

(1)

昨今日本プロ野球では、国内の独立リーグでプレーする選手が指名されることが年々増加している。独立リーグはトーナメントが主の社会人野球と違い、加盟チームが 50~70 試合程のリーグ戦を行い、独自に成績を付ける。また、NPB 球団の 2・3 軍との交流戦をリーグ戦として年に 10 試合以上行うリーグもあることから、社会人野球よりも成績の信ぴょう性があり年間を通じ NPB 球団へのアピールができるなどのメリットがあり、実際に 2023 年のプロ野球ドラフト会議では育成契約を含め過去最多の 23 選手が指名されている。

しかし、社会人野球よりも十分な試合データがあるにも関わらずその成績からプロ野球でどのような成績を残すかの数値的な予測はあまり行われておらず、独立リーグの強みを活かしていきれていないという問題があり、NPB 球団側も入団後の戦力構想が狂うことや指導方針を変更することがあることが考えられる。

(2)

独立リーグは選手の入替えが激しく、複数年での予測は不向きだと考え、今回は昨年福岡ソフトバンクホークス 3 軍との交流戦を各チーム年 12 試合行い、データがまとまっている四国アイランドリーグの昨年の成績データを用いる。

昨年の四国アイランドリーグのチームの打者成績:<https://data.iblj.co.jp/stats/2023/hitting>

昨年の四国アイランドリーグのホークスの成績:<https://data.iblj.co.jp/teams/23704/2023>

昨年の福岡ソフトバンクホークスの 2 軍選手成績:

<https://baseball-data.com/23/stats-farm/hitter2-h/tpa-1.html>

(3)

今回は打者成績の予測を行う。2 軍との比較として、ホークス 3 軍の試合に出ている選手で 2 軍公式戦の成績の内、2・3 軍合計で 90 打席かつ四国アイランドリーグで 35 打席以上かつ 2 軍で 45 打席以上出場している 8 選手の成績をモデルの推定に、四国アイランドリーグのチームの選手の内、150 打席以上出場した 35 選手のデータを用いて、選手の 2 軍成績を予測する。また、データとして打率、試合数、打率、打席、打数、得点、安打、二塁打、三塁打、本塁打、塁打、打点、三振、三振率、四球、四球率、死球、犠打、犠飛、併殺打、出塁率、長打率の内、安打数などは同じ率でも打席数で変われることを考慮し四国アイランドリーグの打率、三振率、四球率、出塁率、長打率を説明変数の候補として 2 軍での打率、三振率、四球率、出塁率、長打率をそれぞれ目的変数とし、ステップワイズによる重回帰モデルで予測する。

kp が三振率、bbp が四球率、obp が出塁率、tyo が長打率。

```

R Console

> bbl <- read.csv(file="C:/Users/hiu/Desktop/B11/Book1.csv", header=T, fileEncoding="Shift-JIS")
> bb <- bbl[,1]
> testp=8;
> prep=testp+35;
>
>
> train.dat <- bb[1:testp,]
> test.dat <- bb[(testp+1):prep,]
> daritu <- bb[,1]
> kp <- bb[,13]
> bbp <- bb[,15]
> obp <- bb[,20]
> tyo <- bb[,21]
> daritu2 <- bb[,22]
> daseki2 <- bb[,24]
> kp2 <- bb[,34]
> bbp2 <- bb[,36]
> obp2 <- bb[,41]
> tyo2 <- bb[,42]
>
> r1 <- lm(daritu2 ~ daritu, data=train.dat);
> r1.stp <- step(r1,direction="forward",scope=list(upper=~daritu + kp + bbp + obp + tyo))
Start: AIC=-46.38
daritu2 ~ daritu

```

Start: AIC=-46.38
daritu2 ~ daritu

```

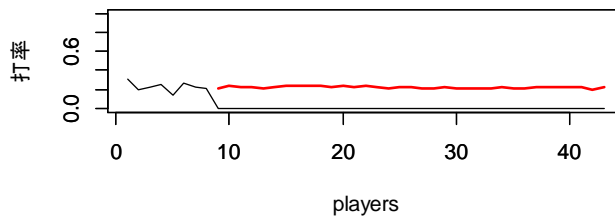
      Df Sum of Sq      RSS      AIC
<none>      0.014731 -46.378
+ kp      1 0.00154601 0.013185 -45.265
+ tyo      1 0.00042396 0.014307 -44.611
+ bbp      1 0.00009876 0.014632 -44.432
+ obp      1 0.00000000 0.014731 -44.378
> step.pred <- predict(r1.stp,test.dat)
> par(mfrow=c(2,1))
> plot(bb[1:prep,22], type="l", ylim=c(0,1),
+ xlab="", ylab="")
> par(new=T)
> pred.data <- as.vector(c(rep(NA, testp), step.pred))
> plot(pred.data, type="l", lty=c(1), lwd=c(2),
+ ylim=c(0,1), xlab="players", ylab="打率", col="red",
+ main="2軍打率予測")
>

```

```
> step.pred
```

9	10	11	12	13	14	15	16	17
0.2155244	0.2379299	0.2201817	0.2241455	0.2141115	0.2274691	0.2369527	0.2306095	0.2341941
18	19	20	21	22	23	24	25	26
0.2348081	0.2166929	0.2377620	0.2157335	0.2383636	0.2188173	0.2154440	0.2224857	0.2210790
27	28	29	30	31	32	33	34	35
0.2143496	0.2144111	0.2200100	0.2083228	0.2052262	0.2103542	0.2128908	0.2202761	0.2077514
36	37	38	39	40	41	42	43	
0.2139488	0.2207519	0.2253522	0.2270250	0.2183725	0.2207519	0.2011169	0.2223567	

2軍打率予測



```
R Console
> r2 <- lm(kp2 ~ kp, data=train.dat);
> r2.stp <- step(r2, direction="forward", scope=list(upper=~daritu + kp + bbp + obp + tyo))
Start: AIC=-41.42
kp2 ~ kp

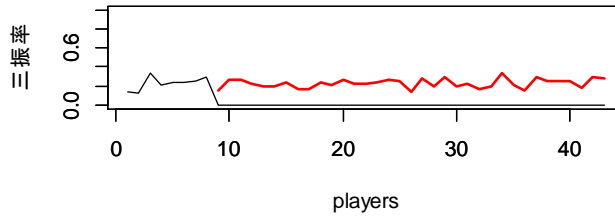
      Df Sum of Sq  RSS   AIC
+ obp   1 0.0088608 0.018532 -42.541
+ bbp   1 0.0069634 0.020430 -41.762
<none>          0.027393 -41.415
+ tyo   1 0.0044064 0.022987 -40.818
+ daritu 1 0.0024133 0.024980 -40.153

Step: AIC=-42.54
kp2 ~ kp + obp

      Df Sum of Sq  RSS   AIC
<none>          0.018532 -42.541
+ bbp   1 0.0016493 0.016883 -41.287
+ daritu 1 0.0007045 0.017828 -40.852
+ tyo   1 0.0002445 0.018288 -40.648
> step.pred <- predict(r2.stp, test.dat)
> par(mfrow=c(2,1))
> plot(bb[1:prep,34], type="l", ylim=c(0,1),
+ xlab="", ylab="")

> step.pred <- predict(r2.stp, test.dat)
> par(mfrow=c(2,1))
> plot(bb[1:prep,34], type="l", ylim=c(0,1),
+ xlab="", ylab="")
> par(new=T)
> pred.data <- as.vector(c(rep(NA, testp), step.pred))
> plot(pred.data, type="l", lty=c(1), lwd=c(2),
+ ylim=c(0,1), xlab="players", ylab="三振率", col="red",
+ main="2軍三振率予測")
> step.pred
      9      10      11      12      13      14      15      16      17
0.1490585 0.2632957 0.2671503 0.2237515 0.1930395 0.1883841 0.2416645 0.1664838 0.1679526
     18      19      20      21      22      23      24      25      26
0.2410286 0.2033963 0.2693432 0.2177875 0.2167191 0.2331761 0.2643265 0.2562567 0.1361719
     27      28      29      30      31      32      33      34      35
0.2803486 0.1943547 0.2869118 0.1936855 0.2186864 0.1722431 0.1874892 0.3405963 0.2078281
     36      37      38      39      40      41      42      43
0.1591250 0.2994096 0.2486792 0.2454744 0.2537237 0.1749699 0.2960979 0.2797447
```

2軍三振率予測



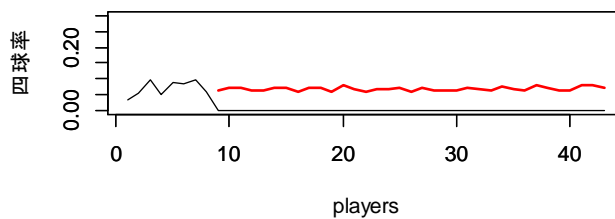
```
> r3 <- lm(bbp2 ~ bbp, data=train.dat);
> r3.stp <- step(r3, direction="forward", scope=list(upper=~daritu + kp + bbp + obp + tyo))
Start: AIC=-57.46
bbp2 ~ bbp
```

```
      Df Sum of Sq      RSS      AIC
<none>          0.0036876 -57.458
+ tyo      1 2.7066e-04 0.0034169 -56.068
+ kp       1 2.0654e-05 0.0036669 -55.503
+ obp      1 2.6240e-06 0.0036849 -55.464
+ daritu   1 1.9660e-06 0.0036856 -55.462
> step.pred <- predict(r3.stp, test.dat)
> par(mfrow=c(2,1))
> plot(bb[1:prep,36], type="l", ylim=c(0,0.3),
+ xlab="", ylab="")
> par(new=T)
> pred.data <- as.vector(c(rep(NA, testp), step.pred))
> plot(pred.data, type="l", lty=c(1), lwd=c(2),
+ ylim=c(0,0.3), xlab="players", ylab="四球率", col="red",
+ main="2軍四球率予測")
> step.pred
```

9	10	11	12	13	14	15	16
0.06312026	0.06953343	0.07295364	0.06279348	0.06359414	0.07021747	0.07120126	0.05869678
17	18	19	20	21	22	23	24

9	10	11	12	13	14	15	16
0.06312026	0.06953343	0.07295364	0.06279348	0.06359414	0.07021747	0.07120126	0.05869678
17	18	19	20	21	22	23	24
0.06988707	0.07304727	0.05969377	0.07939415	0.06749196	0.05679317	0.06834887	0.06593276
25	26	27	28	29	30	31	32
0.06906541	0.05787460	0.06934190	0.06427609	0.06064090	0.06396339	0.07188297	0.06662906
33	34	35	36	37	38	39	40
0.06465444	0.07550110	0.06686970	0.06223431	0.07975208	0.07250041	0.06433472	0.06187217
41	42	43					
0.07806459	0.07825279	0.07216939					

2軍四球率予測



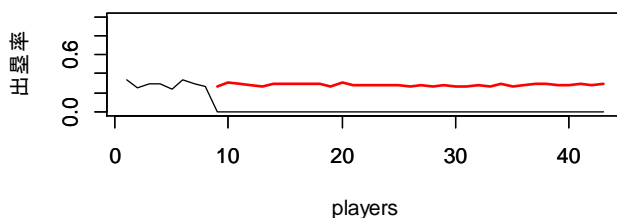
```

> r4 <- lm(obp2 ~ obp, data=train.dat);
> r4.stp <- step(r4,direction="forward",scope=list(upper=~daritu + kp + bbp + obp + tyo))
Start: AIC=-52.51
obp2 ~ obp

      Df Sum of Sq    RSS   AIC
<none>          0.0068456 -52.509
+ tyo      1 0.00081574 0.0060298 -51.524
+ kp       1 0.00035859 0.0064870 -50.939
+ daritu   1 0.00003939 0.0068062 -50.555
+ bbp      1 0.00000172 0.0068438 -50.511
> step.pred <- predict(r4.stp,test.dat)
> par(mfrow=c(2,1))
> plot(bb[1:prep,41], type="l", ylim=c(0,1),
+ xlab="", ylab="")
> par(new=T)
> pred.data <- as.vector(c(rep(NA, testp), step.pred))
> plot(pred.data, type="l", lty=c(1), lwd=c(2),
+ ylim=c(0,1), xlab="players", ylab="出塁率", col="red",
+ main="2軍出塁率予測")
> step.pred
      9      10      11      12      13      14      15      16      17
0.2706660 0.3046050 0.2871930 0.2783849 0.2717174 0.2890296 0.2972343 0.2868573 0.2927982
      18      19      20      21      22      23      24      25      26
0.2706660 0.3046050 0.2871930 0.2783849 0.2717174 0.2890296 0.2972343 0.2868573 0.2927982
      18      19      20      21      22      23      24      25      26
0.3001733 0.2694042 0.3044687 0.2802303 0.2825609 0.2822573 0.2756916 0.2849187 0.2706893
      27      28      29      30      31      32      33      34      35
0.2769604 0.2713327 0.2745379 0.2709811 0.2699987 0.2770015 0.2708628 0.2888952 0.2703216
      36      37      38      39      40      41      42      43
0.2757463 0.2969309 0.2906007 0.2810179 0.2730509 0.2895872 0.2769913 0.2866653

```

2軍出塁率予測



最後に一般に打者の貢献度を示す出塁率+長打率=OPS を表示

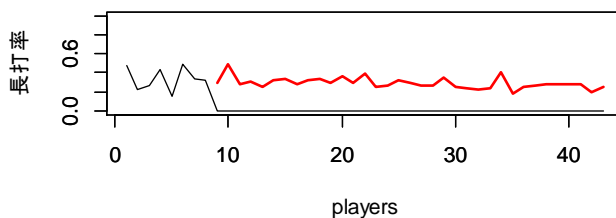
```

> r5 <- lm(tyo2 ~ tyo-1, data=train.dat);
> r5.stp <- step(r5,direction="forward",scope=list(upper=~daritu + kp + bbp + obp + tyo))
Start: AIC=-33.59
tyo2 ~ tyo - 1

          Df Sum of Sq      RSS      AIC
<none>                 0.093571 -33.588
+ obp      1 0.0178844 0.075687 -33.285
+ bbp      1 0.0137906 0.079781 -32.863
+ kp       1 0.0046117 0.088960 -31.992
+ daritu   1 0.0027178 0.090853 -31.824
> step.pred <- predict(r5.stp,test.dat)
> par(mfrow=c(2,1))
> plot(bb[1:prep,42], type="l", ylim=c(0,1),
+ xlab="", ylab="")
> par(new=T)
> pred2.data <- as.vector(c(rep(NA, testp), step.pred))
> plot(pred2.data, type="l", lty=c(1), lwd=c(2),
+ ylim=c(0,1), xlab="players", ylab="長打率", col="red",
+ main="2軍長打率予測")
> step.pred
      9      10      11      12      13      14      15      16      17
0.2944447 0.4865771 0.2788547 0.3115777 0.2573362 0.3174951 0.3305433 0.2847377 0.3217166
     18     19     20     21     22     23     24     25     26
0.3321940 0.2978978 0.3655237 0.2950918 0.3860042 0.2514739 0.2669416 0.3176471 0.2955334
     27     28     29     30     31     32     33     34     35
0.2641896 0.2678073 0.3490333 0.2440609 0.2402474 0.2164787 0.2432885 0.4064876 0.1856985
     36     37     38     39     40     41     42     43
0.2466390 0.2686638 0.2847377 0.2807553 0.2827740 0.2847377 0.1944550 0.2450069
> pred.data+pred2.data
 [1]      NA      NA      NA      NA      NA      NA      NA      NA      NA 0.5651107
[10] 0.7911821 0.5660477 0.5899626 0.5290535 0.6065247 0.6277776 0.5715950 0.6145148 0.6323673
[19] 0.5673020 0.6699924 0.5753221 0.6685652 0.5337313 0.5426332 0.6025658 0.5662228 0.5411500
[28] 0.5391401 0.6235712 0.5150420 0.5102462 0.4934801 0.5141514 0.6953828 0.4560201 0.5223853
[37] 0.5655947 0.5753384 0.5617732 0.5558249 0.5743249 0.4714463 0.5316721

```

2軍長打率予測



(4)

昨年四国アイランドリーグからプロ入りまたは2軍のみの球団に入団し、2軍試合にすでに出場している選手の8/1時点の成績と比較し評価する。

井上絢登(DeNA)

予測(10人目)：打率 0.238 三振率 0.263 四球率 0.0695 出塁率 0.305 長打率 0.430

今季：232 打席打率 0.291 三振率 0.121 四球率 0.0603 出塁率 0.359 長打率 0.456

増田将馬(くふうハヤテ)

予測(15人目)：打率 0.237 三振率 0.242 四球率 0.0712 出塁率 0.297 長打率 0.338

今季：237 打席打率 0.274 三振率 0.176 四球率 0.0773 出塁率 0.332 長打率 0.335

河野聡太（オリックス）

予測(18人目)：打率 0.235 三振率 0.241 四球率 0.073 出塁率 0.300 長打率 0.339

今季：88 打席打率 0.276 三振率 0.102 四球率 0.102 出塁率 0.356 長打率 0.316

打率に関しては予測が実際を 0.04 ほど下回り、三振率は 1/2 程になり、四球率は 0.03~0.006 ほど差があり、出塁率に関しては予測が実際を 0.05 ほど下回り、スケールを考慮するとかなり差があることからあまりいい予想ができていないといえる。長打率は誤差が 0.03 位内に収まっていて、比較的に確に予想ができていているといえる。

今回は切片を考慮しないモデルの予想をしなかったが、考慮することで精度がよくなる可能性がある。また、今回は純粋な打力の評価を考えるため盗塁に関するパラメータを除いたが、走塁に関係ない三振率以外は考慮すると精度が上がると考えられる。

また、NPB 球団 1 球団の選手だけではモデルの推定に使うデータがどうしても少なくなってしまうので、打席数を考慮せずモデルの推定に使うことで精度が上がるかなとも考えた。