골결정력을 발휘하여 많은 득점을 기록했다고 볼 수 있으며, 실제 득점이 xG보다 낮다면 골을 넣어야 하는 쉬운 찬스를 많이 놓치고 있는 것이다. 높은 xG값이 반드시 실제 득점으로 이어져 경기의 승리로 연결되는 것은 아니며 모든 상황들이 철저하게 가정된 완벽한 데이터가 아니다. 하지만 축구계에서 보다 다양한 데이터들이 확보되고, 소위 말하는 경기력 혹은 경기영향력을 측정할 수 있는 많은 지표들이 개발됨에 따라 이들의 선구자격인 xG값은 한 팀의 승률을 파악할 수 있는 충분한 의미를 가지고 있는 데이터라고 생각한다.

이번 연구에서는 축구 경기에 관한 여러 데이터들이 경기승률에 미치는 영향을 파악하고 감독의 입장에서, 혹은 선수로써 보다 많은 경기를 이기기 위해서는 전술적으로 어떠한 점을 연구하고 보완해야 하는지 파악해보고자 한다. 특히 앞서 언급한 클래시컬한 스탯인 점유율과 최근에 떠오르고 있는 기대득점값을 중심으로 분석을 진행하려고 한다.

2. 문헌 조사

문헌조사들을 통해 축구관련 지표의 중요성과 이들이 경기의 승리에 있어서 어떠한 영향을 미치는 지를 파악할 수 있었다. 다음의 글은 빅데이터를 활용한 축구선수 평가 모델을 제시하였다.

『빅데이터를 활용한 축구 선수 평가 모델』

글_김윤후 중앙대학교 컴퓨터공학과 석사과정 소속_지식공학연구실

이번 연구는 팀스탯을 중심으로 경기결과와의 연관성을 파악하는 것에 목적이 있기 때문에 데이터를 기반으로 선수를 평가하고자 하는 위 글과 직접적으로 연관이 있는 것은 아니다. 하지만 포지션 별로 평가에 필요한 데이터를 구분하여 독립변수와 종속변수를 설정하고 다중회귀분석을 실시한 후에 상관관계를 분석했다는 점에서 필자가 얻을 수 있는 힌트들이 있다고 판단하였다.

1) "점유율과 승리 사이에 정말 상관관계가 있을까", 유현태 기자, SPOTV NEWS, 2017.10.27

2) "xG란 무엇인가? : 새로운 선수 평가 방법", 최유진 기자, FOOTBALL TRIBE, 2017.11.18

1번 기사의 경우, 필자와 마찬가지로 프리미어리그에 소속된 클럽들을 표본으로 삼아 점유율이 높은 팀의 승리횟수, 무승부횟수, 패배횟수를 파악하고 이를 비율로 파악하여 최상위권 팀은 점유율을 높이 유지하면서 성적을 내지만, 그 외의 팀들은 점유율이 결과로 연결되는지 장담할 수 없다는 결론을 도출하였다. 2번 기사는 xG라는 지표에 대한 구체적인 개념과 의의에 대해 설명해주고 있다. 위의 2개의 기사들은 분석모델을 설정하여 직접적으로 상관관계를 분석한 연구는 아니지만 이번 연구의 대주제인 점유율과 xG값에 관한 기사로 본격적인 데이터 수집 및 분석에 들어가기에 앞서 연구의 방향성을 결정하는데 도움을 주었다.

3. 데이터 수집

본 연구에 활용된 데이터는 2020-2021 잉글랜드 프리미어리그 20개팀의 경기결과 및 세부 팀 스탯들이다. 데이터를 수집하기 위해서는 XML, JSON, 웹크롤링 등의 방식을 활용할 수 있으나, 아무래도 한국어로 된 사이트에서 데이터를 확보하기에는 한계가 있어 해외의 축구전문 통계사이트들을 활용하여 웹크롤링을 진행하였다. 데이터를 얻기 위해 접속을 시도한 사이트는 크게 다음 2곳이다.

1) 후스코어드닷컴 : https://1xbet.whoscored.com/

2) FBREF : https://fbref.com/en/ / https://fbref.com/en/comps/9/Premier-League-Stats

1번 사이트의 경우, 정말 많은 데이터들을 보유하고 있으며 실제로 국내외 많은 기자나 해설위원들이 참고를 하고 있는 사이트이다. 하지만 웹데이터를 크롤링하는 과정에서 IP문제 등으로 인해 접속이 제한되는 경우가 종종 있었다. 또한 접속이 원활하게 이루어지더라도 id명 혹은 class명을 기준으로 table태그의 크롤링을 시도할 때 알지 못하는 이유로 계속 실패하여 None이 출력되는 상황이 발생하였다. 따라서 2번 사이트에 프리미어리그에 관한 표들을 통해서 데이터를 수집하기로 결정하였다. 가장 기본적인 데이터들이 있는 League Table을 비롯해 Squad Standard Stats, Squad Goalkeeping, Squad Advanced Goalkeeping, Squad Shooting, Squad Passing 등 많은 표를 통해서 EPL 20개 팀에 대한 다양한 세부 데이터들을 확인할 수 있다. 이중 승률데이터의 계산을 위해 등수, 경기 승패 및 기대득점에 관한 기본정보가 담긴 League Table과 점유에 관한 세부 데이터들이 기록된 Squad Possession를 통해 연구에 사용될 데이터를 모두 수집하였다. 사이트에서 확인할 수 있는 원본 데이터의 모습은 다음과 같다.

Rk	Squad	MP	W	D	L	GF	GA	GD	Pts	xG	xGA	xGD	xGD/90	Attendance	Top Team Scorer	Goalkeeper	Notes
1	Manchester City	38	27	5	6	83	32	+51	86	73.3	31.4	+42.0	+1.10	526	İlkay Gündoğan - 13	Ederson	→ UEFA Champions League via league finish
2	Manchester Utd	38	21	11	6	73	44	+29	74	60.2	42.2	+18.0	+0.47	526	Bruno Fernandes - 18	David de Gea	→ UEFA Champions League via league finish
3	Liverpool	38	20	9	9	68	42	+26	69	72.6	45.3	+27.3	+0.72	837	Mohamed Salah - 22	Alisson	→ UEFA Champions League via league finish
4	Chelsea	38	19	10	9	58	36	+22	67	64.0	32.8	+31.2	+0.82	526	Jorginho - 7	Edouard Mendy	→ UEFA Champions League via league finish
5	Leicester City	38	20	6	12	68	50	+18	66	56.0	47.7	+8.3	+0.22	421	Jamie Vardy - 15	Kasper Schmeichel	→ UEFA Europa League via cup win
6	West Ham	38	19	8	11	62	47	+15	65	53.9	48.3	+5.6	+0.15	632	Tomáš Souček, Michail Antonio - 10	Łukasz Fabiański	→ UEFA Europa League via league finish
7	Tottenham	38	18	8	12	68	45	+23	62	54.5	49.5	+5.0	+0.13	632	Harry Kane - 23	Hugo Lloris	→ UEFA Europa Conference League via league finish [1]
8	Arsenal	38	18	7	13	55	39	+16	61	53.5	44.3	+9.2	+0.24	632	Alexandre Lacazette - 13	Bernd Leno	
9	Leeds United	38	18	5	15	62	54	+8	59	57.5	62.9	-5.4	-0.14	421	Patrick Bamford - 17	Illan Meslier	
10	Everton	38	17	8	13	47	48	-1	59	47.2	51.2	-4.1	-0.11	368	Dominic Calvert-Lewin - 16	Jordan Pickford	
11	Aston Villa	38	16	7	15	55	46	+9	55	52.9	52.9	+0.1	0.00	526	Ollie Watkins - 14	Emiliano Martínez	
12	Newcastle Utd	38	12	9	17	46	62	-16	45	41.0	54.0	-13.0	-0.34	526	Callum Wilson - 12	Karl Darlow	
13	Wolves	38	12	9	17	36	52	-16	45	39.9	45.9	-6.0	-0.16	237	Rúben Neves, Pedro Neto - 5	Rui Patrício	
14	Crystal Palace	38	12	8	18	41	66	-25	44	32.4	57.5	-25.0	-0.66	447	Wilfried Zaha - 11	Vicente Guaita	
15	Southampton	38	12	7	19	47	68	-21	43	42.4	54.2	-11.8	-0.31	526	Danny Ings - 12	Alex McCarthy	
16	Brighton	38	9	14	15	40	46	-6	41	51.6	37.7	+13.9	+0.37	523	Neal Maupay - 8	Robert Sánchez	
17	Burnley	38	10	9	19	33	55	-22	39	39.9	57.6	-17.7	-0.47	178	Chris Wood - 12	Nick Pope	
▼ 18	Fulham	38	5	13	20	27	53	-26	28	40.5	53.0	-12.5	-0.33	211	Bobby Reid - 5	Alphonse Areola	Relegated
▼ 19	West Brom	38	5	11	22	35	76	-41	26	33.8	67.7	-34.0	-0.89	283	Matheus Pereira - 11	Sam Johnstone	Relegated
▼ 20	Sheffield Utd	38	7	2	29	20	63	-43	23	31.4	62.4	-31.0	-0.82	263	David McGoldrick - 8	Aaron Ramsdale	Relegated

▲ 팀명, 등수, 승점, 골득실 등 기본적인 데이터들이 포함된 League Table

				Touches							Dribbles					Carries								Receiving			
Squad	# Pl	Poss	90s	Touches	Def Pen	Def 3rd	Mid 3rd	Att 3rd	Att Pen	Live	Succ	Att	Succ%	#Pl	Megs	Carries	TotDist	PrgDist	Prog	1/3	CPA	Mis	Dis	Targ	Rec	Rec%	Prog
Arsenal	29	53.8	38.0	25217	2660	8496	11724	6662	979	23631	303	570	53.2	325	28	17270	91010	51942	2027	533	191	319	361	20487	17802	86.9	1350
Aston Villa	24	48.1	38.0	20985	2696	7056	8919	6192	1060	19152	394	608	64.8	415	19	12344	65238	35018	1414	429	205	443	418	15702	12779	81.4	1279
Brighton	27	51.3	38.0	23519	2703	8395	10204	6485	1020	21874	344	547	62.9	371	17	15362	84064	44546	1667	546	135	433	406	18583	15598	83.9	1265
Burnley	25	41.7	38.0	19236	2320	6089	9006	5107	748	17531	249	420	59.3	269	18	10026	49692	23754	948	342	81	406	335	13976	10644	76.2	975
Chelsea	27	61.4	38.0	29564	2703	9251	14400	7810	1131	27953	368	643	57.2	393	33	19875	99591	56260	2265	679	246	444	422	24517	21646	88.3	1555
Crystal Palace	24	40.1	38.0	19661	2331	7027	9078	4735	706	17884	373	637	58.6	401	21	11233	57886	29995	1211	394	134	425	513	14234	11390	80.0	984
Everton	29	46.5	38.0	22718	2922	8747	10503	4941	700	21133	341	586	58.2	363	30	13795	76593	41361	1388	411	114	430	449	17418	14670	84.2	1081
Fulham	28	49.9	38.0	23627	2640	8285	10981	5883	822	21923	497	761	65.3	523	34	14932	80620	43245	1613	509	133	587	472	18034	15183	84.2	1169
Leeds United	23	57.6	38.0	24596	3071	9481	10576	6094	985	22781	321	558	57.5	351	35	15012	85598	48248	1648	509	157	504	410	19143	15997	83.6	1440
Leicester City	27	54.6	38.0	25031	2426	8195	12125	6207	838	23368	350	629	55.6	379	29	15999	79829	42855	1679	505	135	467	429	19730	16726	84.8	1277
Liverpool	28	62.4	38.0	29822	2223	7936	14808	8910	1372	28088	421	703	59.9	457	28	20288	98263	53343	2281	695	230	482	449	24736	21527	87.0	1949
Manchester City	24	63.9	38.0	30525	2142	7108	16433	9082	1360	29002	483	805	60.0	516	33	22041	117302	67025	3045	809	314	378	418	26077	23578	90.4	1661
Manchester Utd	29	55.8	38.0	26392	2392	8046	12262	7891	1043	24709	424	733	57.8	460	35	17172	94013	53179	2196	676	180	422	468	21335	18495	86.7	1520
Newcastle Utd	27	38.2	38.0	18589	2672	7557	7582	4511	637	16830	352	577	61.0	379	20	10688	53725	27978	1015	355	116	391	388	13307	10617	79.8	779
Sheffield Utd	27	41.5	38.0	20176	2446	6665	8650	5985	703	18555	300	510	58.8	330	28	11340	59011	31817	1212	384	97	411	389	14805	12057	81.4	1024
Southampton	29	52.2	38.0	23420	2440	7993	11611	5273	793	21630	367	631	58.2	395	15	13917	71626	38066	1511	482	130	467	454	17833	14758	82.8	1112
Tottenham	24	51.7	38.0	24321	2708	8302	12150	5317	770	22574	408	673	60.6	435	34	15374	74651	39501	1550	468	135	369	467	19068	16142	84.7	1208
West Brom	30	37.6	38.0	18234	2692	6755	7837	4511	653	16495	294	534	55.1	319	21	9646	49732	25174	955	283	108	459	427	12638	9743	77.1	912
West Ham	24	42.9	38.0	20537	2423	6846	9450	5500	828	18875	325	546	59.5	348	25	12116	68476	34759	1376	451	143	383	380	15380	12533	81.5	1014
Wolves	27	49.3	38.0	22963	2287	7494	11331	5635	759	21244	496	779	63.7	537	29	14457	76002	42595	1794	544	183	453	394	17784	15234	85.7	1077

▲ 점유율과 세부관련 데이터들이 포함된 Squad Possession

언급된 2개의 표의 table태그의 id값을 기준으로 thaed와 tbody의 하위태그 값들을 추출하여 각각 main_df과 pos_df이라는 이름의 데이터프레임을 생성하였다. 2개의 데이터프레임에서 겹치는 열은 Squad로 1등부터 20등까지 각 팀명을 확인할 수 있는 칼럼이다. 이 공통된 Squad 칼럼을 기준으로 inner join을 실행하여 두 테이블을 total_df라는 하나의 데이터프레임으로 병합하였고, total_df를 활용하여 모델을 설정한 뒤 분석을 진행할 것이다. total_df의 모습은 다음과 같다.

```
$ python crawling.py
    Rk           Squad  MP   W   D   L  GF  GA   GD Pts    xG   xGA    xGD xGD/90  ... #Pl Megs Carries TotDist PrgDist  Prog 1/3  CPA Mis Dis  Targ   Rec Rec%  Prog
0    1  Manchester City  38  27   5   6  83  32  +51  86  73.3  31.4  +42.0  +1.10  ... 516   33   22041  117302   67025  3045 809  314 378 418 26077 23578 90.4  1661
1    2  Manchester Utd  38  21  11   6  73  44  +29  74  60.2  42.2  +18.0  +0.47  ... 460   35   17172   94013   53179  2196 676  180 422 468 21335 18495 86.7  1520
2    3        Liverpool  38  20   9   9  68  42  +26  69  72.6  45.3  +27.3  +0.72  ... 457   28   20288   98263   53343  2281 695  230 482 449 24736 21527 87.0  1949
3    4          Chelsea  38  19  10   9  58  36  +22  67  64.0  32.8  +31.2  +0.82  ... 393   33   19875   99591   56260  2265 679  246 444 422 24517 21646 88.3  1555
4    5   Leicester City  38  20   6  12  68  50  +18  66  56.0  47.7   +8.3  +0.22  ... 379   29   15999   79829   42855  1679 505  135 467 429 19730 16726 84.8  1277
5    6         West Ham  38  19   8  11  62  47  +15  65  53.9  48.3   +5.6  +0.15  ... 348   25   12116   68476   34759  1376 451  143 383 380 15380 12533 81.5  1014
6    7        Tottenham  38  18   8  12  68  45  +23  62  54.5  49.5   +5.0  +0.13  ... 435   34   15374   74651   39501  1550 468  135 369 467 19068 16142 84.7  1208
7    8          Arsenal  38  18   7  13  55  39  +16  61  53.5  44.3   +9.2  +0.24  ... 325   28   17270   91010   51942  2027 533  191 319 361 20487 17802 86.9  1350
8    9     Leeds United  38  18   5  15  62  54   +8  59  57.5  62.9   -5.4  -0.14  ... 351   35   15012   85598   48248  1648 509  157 504 410 19143 15997 83.6  1440
9   10          Everton  38  17   8  13  47  48   -1  59  47.2  51.2   -4.1  -0.11  ... 363   30   13795   76593   41361  1388 411  114 430 449 17418 14670 84.2  1081
10  11     Aston Villa  38  16   7  15  55  46   +9  55  52.9  52.9   +0.1   0.00  ... 415   19   12344   65238   35018  1414 429  205 443 418 15702 12779 81.4  1279
11  12    Newcastle Utd  38  12   9  17  46  62  -16  45  41.0  54.0  -13.0  -0.34  ... 379   20   10688   53725   27978  1015 355  116 391 388 13307 10617 79.8   779
12  13         Wolves  38  12   9  17  36  52  -16  45  39.9  45.9   -6.0  -0.16  ... 537   29   14457   76002   42595  1794 544  183 453 394 17784 15234 85.7  1077
13  14  Crystal Palace  38  12   8  18  41  66  -25  44  32.4  57.5  -25.0  -0.66  ... 401   21   11233   57886   29995  1211 394  134 425 513 14234 11390 80.0   984
14  15     Southampton  38  12   7  19  47  68  -21  43  42.4  54.2  -11.8  -0.31  ... 395   15   13917   71626   38066  1511 482  130 467 454 17833 14758 82.8  1112
15  16       Brighton  38   9  14  15  40  46   -6  41  51.6  37.7  +13.9  +0.37  ... 371   17   15362   84064   44546  1667 546  135 433 406 18583 15598 83.9  1265
16  17         Burnley  38  10   9  19  33  55  -22  39  39.9  57.6  -17.7  -0.47  ... 269   18   10026   49692   23754   948 342   81 406 335 13976 10644 76.2   975
17  18          Fulham  38   5  13  20  27  53  -26  28  40.5  53.0  -12.5  -0.33  ... 523   34   14932   80620   43245  1613 509  133 587 472 18034 15183 84.2  1169
18  19      West Brom  38   5  11  22  35  76  -41  26  33.8  67.7  -34.0  -0.89  ... 319   21    9646   49732   25174   955 283  108 459 427 12638  9743 77.1   912
19  20    Sheffield Utd  38   7   2  29  20  63  -43  23  31.4  62.4  -31.0  -0.82  ... 330   28   11340   59011   31817  1212 384   97 411 389 14805 12057 81.4  1024

[20 rows x 45 columns]
```

▲ Squad 칼럼을 기준으로 main_df와 pos_df를 inner join한 total_df

total_df에 모델의 종속변수에 해당하는 승률인 Win Rate 칼럼을 추가하기 위한 작업을 진행하였다. 먼저 데이터 타입을 판다스의 to_numeric 메소드를 활용하여 전부 숫자형으로 변경하였다. 현재 문자형 데이터인 팀이름, 선수명, 메모 등은 분석에 있어서 필요하지 않을 것으로 판단하여 NaN(Not a Number)로 결측치 처리하였다. 이후 승리경기 수인 W 칼럼의 데이터를 전체 경기수인 38로 나눈 후 100을 곱하고 이를 소수점 한 자리까지 표기한 값을 차례대로 Win Rate 칼럼에 추가하였다.

4. 모형 설정

연구 목적에서 언급되었듯이 필자는 이번 연구를 통해 크게 두가지 상관관계를 파악하고자 한다. 첫번째는 점유율과 승률의 상관관계이며, 두번째는 기대득점값과 승률의 상관관계이다. 전자는 점유율에 해당하는 칼럼인 Poss칼럼과 승률 칼럼인 Win Rate 칼럼 간의 회귀분석, 후자는 기대득점값에 해당하는 xG 칼럼과 승률 칼럼인 Win Rate 칼럼 간의 회귀분석을 통해 파악할 수 있다. 그리고 다음과 같은 가설을 설정하게 되었다.

가설1. 점유율은 경기승률에 정(+)의 영향을 미칠 것이다.

가설2. 기대득점값은 경기승률에 정(+)의 영향을 미칠 것이다.

하지만 필자가 구축한 total_df 데이터프레임에는 Poss 칼럼과 xG 칼럼 이외에도 이 2개의 데이터에 관한 더욱 세부적인 지표들이 많이 들어있다. 그렇기 때문에 다양한 요소들이 승률에 미치는 영향을 파악하고자 단순회귀분석을 진행하지 않고 다중회귀모형을 설정하여 구체적으로 요소 간의 영향력을 파악하고 결론을 도출하고자 한다.

본격적인 분석을 시행하기에 앞서 total_df의 각 요소 간의 상관관계를 파악하였다. 이 데이터프레임은 총 45개의 칼럼을 보유하였고 모든 칼럼들을 다 활용하여 분석을 진행하는 것은 의미가 없을 것으로 판단하여, 종속변수인 Win Rate와 강한 연관성을 갖는 지표만을 해당 연구의 독립변수로 사용하려고 한다. 여러 상관계수를 측정할 수 있는 지표들 중 가장 일반적인 피어슨 상관계수 값을 추출하였고, 출력된 상관계수의 절댓값이 0.7 이상인 경우에만 강한 상관관계를 가지고 있다고 판단하였다. 이 작업을 통해 다음과 같은 출력값을 확인하였으며, 승률과 강한 연관성을 띠고 있는 21개의 요소를 추출하였다. 이 중에서 Win Rate 칼럼 계산에 활용된 승, 패, 승점, 그리고 등수와 자기 자신인 승률 칼럼은 제외하여 총 16개의 칼럼을 최종 독립변수로 선택하였다.

	Win Rate
Rk	-0.961770
W	0.999998
L	-0.884182
GF	0.941855
GA	-0.748657
GD	0.947014
Pts	0.988002
xG	0.860903
xGD	0.814275
xGD/90	0.813454
Touches	0.705578
Mid 3rd	0.709707
Live	0.705946
Carries	0.710028
TotDist	0.711492
PrgDist	0.708432
Prog	0.724647
CPA	0.710133
Targ	0.721438
Rec	0.724568
Win Rate	1.000000

▲ Win Rate와 강한 상관관계를 가지고 있는 21개의 요소 (피어슨 상관계수 >= 0/7)

5. 모형 분석

16개의 최종 독립변수를 골 관련 지표 6개와 점유 관련 지표 10개로 나누어서 2개의 다중회귀모형을 만들어 최종 분석을 진행하고자 한다. 본격적으로 다중회귀분석을 실시하기에 앞서 다중회귀분석의 전제조건에 대해 파악할 필요가 있다. 모델을 설정하고 요약통계량과 VIF계산의 과정을 통해 필자가 설정한 모델이 적합한 모델인지 판단하는 과정을 거칠 예정이다. 다중회귀분석의 4가지 조건은 다음과 같다.

1) 정규분포성

2) 등분산성

3) 선형성

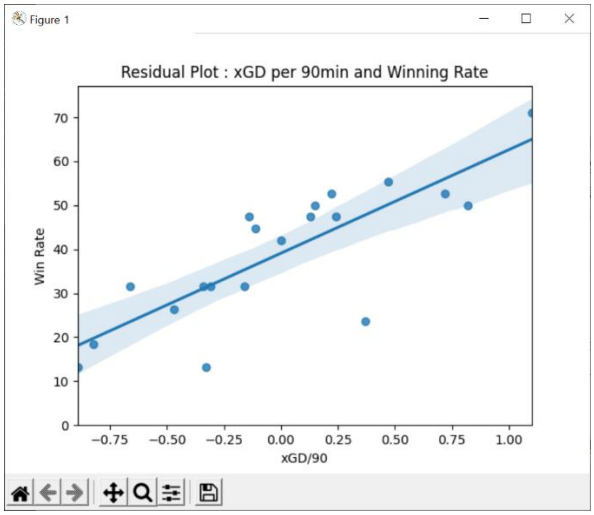
4) 독립변수들 간의 다중공선성이 없어야함

<모델1>

먼저 전체적인 상관관계가 더 강한 골 관련 지표들을 활용하여 다중회귀모형을 설정하였다. GF, GA, GD, xG, xGD, xGD/90을 독립변수, Win Rate을 종속변수로 설정하여 모델을 만들고 해당 모델의 요약통계량의 p-value와 VIF값을 확인하였다. p-value를 통해 유의수준 5%에서 GA와 xG가 통계적으로 유의미하지 않음을 확인하였다. VIF는 다중회귀모델에서 독립변수 간 상관관계가 있는지 측정하는 척도로 일반적으로 VIF가 10이 넘으면 다중공선성이 있다고 판단하고, 5가 넘으면 주의할 필요가 있는 것으로 판단한다. VIF계산을 통해 확인한 결과, GF, GA, GD의 VIF값은 무한대로 발산하는 inf에 해당하여 언급된 3개의 칼럼은 사용할 수 없을 것으로 판단했다. 따라서 기존에 설정된 6개의 골 관련 지표들 중 xGD와 xGD/90만을 활용하여 최종 분석을 진행할 수 있을 것이다. xGD는 expected goals difference의 약자로 expected goals인 xG에서 expected goals allowed인 xGA를 뺀 값이다. 즉 기대득점과 기대실점의 차이라고 볼 수 있다. xGD/90은 xGD를 축구 정규시간인 90분에 맞춰서 계산한 것이다. 즉 xGD와 xGD/90은 사실상 같은 의미를 가지고 있는 지표라고 볼 수 있다. 따라서 이 2개의 지표 중 VIF값이 더 작게 나온 xGD/90만을 독립변수로 설정하여 Win Rate와의 관계를 파악하는 단순회귀분석을 시행하였다. 그 결과 다음과 같은 y절편으로 39.1078, 기울기로 23.5586, 결정계수로 0.6617에 해당하는 값을 얻을 수 있었다. 최종 회귀방정식은 다음과 같으며 해당 회귀선을 통해 66.17%를 설명한다.

Win Rate = 39.1077 + 23.5586 * xGD/90

Regression Plot은 다음의 그래프와 같다. 대체적으로 선형의 모양을 잘 나타내고 있으며 xGD/90값이 높을수록 승률이 올라가는 점을 확인할 수 있다.



▲ 모델1의 Regression Plot

```
                            OLS Regression Results
==============================================================================
Dep. Variable:               Win Rate   R-squared (uncentered):                   0.992
Model:                            OLS   Adj. R-squared (uncentered):              0.990
Method:                 Least Squares   F-statistic:                              381.0
Date:                Wed, 23 Jun 2021   Prob (F-statistic):                    3.02e-15
Time:                        01:18:12   Log-Likelihood:                         -54.558
No. Observations:                  20   AIC:                                      119.1
Df Residuals:                      15   BIC:                                      124.1
Df Model:                           5
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
GF             0.4610      0.098      4.727      0.000       0.253       0.669
GA             0.1566      0.121      1.297      0.214      -0.101       0.414
GD             0.3044      0.071      4.311      0.001       0.154       0.455
xG             0.1368      0.214      0.638      0.533      -0.320       0.594
xGD           25.4151      9.295      2.734      0.015       5.604      45.226
xGD/90      -967.7156    351.437     -2.754      0.015   -1716.785    -218.646
==============================================================================
Omnibus:                        0.457   Durbin-Watson:                   1.913
Prob(Omnibus):                  0.796   Jarque-Bera (JB):                0.570
Skew:                          -0.258   Prob(JB):                        0.752
Kurtosis:                       2.353   Cond. No.                     3.25e+16
==============================================================================
```

▲ 모델1의 요약통계량

	factor	VIF
0	GF	inf
1	GA	inf
2	GD	inf
3	xG	1.324378e+02
4	xGD	3.692552e+04
5	xGD/90	3.651349e+04

▲ 모델1의 다중공선성 확인

<모델2>

두번째로 점유 관련 지표들을 독립변수로 설정하여 다중회귀분석을 진행하였다. 승률과 어느정도 상관관계를 가지고 있는 것으로 예상되는 점유 관련 칼럼이 총 10개가 추출되었다. 하지만 10가지를 모두 사용하기에는 변수수가 너무 많아 유의수준 및 다중공선성 등의 문제가 발생하여 원활하게 분석이 이루어지지 않을 것이라고 판단하였다. 따라서 10개의 지표 중 피어슨 상관계수 값이 거의 제일 높으면서도 필자의 개인적인 판단으로 실제 볼 소유와 공격 전개에 유의미한 영향을 줄 수 있을 것으로 예상되는 2개의 독립변수를 결정하여 모델을 설정하였다. 각 독립변수에 대한 설명은 다음과 같다.

- Targ : Numbers of times a player was the target of an attempted pass

- Prog : Completed passes that move the ball towards the opponent's goal at least 10 yards from its furthest point in the last six passes, or any completed pass into the penalty area. Excludes passes from the defending 40% of the pitch

이후 다중회귀모형의 적합성을 파악하고자 해당 모델의 요약통계량의 p-value와 VIF값을 확인하였다.

```
                            OLS Regression Results
==============================================================================
Dep. Variable:               Win Rate   R-squared (uncentered):                   0.939
Model:                            OLS   Adj. R-squared (uncentered):              0.932
Method:                 Least Squares   F-statistic:                              138.1
Date:                Wed, 23 Jun 2021   Prob (F-statistic):                    1.20e-11
Time:                        10:59:50   Log-Likelihood:                         -75.138
No. Observations:                  20   AIC:                                      154.3
Df Residuals:                      18   BIC:                                      156.3
Df Model:                           2
Covariance Type:            nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Targ          -0.0019      0.003     -0.579      0.570      -0.009       0.005
Rec            0.0048      0.004      1.239      0.231      -0.003       0.013
==============================================================================
Omnibus:                        1.920   Durbin-Watson:                   1.212
Prob(Omnibus):                  0.383   Jarque-Bera (JB):                1.129
Skew:                          -0.582   Prob(JB):                        0.569
Kurtosis:                       2.963   Cond. No.                         51.0
==============================================================================
```

▲ 모델2의 요약통계량

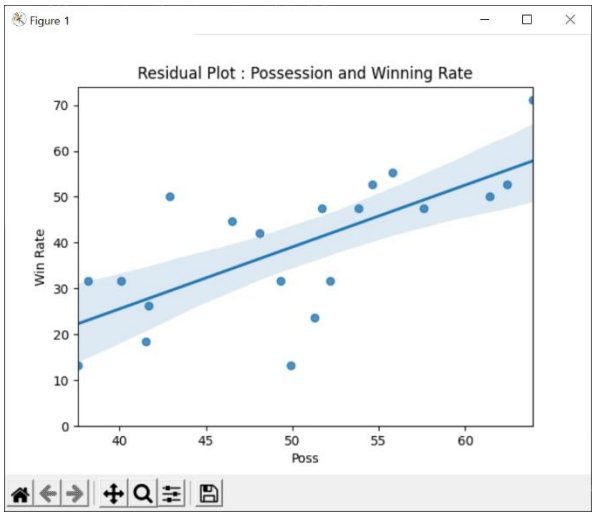
	factor	VIF
0	Targ	632.961868
1	Rec	632.961868

▲ 모델2의 다중공선성 확인

제시된 표들을 통해 p-value와 VIF값을 출력해보니 Targ와 Rec변수 모두 p-value가 0.05보다 훨씬 크기 때문에 95%의 신뢰수준에서 유의하지 않으며, VIF값도 매우 크게 측정되었기 때문에 두 독립변수 간의 상관성도 상당하여 다중회귀분석을 실시할 수 없을 것이라는 결론을 내리게 되었다. 하지만 앞서 언급된 것처럼 본 연구는 점유율과 승률의 관계를 파악하는 것이 목적이었기 때문에 점유 관련 세부지표를 활용하는 것 대신 점유율인 Poss 칼럼을 독립변수로, Win Rate 칼럼을 종속변수로 하는 단순회귀모형을 설정하여 추가적인 분석을 진행하기로 결정했다. 그 결과 다음과 같은 y절편으로 -28.4205, 기울기로 1.3495, 결정계수로 0.4872에 해당하는 값을 얻을 수 있었다. 최종 회귀방정식은 다음과 같으며 해당 회귀선을 통해서 48.72%를 설명할 수 있다. 모델1보다 확실히 기울기로 낮고 설명력도 떨어지는 점을 확인할 수 있다.

Win Rate = -28.4205 + 1.3495 * Poss

Regression Plot은 다음의 그래프와 같다. 대체적으로 선형의 모양을 잘 나타내고 있으며, Poss값이 높을수록 승률이 올라가는 점을 확인할 수 있다. 하지만 모델1의 Regression Plot에 비해 회귀선으로부터 멀리 떨어져 있는 값들이 훨씬 많은 것으로 보아 설정된 두 변수 간 상관관계가 엄청나게 높다고는 볼 수 없을 것이다.



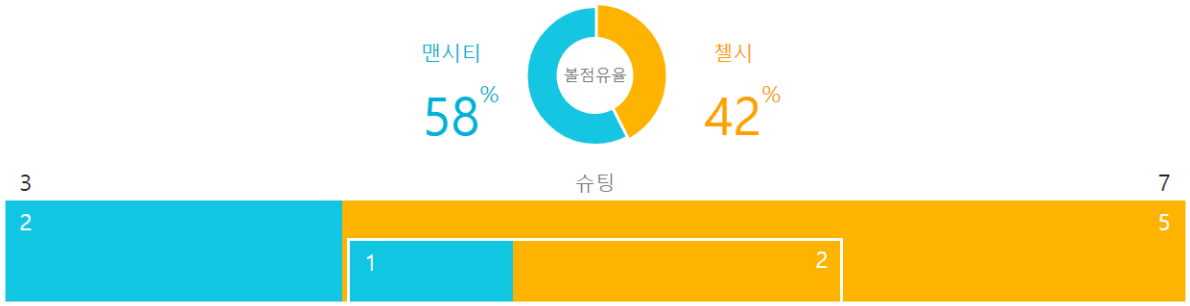
▲ 모델2의 Regression Plot

6. 결론

이번 연구를 통해 축구 경기의 승률에 미치는 다양한 스탯들을 확인해보고, 특히 점유율과 기대득점값을 중심으로 승률과의 연관성을 파악해보고자 하였다. 필자가 내린 결론은 다음과 같다.

점유과 빌드업에 관해 측정할 수 있는 다양한 지표들이 존재하지만, 결국 축구는 상대보다 많은 골을 넣어야 승리할 수 있는 스포츠이기 때문에 기대득점값을 높이고, 기대실점값을 줄여야 경기를 승리로 이끌 수 있다.

점유는 큰 틀에서 proactive한 경기를 펼칠 것인지, reactive한 경기를 할 것인지에 대한 방향성을 결정하는 매우 중요한 요소 중 하나이다. 공 점유시간이 많다는 것은 그만큼 경기를 공격적이고, 주도적으로 진행하겠다는 의미이다. 하지만 최근에는 선수비 후역습의 컨셉으로 상대에게 점유를 내주더라도 빠른 공수전환을 통해 상대팀의 수비 뒷공간의 허점을 노려 득점을 올리고 승리를 차지하는 경우가 정말 많아지고 있다. 가장 최근에 펼쳐진 챔피언스리그 결승전에서도 이러한 경향을 볼 수 있었다.



▲ 20-21 Uefa 챔피언스리그 결승전 경기스탯 (맨시티 vs 첼시)

첼시가 점유율에서는 밀리는 모습을 보였으나 효과적인 공격시도로 더 많은 슈팅과 유효슈팅을 기록했으며 결국 한 골을 넣어 1:0으로 승리했다. 결국 점유보다는 득점에 관한 지표들을 개선하는 것이 경기를 승리로 이끌어갈 수 있는 좋은 전략이다. 한정된 공격기회 속에서도 찬스를 만들어 최종 슈팅 및 유효슈팅까지 이어질 수 있도록 하는 것이 중요하다. 물론 xG 혹은 xGD값이 높게 측정되었다고 해서 실제로 같은 숫자만큼의 득점이 기록되는 것은 아니다. 하지만 골대거리, 슈팅의 각도, 슈팅 유형(오른발/왼발/머리), 골키퍼와의 1:1여부, 공격상황의 유형, 세컨볼의 여부 등 다양한 상황을 고려하여 측정된 과학적인 지표가 바로 xG값인 만큼 이를 개선하기 위한 노력은 경기 승리에 있어서 매우 큰 영향을 줄 수 있을 것이라고 생각한다.

7. 부록

```
import pandas as pd
from bs4 import BeautifulSoup
from pandas.core.reshape.merge import merge
import requests
from sklearn.linear_model import LinearRegression
import seaborn as sns
import matplotlib as plt
import statsmodels.api as sm
from statsmodels.stats.outliers_influence import variance_inflation_factor

# 웹크롤링 시작
url = "https://fbref.com/en/comps/9/Premier-League-Stats"
page = requests.get(url = url)
soup = BeautifulSoup(page.text, 'html.parser')

# main table 생성 -> 순위에 관한 기본 정보
main_table = soup.find("table", {"id" : "results107281_overall"})

for i in main_table.find_all('thead') :
    title = i.find_all('th')
    row_title = [th.text for th in title]

main_df = pd.DataFrame(columns = row_title)

for j in main_table.find_all('tr')[1:] :
    rank = j.find_all('th')
    data = j.find_all('td')
    row_rank = [th.text for th in rank]
    row_data = [tr.text for tr in data]
    row = row_rank + row_data
    new_row = []
    for a in row :
        a = a.strip()
        new_row.append(a)
    length = len(main_df)
    main_df.loc[length] = new_row

# print(main_df)
# main_df.to_csv("main_df.csv")

# pos table 생성 -> 점유율에 관한 세부적인 수치
pos_table = soup.find("table", {"id" : "stats_squads_possession_for"})

for i in pos_table.find_all('thead') :
    title = i.find_all('th')
```

```
    row_title = [th.text for th in title]

row_title = row_title[6:]

pos_df = pd.DataFrame(columns = row_title)

for j in pos_table.find_all('tr')[2:] :
    rank = j.find_all('th')
    data = j.find_all('td')
    row_rank = [th.text for th in rank]
    row_data = [tr.text for tr in data]
    row = row_rank + row_data
    length = len(pos_df)
    pos_df.loc[length] = row

# print(pos_df)
# pos_df.to_csv("pos_df.csv")

# 2 개의 데이터프레임 합치기 / 기준열 = Squad(팀명)
total_df = pd.merge(main_df, pos_df, left_on="Squad", right_on="Squad", how="i
nner")

# print(total_df)
# total_df.to_csv("total_df.csv")

# 승률 열을 추가 (백분율 단위) -> 모든 모형의 종속변수로 활용
# 데이터 타입을 전부 숫자형으로 변경, 문자형 데이터는 팀이름, 선수명, 메모 등으로 분
석에 있어서 필요하지 않을 것으로 판단하여 결측치 처리
total_df = total_df.apply(pd.to_numeric, errors="coerce").fillna("NaN")
wins = total_df["W"].tolist()

win_rate = []
for i in range(len(wins)) :
    x = round(wins[i] / 38 * 100, 1)
    win_rate.append(x)
total_df['Win Rate'] = win_rate

# print(total_df)

# 각 요소 간 상관관계 파악 -> 피어슨 상관계수 값으로 비교
corr = total_df.corr(method = "pearson")[["Win Rate"]]

# 절대값이 0.7 이상인 요소만 강한 상관관계를 가지고 있다고 판단
strong_corr = abs(corr["Win Rate"]) >= 0.7
subset_df = corr[strong_corr]

# # 승률과 강한 상관관계를 가지고 있는 21 개의 요소 추출 / 이 중 당연한 요소인 등수,
 승, 패, 승점과 자기 자신인 승률은 제외 후 16 개의 요소를 가지고 단순회귀분석을 진행
```

```
# # 16 개의 요소를 1) 골 관련 지표, 2) 점유 관련 지표으로 나누어서 다중회귀분석 진행

# 첫번째 모델 - 골 관련 지표
Z = total_df[['GF','GA','GD','xG','xGD','xGD/90']]
Y = total_df['Win Rate']
model1 = sm.OLS(Y, Z)
res1 = model1.fit()
# print(res1.summary())

# 유의수준 5%에서 GA, xG 가 통계적으로 유의미하지 않음을 확인
# 다중공선성을 확인하기 위해 VIF 를 계산
vif1 = pd.DataFrame({'factor': column, 'VIF': variance_inflation_factor(model1
.exog, i)}
             for i, column in enumerate(model1.exog_names)
             if column != 'Intercept')

# print(vif1)

# 다중공선성을 위해 VIF 를 계산하니 GF,GA,GD 는 무한대값이 나오며, xG, xGD, xGD/90
은 VIF 가 낮게 측정되었으나, xG 는 p-value 에 의해 유의하지 않은 것으로 판단
# 따라서 통계적으로 유의미하며 다중공선성이 존재하지 않는 xGD 와 xGD/90 를 활용하여
모델을 다시 설정
# xGD/90 은 90 분당 xGD 를 의미하므로 사실상 같은 변수라고 판단하여 xGD/90 만을 독립
변수로 설정하여 단순회귀분석을 진행
lm = LinearRegression()
X = total_df[["xGD/90"]]
Y = total_df["Win Rate"]
lm.fit(X, Y)
Yhat = lm.predict(X)

a = (lm.intercept_) # y 절편
b = (lm.coef_) # 기울기
r_squared = (lm.score(X,Y)) # 결정계수
# print(a, b, r_squared)

# 첫번째 모델 regression plot 그리기
sns.regplot(X, Y)
plt.pyplot.ylim(0,)
plt.pyplot.title("Residual Plot : xGD per 90min and Winning Rate")
plt.pyplot.show()

# 두번째 모델 - 점유 관련 10 개의 지표 중 상관계수 값이 제일 높은 2 가지만 선택하여
진행
# Z = total_df[['Touches','Mid 3rd','Live','Carries','TotDist','PrgDist','Prog
','CPA','Targ','Rec']]
Z = total_df[['Targ', 'Rec']]
Y = total_df['Win Rate']
model2 = sm.OLS(Y, Z)
```

```
res2 = model2.fit()
# print(res2.summary())

# 유의수준 5%에서 모든 변수가 통계적으로 유의미하지 않음을 확인
# 다중공선성을 확인하기 위해 VIF 를 계산
vif2 = pd.DataFrame({'factor': column, 'VIF': variance_inflation_factor(model2
.exog, i)}
             for i, column in enumerate(model2.exog_names)
             if column != 'Intercept')

# print(vif2)

# 다중공선성을 위해 VIF 를 계산하니 값이 너무 높게 측정되어서 해당 모델을 사용하지
않기로 결정

# 초기 가설에 대한 추가 모델을 설정
# 점유율 - 승률 간의 단순회귀분석 실행
lm = LinearRegression()
X = total_df[["Poss"]]
Y = total_df["Win Rate"]

lm.fit(X, Y)
Yhat = lm.predict(X)

a = (lm.intercept_) # y 절편
b = (lm.coef_) # 기울기
r_squared = (lm.score(X,Y)) # 결정계수
# print(a, b, r_squared)

sns.regplot(X, Y)
plt.pyplot.ylim(0,)
plt.pyplot.title("Residual Plot : Possession and Winning Rate")
plt.pyplot.show()
```