야구 타선 모델링

- 효율적 야구 타선 짜기 -

고려대학교 공과대학 산업경영공학과
경영공학개론 29조
2010170813 강정인
2015120239 유아영
2015170378 정은영
2016170826 임대현

목 차

Ⅰ. 요약

본 29조는 야구 선수들의 능력을 수치화한 데이터를 통해 가장 효율적이며 승률이 높은 모델을 구성하는 것을 목적으로 삼았다. 모델링은 4명의 조원 모두가 참여하였으며 Java, Gurobi 프로그램을 이용하였다. 최적화에 사용된 모델은 경영공학개론 수업시간에 배운 모델들에 기반하여 작성되었다. 데이터는 두산선수의 전체기록을 사용하였으며, 714개의 data를 수집하였다. 실험의 영가설 설정에 앞서, 과정 전반에 적용된 가설은 다음과 같음을 밝힌다. 프로야구에서의 임의의 두 팀을 설정하고, 상대팀의 주전선수를 설정하여 상대팀에 대한 우리팀의 능력데이터에 적절한 비중을 두어 그 능력치를 최대화하는 모델링을 하였다. 단지 임의의 팀에 대한 승률을 따지는 것이 아닌 정해진 상대팀에 대한 모델링을 구성하여 상대팀의 데이터에 따라 타선이 달라질 수 있는 좀 더 구체적인 타선구성을 제시한다. 이 과정에서 수집된 데이터들의 비중을 통계적으로 설정함으로써 모델의 타당성을 더했다. 마지막으로 모델링의 결과와 보충 점을 기술함으로써 후속 연구에 도움이 될 수 있도록 하였다. 실험 전체의 Raw data는 보고서의 가장 뒷장에 첨부하였다.

Ⅱ. 실험의 목적 및 가정

2-1. 모델링 수행 단계

1) 문제 및 가설

본 실험은 프로야구주전선수의 능력들을 수치화시키고 최적 모델을 구성함으로써, 상대팀에 따른 최적의 타선을 짜는 것을 목적으로 한다. 우리팀 타자에게는 도루(wSB[1)]), 타율(BABIP[2)]), 장타율(OPS[3)]), 순수출루율(IsoD[4)]), 수비관련득점기여([5)]RNG)(제약), 다른 네 개의 데이터가 부여되며, 투수에게는 수비무관평균자책점([6)]FIP), 이닝(IP)(제약), 다른 두 개의 데이터가 주어진다. 선수의 능력들에 승리에 영향을 미치는 정도로 parameter를 부여함으로써 그 합을 최대화해서 최적의 타선을 구성하기로 한다.

프로야구를 기반으로 우리 팀은 두산베어스, 상대팀은 삼성라이온즈로 놓았다. 삼성라이온즈의 작년 경기 기반으로 삼성라이온즈의 주전선수를 구성하였다.

---

1) wSB(Weighted Stolen Base Runs)=SB*runSB+CS*runCS-리그평균값
※SB=도루, CS=도루실패
※runSB=도루로 인해 변화하는 평균 기대 득점 값(~0.2)
※runCS=도루 실패로 인해 변화하는 평균 기대 독점 값(~0.38~-0.45)

2) BABIP(Batting Average on Balls In Play)은(는) 인플레이(홈런이 아닌 페어플레이)로 된 경우에 대한 타자의 타율이나,투수의 피안타율을 계산하는 야구 기록 중 하나이다.
계산방법 BABIP=(안타-홈런)/(타수-삼진-홈런+희생플라이)

3) OPS=출루율(OBP)+장타율(SLG)

4) IsoD=isolated discipline
※순수출루율=출루율-타율

5) RNG-수비범위관련득점기여, 실책과 관련이 있다.

6) FIP=(HR*13+(BB+HBP-IBB)*3-K*2)/IP+3.20 / 투수의 진가를 알 수 있는 지표 중 하나로 운적인 요소를 제외한 스탯이다.
※HR=홈런
※BB=4구
※HBP=몸에 맞는 볼
※IBB=고의4구
※K=삼진
※IP=이닝

삼성라이온즈(상대팀)
투수 | 차우찬, 김기태, 윤성환
타자 | 박해민 박한이 구자욱 최형우 이승엽 백상원 조동찬 이지영 김상수

두산베어스의 선수는 데이터가 너무 없는 선수를 제외하고 모두 타선목록에 넣었다.
두산베어스(우리 팀)

투수	유희관, 이현승, 김성배, 김승희, 김강률, 고봉재, 함덕주, 이현호, 장원준, 니퍼트
포수	양의지, 박세혁
내야수	허경민, 최주한, 오재원, 김재호, 오재일, 류지혁, 에반스, 신성현
외야수	국해성, 민병헌, 김재환, 박건우

타선은 1번부터 9번까지로 구성되며 각각 중요한 역할이 다르다.

1번 타자에게 가장 중요시 여겨지는 능력은 출루율(IsoD)과 도루율(wSB)이다.
1번 타자의 출루율이 중요한 통계적 이유가 있다. 보통 이닝 선두타자로 나오는 빈도가 더 높기 때문에 똑같은 횟수의 출루기회가 있을 때 이닝 선두타자가 출루하는 것이 더 많은 득점을 얻을 기회가 되기 때문이다. 또한 도루율도 중요한데, 도루를 성공했을 경우 다음 타자가 2루타 이상을 친다면 바로 득점으로 이어지기 때문이다.

2번 타자에게 가장 중요시 여겨지는 능력은 출루율(IsoD)와 장타율(Slugging Average)이다.
출루를 잘하는 선수가 리드오프에 있고 장타력이 좋은 선수가 3, 4, 5번에 있다면 2번 선수는 아웃당하지 않는 능력, 즉 출루율과 멀리치는 능력, 즉 장타율이 둘 다 중요하다. 상대적으로 2번 선수의 출루율(IsoD)이 4번 선수의 출루율 보다 높아야 되며, 4번 선수의 장타율(OPS)이 2번 선수의 장타율 보다 높아야 한다.

3번 타자에게 가장 중요시 여겨지는 능력은 타율(BABIP)과 장타율(OPS)이다.
상대적으로 3번 타자에는 2루타를 많이 치는 타자를 배치하는 것이 유리하다.

4번 타자에게 가장 중요시 여겨지는 능력은 장타율(OPS)이다.
장타율이 가장 중요하다고 보면 된다. 어느 타순이든 장타력은 높을수록 좋지만 같은 장타력을 가진 선수일 경우 4번에 있을 때 효율성이 가장 좋다. 왜냐하면 필드 각 루에 자리잡고 있는 선수들이 가장 많은 시기이기 때문이다.

5번 타자에게 가장 중요시 여겨지는 능력은 타율(BABIP)이다.
팀재 상위 다섯 번 째에 위치하는 타자가 주로 출전한다.

6번 타자에게 가장 중요시 여겨지는 능력은 도루율(wSB)이다.
도루능력을 가장 효과적으로 발휘할 수 있는 슬롯이 6번 슬롯이다. 도루의 가치는 이어지는 타자들의 장타율이 높을수록 낮아지고 낮을수록 높아진다. 1번과 2번 타자의 뒤에는 장타율이 높은 타자가 배치되었고, 6번 타자 뒤에는 그렇지 않기 때문에 도루 능력은 1번보다 6번

타자에게서 더 많이 요구된다.

7번, 8번, 그리고 9번 타자는 그다지 큰 의미 부여를 하지 않는다.
마지막 슬롯에 서는 타자들은 능력치가 거의 비슷하며, 특징이 희미하다.

투수(FIP)에게 요구되는 가장 중요한 능력치는 바로 방어율이다. 방어율 수치를 산정할 때, 팀의 수비력까지 포함된 방어율이 아닌 투수 개인의 능력만 고려한 FIP의 데이터가 사용된다.

이러한 차이를 기반으로 본 29조는 각 능력치 별로 다른 가중치를 가정하였다. 하지만 가중치의 경우, 임의의 parameter를 설정하면 중요성을 왜곡하는 효과가 나타날 수 있다. 예를 들어, 어떤 타자에게 중요시 여겨지는 능력이 A능력보다 B능력이라고 가정하자. 여기서 만약 A능력이 1미만의 값을 가지고, B능력이 1이상의 값을 가진다면, 같은 비중을 줄 경우에 그 중요도를 반영할 수 없게 된다. 이러한 이유로 승리한 경기의 타선에서의 선수들의 데이터를 통계적으로 조사 ,정리해 반영하기로 결정하였다.

2) 목적식

Max (z=$\sum_{i=1}^{24}\sum_{k=1}^{11}\sum_{j=1}^{5} a_{jk}c_{ij}x_{ik}$)

aik: k번째 타선에 j능력을 가지고 있는 선수가 위치할 때, 그 능력치 (j=1, 2, 3, 4능력은 타자의 능력 / j =5능력은 투수의 능력)
cij:구단의 선수를 1-24명이라고 가정할 때 i번째 선수의 j번째 능력 (i=1,2, ...24)(j=1,2, ...6)
xik: i번째 선수가 entry k번에 할당되는지 binary xik∈{0,1}
bi: 한 명의 투수 평균 소화 이닝 수
di: 16년도 두산 타자 선수별 RNG수치

각 번호에 배정되는 선수들의 능력을 얻기 위해 전체 선수가 각 타선번호에 해당되는지에 대한 여부를 나타내는 Decision Variable인 xik와 그 해당 타자의 능력치를 나타내는 DV인 cij를 사용한다. 또한 이 능력치들이 번호에 따라 중요도가 다르기 때문에 이를 위해 각 수준마다 측정된 변수 값ajk을 통해 통계적으로 유의미한 차이가 있는지 비교하였다.

3) 독립 변수

원래는 타율(BABIP)과 장타율(OPS)에는 상관관계가 있지만 독립적이라고 가정하고 parameter를 설정한다.

4) 종속 변수

본 실험에서 조사된 능력 값은 모두 적절한 parameter를 곱해 나타낸다. 이로 수정된 능력 값의 결과는 다음과 같다.

* [수식] 능력 Cij의 수정된 능력

$\sum\sum c_{ij}a_{jk}$

5) 통제 변수

독립 변수와 종속 변수를 제외한 모든 변수가 통제 변수이다. 본 목적식에서 제어한 통제 변수의 경우 계절이나 구장에 상관없이 능력치가 동일하다고 가정한다.

2-2. 제약식

본 29조는 38개의 제약식을 적용하였다.

투수에 대한 제약식은 총 13개다.

1) $\sum_{k=10}^{11} x_{1k} \le 1$, $\sum_{k=10}^{11} x_{2k} \le 1$, $\sum_{k=10}^{11} x_{3k} \le 1$, $\sum_{k=10}^{11} x_{4k} \le 1$, $\sum_{k=10}^{11} x_{5k} \le 1$, $\sum_{k=10}^{11} x_{6k} \le 1$, $\sum_{k=10}^{11} x_{7k} \le 1$, $\sum_{k=10}^{11} x_{8k} \le 1$, $\sum_{k=10}^{11} x_{9k} \le 1$, $\sum_{k=10}^{11} x_{10k} \le 1$

2) $\sum_{i=1}^{10} x_{i10} = 1$, $\sum_{i=1}^{10} x_{i11} = 1$

3) $b_i \ge 5 - M(1 - x_{ik})$ M은 무한에 가까운 수이나, 모델링을 위해서 6으로 가정한다.

포수에 대한 제약식은 총 3개다.

1) $x_{11,9} \le 1$, $x_{12,9} \le 1$

2) $x_{11,9} + x_{12,9} \le 1$

타자에 대한 제약식은 총 21개다. (포수 제외)

1) $\sum_{k=1}^{8} x_{13k} \le 1$, $\sum_{k=1}^{8} x_{14k} \le 1$, $\sum_{k=1}^{8} x_{15k} \le 1$, $\sum_{k=1}^{8} x_{16k} \le 1$, $\sum_{k=1}^{8} x_{17k} \le 1$, $\sum_{k=1}^{8} x_{18k} \le 1$, $\sum_{k=1}^{8} x_{19k} \le 1$, $\sum_{k=1}^{8} x_{20k} \le 1$, $\sum_{k=1}^{8} x_{21k} \le 1$, $\sum_{k=1}^{8} x_{22k} \le 1$, $\sum_{k=1}^{8} x_{23k} \le 1$, $\sum_{k=1}^{8} x_{24k} \le 1$

2) $\sum_{i=13}^{24} x_{i1} = 1$, $\sum_{i=13}^{24} x_{i2} = 1$, $\sum_{i=13}^{24} x_{i3} = 1$, $\sum_{i=13}^{24} x_{i4} = 1$, $\sum_{i=13}^{24} x_{i5} = 1$, $\sum_{i=13}^{24} x_{i6} = 1$, $\sum_{i=13}^{24} x_{i7} = 1$, $\sum_{i=13}^{24} x_{i8} = 1$

3) $d_i \ge -0.2 - N(1 - x_{ik})$ N은 무한에 가까운 수이나, 모델링을 위해서 임의의수 50으로 가정한다.

추가적인 제약식으로는

$x_{ik} \in 0,1$ 가 있다.

2-3. 데이터 수집 방법

모델링의 구체적인 데이터를 얻기 위해 kbreport.com 사이트에서 상대팀에 대한 우리팀 선수들의 데이터들을 확인할 수 있다. 또한 Statiz 사이트의 팀정보, 선수정보, 기록실의 자료들을 엑셀로 export해서 사용하였다.

Ⅲ. 관련이론

1) LP모델링

linear programming이라고도 한다. 어떠한 상황에서 여러 가능성이 있을 때, 그중에서 가장 적절한 것을 찾아내는 방법을 최적화이론이라고 한다. 선형계획법은 최적화이론의 한 분야로서 제약조건이 부등식 혹은 방정식으로 나타내지고 알고자 하는 값을 나타내는 목적함수 (Objective Function)가 존재하는 모델링이다. 좌표평면에 제한 조건을 도시하여 목적함수의 최댓값 또는 최솟값을 찾는 방법을 선형계획법의 기하학적인 방법이라고 한다.

2) Big M Modeling

제약식 한 개를 redundant하게 만들기 위해서 굉장히 큰 수 M을 활용하는 방법으로써, 거의 무한에 가까운 M을 사용하는 것이다. 예를 들어 $\sum_{i=1}^{n} a_i x_i \leq b$ 인 경우, RHS에 M을 더하게 된다면 이 제약식은 무의미 해진다.

3) 통계적 유의성, 실제적 유의성 및 실용적 유의성

성태제(2016)은 그의 책에서 통계적 유의성을 “연구자가 얻은 어떤 검정 통계 값을 자신이 설정한 유의수준에 입각하여 판단해 볼 때 영가설을 기각할 만큼 유의한 것을 뜻한다.”[7] 고 말하고 있다. 이에 반해 실제적 유의성은 “통계적 검정에 입각하지 않고 실제적 상황에서 연구자가 얻은 추정치, 즉 표본의 평균, 집단 간의 차이, 상관계수 등이 의미가 있는지에 대한 것”[8]이라고 밝혔다. 즉, 가설 검정을 통해 어떤 사실을 발견하였다고 하더라도 이 사실이 실제적으로 의미가 있는지는 별개라는 것이다. 실제로 두 대상 사이의 통계적 유의성이 존재하나 의미상 연관 관계가 미약할 경우 통계적 유의성은 존재하나 실제적 유의성이 존재하지 않는다고 말한다.

이와 연관 지어 생각할 수 있는 개념으로 ‘실용적 유의성’ 개념이 있다. Wickens et al.(2008)은 서로 다른 두 집단 간의 차이가 크지 않음에도 불구하고 통계적으로 다를 수 있다는 점을 주의해야한다고 서술하고 있다. 예를 들어 시스템을 개선시키기 위해 많은 돈을 지불하였으나 과제의 수행률이 미약하게 증가하는 경우, 통계적으로 유의하다고 하더라도 실용적 유의성이 확보되지 않을 수 있다고 말한다.[9]

Ⅳ. 실험 결과

본 실험의 Raw Data는 Excel 2010에서 작성한 자료를 토대로 <첨부 1. Raw data of the experiment>에 작성하였다.

표는 삼성에 대한 두산 선수들의 능력들의 수치이다.

---

7) “같은책, pp.294.”
8) ibid.
9) "Wickens, 앞의 책, pp.31-32"

#	선수명	팀명	상대	타석	타율	BABIP	볼넷%	삼진%	볼/삼	ISO	타수/홈런	OPS	RC	RC/27	wRC	SPD	wSB	wOBA	wRAA	WAR
1	오재원	두산	삼성	52	0.217	0.323	11.5	28.8	0.40	0.022	-	0.547	3.92	2.86	2.90	3.64	**0.38**	0.267	-3.78	-
2	정수빈	두산	삼성	28	0.115	0.130	3.6	10.7	0.33	0.000	-	0.263	0.34	0.36	-1.95	2.11	**0.34**	0.129	-5.55	-
3	오재일	두산	삼성	46	0.375	0.414	13.0	17.4	0.75	0.225	13.33	1.057	11.91	12.86	10.72	3.61	**0.22**	0.462	4.81	-
4	민병헌	두산	삼성	54	0.364	0.368	14.8	9.3	1.60	0.227	22.00	1.054	13.60	12.66	12.40	4.42	**0.22**	0.458	5.46	-
5	김재환*	두산	삼성	53	0.238	0.296	20.8	24.5	0.85	0.167	21.00	0.801	7.69	6.29	7.90	3.13	**0.22**	0.369	1.09	-
6	류지혁	두산	삼성	10	0.333	0.333	10.0	20.0	0.50	0.334	9.00	1.067	2.71	12.20	2.24	5.42	**0.17**	0.452	0.95	-
7	에반스	두산	삼성	61	0.269	0.263	13.1	16.4	0.80	0.308	13.00	0.954	10.60	6.98	11.31	0.13	**0.04**	0.409	3.47	-
8	양의지	두산	삼성	42	0.368	0.303	7.1	4.8	1.50	0.343	9.50	1.116	10.11	10.11	9.91	0.47	**0.03**	0.465	4.52	-
9	최주환	두산	삼성	25	0.364	0.364	0.0	4.0	0.00	0.091	-	0.830	4.07	6.86	3.69	3.96	**0.02**	0.368	0.47	-
10	국해성	두산	삼성	21	0.211	0.267	9.5	19.0	0.50	0.052	-	0.549	1.31	2.22	1.05	-0.02	**0.01**	0.261	-1.65	-
11	박세혁	두산	삼성	24	0.095	0.133	8.3	25.0	0.33	0.048	-	0.317	0.67	0.91	-1.04	1.81	**0.01**	0.158	-4.12	-
12	이원석	두산	삼성	4	0.250	0.000	0.0	25.0	0.00	0.750	4.00	1.250	1.00	9.00	1.03	-	**0.00**	0.489	0.52	-
13	이우성	두산	삼성	5	0.200	0.333	0.0	40.0	0.00	0.000	-	0.400	0.20	1.35	-0.10	1.81	**0.00**	0.184	-0.74	-
14	조수행	두산	삼성	1	0.000	0.000	0.0	0.0	-	0.000	-	0.000	0.00	0.00	-0.19	-	**0.00**	0.000	-0.32	-
15	김동한	두산	삼성	1	0.000	-	0.0	100.0	0.00	0.000	-	0.000	0.00	0.00	-0.19	-	**0.00**	0.000	-0.32	-
16	서예일	두산	삼성	0	-	-	-	-	-	-	-	-	-	-	-	-	**0.00**	-	-	-
17	박건우	두산	삼성	68	0.410	0.469	5.9	16.2	0.36	0.262	30.50	1.120	18.93	13.10	17.16	8.72	**-0.07**	0.483	8.43	-
18	김재호	두산	삼성	58	0.224	0.233	12.1	10.3	1.17	0.062	49.00	0.602	5.09	3.35	4.16	4.03	**-0.25**	0.285	-3.29	-
19	허경민	두산	삼성	68	0.190	0.200	5.9	10.3	0.57	0.064	63.00	0.504	3.93	2.00	1.92	5.05	**-0.25**	0.237	-6.82	-
20	정진호	두산	삼성	4	0.000	0.000	25.0	0.0	-	0.000	-	0.250	0.00	0.00	-0.09	4.46	**-0.46**	0.181	-0.60	-

앞서 이론에서 언급했듯, 현재의 능력치가 곧바로 타선의 선수의 능력치를 의미한다고 보기 어렵다. 종속변수에는 선수의 타순에 따른 parameter ajk가 존재하기 때문이다.
데이터가 너무 방대하기 때문에 java를 통해 정리한다.

Ⅴ. 결과 분석

본 장에서는 지금까지 수집된 데이터에 적절한 parameter가 적용되었는지 확인하고 모델링의 결과값을 찾는다. 이후에는 지금까지의 타선과 비교하여 선발목록의 차이가 있는지 확인해본다.

5-1. Parameter 결정

총 열 팀의 프로야구 구단이 있다. 각 팀별 1번 타자의 능력치: 도루율, 출루율, 장타율, 타율을 조사하였다. 예를 들어, Q라는 타자의 도루율이a, 출루율이 b, 장타율이 c, 타율이 d라고 하자. a+b+c+d의 값이 e라면, 이 $\frac{a}{e}+\frac{b}{e}+\frac{c}{e}+\frac{d}{e}=1$ 의 식(1)을 도출할 수 있다. 그리고, 결정변수를 다음과 같이 설정한다. a_{jk} j는 1, 2, 3, 4이고, k는 1, 2, 3, 4, 5, 6, 7, 8, 9로 정의한다. 결정변수 하나는 위 식(1)의 구성요소인 분수들의 값을 의미한다.

또한, 팀 전체에서 타자와 투수의 중요도 가중치를 설정하기 위해 WAR이라는 지표를 사용하였다. WAR이란 대체 선수 대비 승리 기여도를 의미하며, Wins Above Replacement의 약자이다. $\frac{RAR}{(R/W)}$=WAR 로 계산된다. RAR은 타격에서의 기여도, 수비에서의 기여도, 포지션에 따른 조정, 주루에서의 기여도의 합으로 계산되며, (R/W)는 1승에 해당하는 득점을 나타내는 지표이다.

프로야구 전체 팀의 타자 : 투수 WAR 비율을 구하면 4.59 : 3.74가 나온다. 본 29조가

식(1)에서 한 타선의 파라미터의 합이 1이기 때문에 타선 9개의 합을 9가 된다. 그리하여, 다음과 같은 비례식을 작성할 수 있다. 4.59 : 3.74 = 9 : 7.33 이 된다. 하지만, 본 29조가 가정한 투수는 총 2명이기 때문에, $a_{5,10}$과 $a_{5,11}$은 7.33을 2로 나눈 3.67의 값이 되어야 한다.

ajk	도루	출루	장타	타율	투수능력
k/j	1	2	3	4	5
1	0.368636	0.2101	0.245146	0.176118	0
2	0.388448	0.198908	0.246839	0.165805	0
3	0.311283	0.225422	0.279004	0.184291	0
4	0.205346	0.254563	0.345306	0.194785	0
5	0.324133	0.21571	0.283003	0.177154	0
6	0.293759	0.229107	0.284844	0.19229	0
7	0.23655	0.242836	0.308732	0.211882	0
8	0.324411	0.246951	0.239512	0.189125	0
9	0.380736	0.211814	0.231595	0.175855	0
10	0	0	0	0	3.74
11	0	0	0	0	3.74

프로야구 전체 팀의 주전선수들의 데이터의 평균을 구해 ajk의 값을 구할 수 있었다. ajk를 구하기 위한 데이터는 Appendix에 첨부해놓았다.

5-2. Java를 통한 능력상수결정

```
package Assignment;

public class RealSize {
	public static void main(String[] args){
		int Entry = 11;
		int Dusan = 24;
		int Spec =5;
		double[]
D={1.54,0.13,0.027,0.4,-0.03,0.1,0.06,0.39,0.26,-0.05,-0.22,-10.5,-0.05,5.16,7.73,-0.54,6.2
5,-1.49,-0.6,0.33,-4.6,-3.7,-1.4
};//D=16년도 두산 타자 선수별 RNG수치
		double[]
B={6.32,1.02,0.92,1.07,1.12,0.924,0.55,2.67,6.22,6.12,0,0,0,0,0,0,0,0,0,0,0,0,0};
//B=각 순서 선수의 평균 소화 이닝 수
		double[][] A= {{0.368635679,0.210100174,0.24514642,0.176117727,0
},
{0.388448276,0.198908046,0.24683908,0.165804598,0
},
{0.311282899,0.225421928,0.279003895,0.184291278,0
},
{0.205345502,	0.254563233,	0.345306389,
0.194784876,	0
},
{0.324133389,	0.215709753,	0.283003053,
0.177153805,	0
},
{0.29375922, 0.229106536,	0.28484382,
0.192290424,	0
},
{0.236550419,	0.242835902,	0.30873209,
0.21188159,	0
},
{0.324411381,	0.246951453,	0.239512233,
0.189124933,	0
},
{0.380735504,	0.211814289,	0.231595405,
0.175854803,	0
},
{0,	0,	0,	0,	3.74
},
{0,	0	,0,	0,	3.74
}};//Aki : k entry에 j번째 능력 비중
		double[][] C= {{0,	0,	0,	0,	4.85
},
{0,	0,	0,	0,	4.29
},
{0,	0	,0,	0,	5.16
},
{0,	0	,0	,0,	5.07
},
{0,	0,	0	,0,	5.65
},
{0,	0,	0	,0,	3.51
},
{0,	0	,0,	0,	4.39
},
{0,	0,	0	,0,	4.7
},
{0,	0,	0	,0,	4.68
},
{0,	0,	0	,0,	4.09
},
{0.5,	0.313,	0.43,	0.264,	0
},
{0	,0.174	,0.143,	0.095,	0
},
{0.66	,0.313,	0.313,	0.239,	0
},
{0	,0.361,	0.466,	0.284,	0
},
```

```
                                        {0.631,  0.396    ,0.43    ,0.32,    0
},                                      {0.2,    0.349    ,0.315,  0.268,   0
},                                      {0.5,    0.37     ,0.486,  0.292    ,0
},                                      {1,      0.364,    0.6,     0.3,     0
},                                      {0       ,0.377    ,0.577,  0.269,   0
},                                      {0,      0,        0        ,0,      0
},                                      {0       ,0.279    ,0.421,  0.211,   0
},                                      {0.3,    0.418    ,0.513,  0.34,    0
},                                      {1,      0.375    ,0.414,  0.224,   0
},                                      {0.67,   0.434    ,0.633,  0.398,   0
}};//C 선수들의 능력
                    for(int i=0;i<Dusan;i++){
                              for(int k=0; k< Entry; k++){
                                        for(int j=0;j<Spec; j++){
                                        System.out.print(C[i][j]*A[k][j]+"
X_"+(i+1)+"_"+(k+1)+" + ");
                                                  }//목적식  부분 출력
                                        }
                                        System.out.println();
                    for(int i=0;i<Entry;i++){
                              System.out.print("C"+(i+1)+": ");
                              for(int j=0;j<Dusan;j++){
                                        System.out.print("X_"+(j+1)+"_"+(i+1)+" + ");
                              }
                              System.out.println("= 1");
                    }//제약조건, 각 범주당 한 명씩만 선수를 할당,
                    for(int i=0;i<Dusan;i++){
                              System.out.print("C"+(i+12)+": ");
                              for(int j=0;j<Entry;j++){
                                        System.out.print("X_"+(i+1)+"_"+(j+1)+" + ");
                              }
                              System.out.println("<= 1");
                    }// 모든 선수는 <=1로 한 포지션 이하에 할당
                    System.out.println();

                    System.out.println("Bounds");
                    System.out.println("Binaries");
                    for(int i=0;i<Entry;i++){
                              for(int j=0;j<Dusan;j++){
                                        System.out.print("X_"+(j+1)+"_"+(i+1)+" ");
                              }//Binary 값 선언
                    }
                    System.out.println("Ends");

                    for(int k=0;k<8;k++){
                              for(int i=0;i<Dusan;i++){
                                        System.out.println(" C"+(i+36+Dusan*k)+": 50
X_"+(i+1)+"_"+(k+1)+ " <= " +(0.2+50+D[i]));
                              }         }//일정 방어 능력 이하의 타자선수 선택을
제한(-0.2)
                    for(int k=0;k<2;k++){
                              for(int i=0;i<Dusan;i++){
                                        System.out.println(" C"+(i+228+Dusan*k)+": 50
X_"+(i+1)+"_"+(k+10)+" <= "+(B[i]+ 6 + 50) );
                              }         }//일정 이상 (6회) 던질 수 있는 투수 선정
          }
```

승리한 경기의 선수들의 능력의 평균을 구해 각 타선마다 중요도를 배분한다.

따라서 본 보고서에서는 타선에 따른 능력의 중요도를 타당화 하기위해 data를 분석에 사용하였다.

5-2. Gurobi을 통한 결과의 해석

Gurobi를 통한 결과는 다음과 같다. (Gurobi 모델은 다다음장<모델 및 알고리즘>에서 설명한다.)

투수 : 유희관, 장원준

타수 : 류지혁(1), 김재환(2), 양의지(3), 에반스(4), 박건우(5), 오재일(6), 최주환(7), 오재원(8), 허경민(9)

Ⅵ. 결론

6-1. 모델링의 결론

모델링의 최종적인 결론은 다음과 같다.

삼성 주전에 대한 두산의 최적 타선은 다음과 같다.

투수 : 유희관, 장원준

타수 : 류지혁(1), 김재환(2), 양의지(3), 에반스(4), 박건우(5), 오재일(6), 최주환(7), 오재원(8), 허경민(9)

6-2. 모델링의 한계점 및 후속 연구 방향

스포츠 선수의 능력을 수치화시켜 이 값들을 적절히 배치함으로써 선수타선을 최적화 시키는 모델링을 진행하여 스포츠 경기에 대한 감독의 개인적인 능력과 다르게 데이터를 통한 통계적 분석으로 효율적인 경기를 이끌어나갈 수 있음을 알게되었다. 이 과정에서 통계 패키지의 다양한 기능들을 활용하고 몰랐던 기능들을 습득하면서 새로운 사실들을 알아갈 수 있었던 계기가 되었음은 분명하다.

두산의 실제 주전선수 타선과 모델링한 선수가 차이가 있었다. 모델링을 하면서 최초의 목적 이외에도 발견하게 된 몇 가지 사실이 있다. 첫째로 최적화 모델은 보통 승률을 계산하여 그 값을 다른 값과 비교하는 등 최적에 대한 증명이 대부분 존재하는데, 우리의 모델에서는 타선의 최적을 설명할 방법이 없다. 파라메타를 통계적으로 분석하여 타당성을 더했으나, 이는 과거의 행적으로 인한 데이터이기 때문에 매우 타당하다고는 볼 수 없다. 프로야구 게임 시뮬레이션을 돌리려고 했지만 이는 게임의 설정에 따른 선수의 능력치가 다르고, 게임 능력에 따른 차이가 있기 때문에 확신할 수 없는 자료이다.

두 번째로 환경에 따른 변화이다. 선수에 따라서 계절에 따른 능력차이가 있을 수 있다. 또한 상대팀 전체가 아닌 상대팀 투수에 따라 즉, 좌투냐 우투냐에 따라 차이가 있을 수 있다. 데이터가 충분하지 않기 때문에 기간을 넓게 설정했지만, 해마다 선수들의 능력이 달라지기 때문에 변화가 있을 수 있다.

세 번째로 모델링이 자체적으로 가지고 있는 일부 오류를 해결해야 한다. 상대팀의 투수를 3명으로 가정을 했지만 보통 5명의 투수가 교체되기도 하며 상대팀 투수의 변화와 지명타자가 경기에 영향을 미칠 수 있음을 무시한 것이다. 또한 수비 능력치에 따른 제약식만 두었을 뿐 1루수,2루수,유격수 등 수비포지션에 대한 고려를 하지 않았던 것이 오류이다.

네 번째로 선수들의 체력을 고려하지 않은 것에서 오는 오류이다. 투수의 경우 선발투수는 한 경기를 뛰면 다음경기는 쉬어야한다. 하지만 우리가 모델링한 유희관과 장원준선수의 경우 둘다 선발투수이기 때문에 삼성을 이기기 위한 최적타선의 선수이기는 하나 다음 경기의 승패에 영향을 미칠 것으로 보인다.

이 세가지 오류의 원인을 개선할 방법은 다음과 같다.

첫째는 parameter를 설정함에 있어 좀 더 방대한 데이터들을 가지고 분석한다면 좀 더 정확한 모델링이 될 것이다.

둘째는 수비 포지션이 정해져 있을 경우 수비 포지션의 명수의 제약도 넣어야 한다.

셋째는 투수의 설정 수를 늘려(예: 2명->5명) 최소소화이닝의 제약식을 선발투수에게만 적용하는 방식을 택한다.

Ⅶ. 모델 및 알고리즘

7-1. 관련이론

1) Linear Programming(LP)

Linear programming (LP) (also called linear optimization) is a method to achieve the best outcome (such as maximum profit or lowest cost) in a mathematical model whose requirements are represented by linear relationships. Linear programming is a special case of mathematical programming (mathematical optimization).

2) Integer Programming(IP)

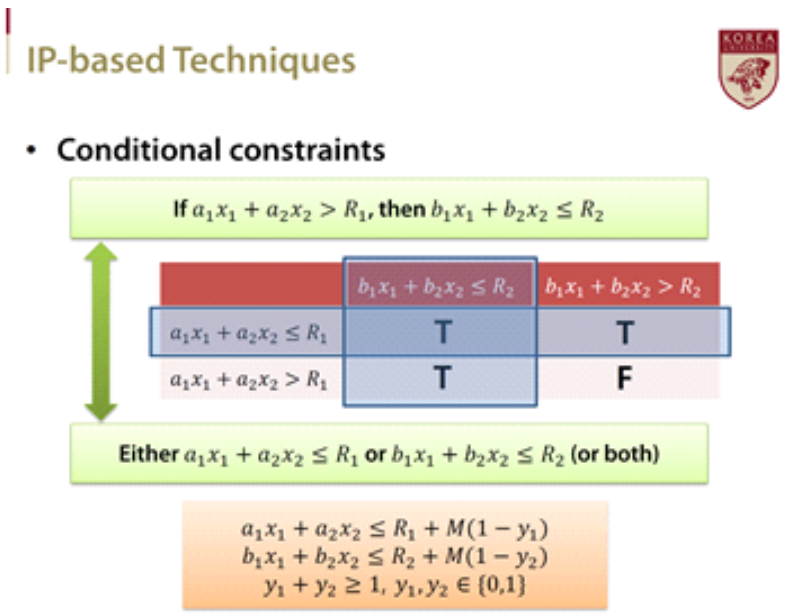
An integer programming problem is a mathematical optimization or feasibility program in which some or all of the variables are restricted to be integers. In many settings the term refers to integer linear programming (ILP), in which the objective function and the constraints (other than the integer constraints) are linear.

3) Knapsack Problem

The knapsack problem or rucksack problem is a problem in combinatorial optimization: Given a set of items, each with a weight and a value, determine the number of each item to include in a collection so that the total weight is less than or equal to a given limit and the total value is as large as possible. It derives its name from the problem faced by someone who is constrained by a fixed-size knapsack and must fill it with the most valuable items.

The problem often arises in resource allocation where there are financial constraints and is studied in fields such as combinatorics, computer science, complexity theory, cryptography, applied mathematics, and daily fantasy sports.

4) LP based Techinics



7-2. 프로야구타선최적 모델 (Gurobi)

다음은 Gurobi 실험 환경을 통해 구성한 프로야구타선최적 모델이다.

```
Maximize
   18.139 X_1_10 + 18.139 X_1_11 + 16.0446 X_2_10 + 16.0446 X_2_11
    + 19.2984 X_3_10 + 19.2984 X_3_11 + 18.9618 X_4_10 + 18.9618 X_4_11
    + 21.131 X_5_10 + 21.131 X_5_11 + 13.1274 X_6_10 + 13.1274 X_6_11
 ......//이런식으로 모든 Xik와 상수들을 곱해 더한다.
Subject To
 C1: X_1_1 + X_2_1 + X_3_1 + X_4_1 + X_5_1 + X_6_1 + X_7_1 + X_8_1 + X_9_1
    + X_10_1 + X_11_1 + X_12_1 + X_13_1 + X_14_1 + X_15_1 + X_16_1 + X_17_1
    + X_18_1 + X_19_1 + X_20_1 + X_21_1 + X_22_1 + X_23_1 + X_24_1 = 1
//이런식으로 C275까지 작성한다.

Bounds
Binaries
// X_1_1부터 X_24_11 첨부한다.
End
```

Ⅷ. Reference

-Wikipedia
-STATIZ
-한국프로야구 통계기록실 KBReport.com (케이비리포트)
-`제이버 매트릭스, 제이스의 〃이름 없는 블로그 : 네이버블로그
-경영공학개론 수업자료

Ⅸ. Appendix

<첨부 1. Raw data of the experiment>

	cij	도루율	출루율	장타율	타율	투수이닝	RNG	FIP
	i/j	1	2	3	4	bi	di	
유희관	1					6.32	1.54	4.85
이현승	2					1.02	0.13	4.29
김성배	3					0.92	0	5.16
김승희	4					1.07	0.27	5.07
김강률	5					1.12	0.4	5.65
고봉재	6					0.924	-0.03	3.51
함덕주	7					0.55	0.1	4.39
이현호	8					2.67	0.06	4.7
장원준	9					6.22	0.39	4.68
니퍼트	10					6.12	0.26	4.09
양의지	11	0.5	0.313	0.43	0.264	0	-0.05	
박세혁	12	0	0.174	0.143	0.095	0	-0.22	
허경민	13	0.66	0.313	0.313	0.239	0	-10.5	
최주환	14	0	0.361	0.466	0.284	0	-0.05	
오재원	15	0.631	0.396	0.43	0.32	0	5.16	
김재호	16	0.2	0.349	0.315	0.268	0	7.73	
오재일	17	0.5	0.37	0.486	0.292	0	-0.54	
류지혁	18	1	0.364	0.6	0.3	0	6.25	
에반스	19	0	0.377	0.577	0.269	0	-1.49	
신성현	20	0	0	0	0	0	-0.6	
국해성	21	0	0.279	0.421	0.211	0	0.33	
민병헌	22	0.3	0.418	0.513	0.34	0	-4.6	
김재환	23	1	0.375	0.414	0.224	0	-3.7	
박건우	24	0.67	0.434	0.633	0.398	0	-1.4	
		<삼성주전에 대한 두산선수들의 능력 데이터>						

1번	도루율	홈루율	장타율	타율	합
기아	0.6875	0.35	0.515	0.297	1.8495
엔씨	0.8	0.371	0.387	0.314	1.872
두산	0.25	0.385	0.442	0.314	1.391
스크	1	0.355	0.274	0.253	1.882
엘지	0.66	0.355	0.353	0.287	1.655
넥센	0.8	0.365	0.425	0.316	1.906
롯데	0	0.378	0.657	0.343	1.378
한화	0.428	0.366	0.403	0.301	1.498
kt	0.823	0.318	0.328	0.284	1.753
삼성	0.789	0.312	0.364	0.271	1.736
평균	**0.62375**	**0.3555**	**0.4148**	**0.298**	**1.69205**

2번	도루율	홈루율	장타율	타율	합
기아	0.5	0.365	0.437	0.337	1.639
엔씨	0.5	0.359	0.277	0.246	1.382
두산	1	0.323	0.508	0.288	2.119
스크	0.5	0.297	0.672	0.297	1.766
엘지	1	0.455	0.435	0.348	2.238
넥센	0.714	0.356	0.411	0.301	1.782
롯데	0.75	0.418	0.496	0.336	2
한화	0.625	0.32	0.405	0.276	1.626
kt	0.67	0.284	0.323	0.2	1.477
삼성	0.5	0.284	0.331	0.256	1.371
평균	**0.6759**	**0.3461**	**0.4295**	**0.2885**	**1.74**

3번	도루율	홈루율	장타율	타율	합
기아	1	0.214	0.267	0.171	1.652
엔씨	1	0.373	0.366	0.323	2.062
두산	0.875	0.376	0.435	0.288	1.974
스크	0.25	0.418	0.618	0.29	1.576
엘지	0.75	0.399	0.423	0.337	1.909
넥센	0	0.388	0.451	0.336	1.175
롯데	0	0.386	0.424	0.303	1.113
한화	1	0.432	0.663	0.378	2.473
kt	0	0.444	0.496	0.396	1.336
삼성	0.4	0.39	0.585	0.301	1.676
평균	**0.5275**	**0.382**	**0.4728**	**0.3123**	**1.6946**

4번	도루율	홈루율	장타율	타율	합
기아	0	0.455	0.643	0.343	1.441
엔씨	0.5	0.33	0.384	0.22	1.434
두산	0.66	0.418	0.55	0.329	1.957
스크	0.5	0.385	0.698	0.26	1.843
엘지	0.33	0.33	0.406	0.269	1.335
넥센	0	0.417	0.544	0.342	1.303
롯데	0	0.438	0.55	0.368	1.356
한화	0.66	0.393	0.548	0.319	1.92
kt	0	0.359	0.479	0.263	1.101
삼성	0.5	0.38	0.495	0.275	1.65
평균	**0.315**	**0.3905**	**0.5297**	**0.2988**	**1.534**

5번	도루율	홈루율	장타율	타율	합
기아	1	0.372	0.48	0.31	2.162
엔씨	0.5	0.363	0.524	0.329	1.716
두산	1	0.429	0.554	0.339	2.322
스크	0.4	0.317	0.508	0.261	1.486
엘지	0.8	0.303	0.336	0.259	1.698
넥센	0.714	0.353	0.473	0.273	1.813
롯데	0	0.37	0.458	0.296	1.124
한화	0	0.426	0.532	0.345	1.303
kt	0	0.341	0.385	0.28	1.006
삼성	1	0.329	0.477	0.267	2.073
평균	**0.5414**	**0.3603**	**0.4727**	**0.2959**	**1.6703**

6번	도루율	홈루율	장타율	타율	합
기아	0.5	0.379	0.457	0.305	1.641
엔씨	0.33	0.387	0.456	0.283	1.456
두산	0.75	0.396	0.429	0.317	1.892
스크	0	0.289	0.39	0.236	0.915
엘지	0	0.313	0.452	0.29	1.055
넥센	0	0.349	0.387	0.269	1.005
롯데	1	0.404	0.359	0.303	2.066
한화	1	0.291	0.373	0.235	1.899
kt	1	0.352	0.609	0.348	2.309
삼성	0	0.412	0.529	0.412	1.353
평균	**0.458**	**0.3572**	**0.4441**	**0.2998**	**1.5591**

7번	도루율	홈루율	장타율	타율	합
기아	0	0.376	0.45	0.29	1.116
엔씨	0	0.336	0.4	0.314	1.05
두산	0	0.385	0.502	0.294	1.181
스크	0	0.346	0.507	0.301	1.154
엘지	1	0.348	0.455	0.318	2.121
넥센	0	0.333	0.417	0.284	1.034
롯데	1	0.417	0.545	0.364	2.326
한화	0.5	0.351	0.416	0.416	1.683
kt	0	0.363	0.465	0.273	1.101
삼성	1	0.338	0.411	0.281	2.03
평균	**0.35**	**0.3593**	**0.4568**	**0.3135**	**1.4796**

8번	도루율	홈루율	장타율	타율	합
기아	1	0.286	0.27	0.22	1.776
엔씨	0	0.342	0.306	0.271	0.919
두산	0.556	0.339	0.305	0.22	1.42
스크	0	0.264	0.304	0.232	0.8
엘지	0	0.33	0.372	0.291	0.993
넥센	0	0.333	0.256	0.231	0.82
롯데	0.66	0.388	0.378	0.244	1.67
한화	0.5	0.326	0.23	0.216	1.272
kt	0.8	0.286	0.379	0.26	1.725
삼성	0.714	0.326	0.323	0.281	1.644
평균	**0.423**	**0.322**	**0.3123**	**0.2466**	**1.3039**

9번	도루율	홈루율	장타율	타율	합
기아	0.33	0.41	0.45	0.36	1.55
엔씨	0.778	0.362	0.365	0.341	1.846
두산	0.833	0.303	0.267	0.233	1.636
스크	0	0.326	0.37	0.244	0.94
엘지	1	0.316	0.35	0.279	1.945
넥센	0	0.235	0.429	0.214	0.878
롯데	0.5	0.271	0.265	0.176	1.212
한화	0.66	0.315	0.323	0.283	1.581
kt	1	0.294	0.333	0.245	1.872
삼성	0.5	0.284	0.255	0.212	1.251
평균	**0.5601**	**0.3116**	**0.3407**	**0.2587**	**1.4711**

<ajk를 구하기 위한 프로야구 전체 구단의 주전타선 능력 데이터>