NBA Draft picks

By Sample Student
Model Used: Zero Truncated Negative Binomial

Why this topic?

I love playing and watching basketball. I also like to see who gets drafted in the NBA draft. It is exciting to see the new prospects for the teams and I wanted to see if there was a way to predict draft picks accurately.

Data Set

The predictors used for this project include TDS (Top Division School), Position played, years played in college, number that they were drafted, and age when they were drafted.

tds	position	yrs	ndraft	age
no	G	0	3	19
no	C	0	15	18
yes	G	3	7	21
no	C	2	41	20
yes	F	1	2	20
yes	G	1	11	20
yes	F	1	3	19
yes	G	2	13	21
yes	F	4	27	22
yes	G	2	3	20
yes	C	1	1	19
yes	F	2	12	20
yes	G	3	29	21
yes	F	1	1	19
yes	G	1	17	19
yes	F	2	1	20
yes	C	1	22	19
no	G	2	3	20
yes	G	1	3	19
no	C	2	5	20
no	C	3	4	21
no	C	2	14	20
no	C	3	9	21
no	F	1	7	19
yes	G	4	10	22
yes	G	4	33	22
no	C	2	16	20
yes	G	1	7	19
no	F	1	1	19
yes	C	2	31	20
no	F	2	29	20
no	C	1	32	19
no	G	1	3	19
yes	F	2	23	20

Sas and R codes

```r
draft.data<- suppresswarnings(read.csv(file="C:/Users/eddie/OneDrive/Documents/R
                    sep = ",", skipNul = T))
library(VGAM)

draft.data$tds<- factor(draft.data$tds)
draft.data$pos<- factor(draft.data$pos)

#specifying reference level
tds.rel<- relevel(draft.data$tds, ref="yes")
pos.rel<- relevel(draft.data$pos, ref="C")

#fitting zero-truncated negative binomial model
library(VGAM)
summary(fitted.model<- vglm(draftnumber ~ pos + yrs + age + tds.rel,
                            data=draft.data, family=posnegbinomial()))

#checking model fit
null.model<- vglm(draftnumber ~ 1, data=draft.data, family=posnegbinomial())
print(deviance<- -2*(logLik(null.model)-logLik(fitted.model)))
print(p.value<- pchisq(deviance, df=3, lower.tail = FALSE))

#using fitted model for prediction
print(predict(fitted.model, data.frame(pos="G", yrs=3, age = 21,
                                       tds.rel="yes"), type="response"))
```

```
;
proc format;
 value $positionfmt "C"="Cref_position";
 value $tds "no"="ref";
 run;
 /*fitting zero-truncated negative binomial model*/
proc fmm;
  class position tds;
  model ndraft = tds position yrs age / dist=truncnegbin;
 run;

 /*checking model fit*/
proc fmm;
  model ndraft = / dist=truncnegbin;
 run;
```

R Outputs

```
null.model<- vglm(draftnumber ~ 1, data=draft.data, family=posnegbinomial())
print(deviance<- -2*(logLik(null.model)-logLik(fitted.model)))
1] 13.13612
print(p.value<- pchisq(deviance, df=3, lower.tail = FALSE))
1] 0.004351303

#using fitted model for prediction
print(predict(fitted.model, data.frame(pos="G", yrs=2, age = 20,
                                       tds.rel="yes"), type="response"))
    [,1]
9.655961
```

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept):1  1.74271    9.64098   0.181   0.8566
(Intercept):2  0.33902    0.29567   1.147   0.2515
posF          -0.65559    0.34933  -1.877   0.0606 .
posG          -0.63042    0.33174  -1.900   0.0574 .
yrs            0.43329    0.51198   0.846   0.3974
age            0.01137    0.53103   0.021   0.9829
tds.relno      0.03144    0.29637   0.106   0.9155
---
```

SaS Outputs

Parameter Estimates for Truncated Negative Binomial Model

Effect	tds	position	Estimate	Standard Error	z Value	Pr > |z|
Intercept			1.1121	9.8228	0.11	0.9099
tds	no		0.03144	0.3234	0.10	0.9226
tds	yes		0	.	.	.
position		C	0.6304	0.3674	1.72	0.0861
position		F	-0.02514	0.3501	-0.07	0.9427
position		G	0	.	.	.
yrs			0.4333	0.5204	0.83	0.4050
age			0.01137	0.5363	0.02	0.9831
Scale Parameter			0.7125	0.2190		

Iteration History

Iteration	Evaluations	Objective Function	Change	Max Gradient
0	5	183.65975988	.	7.793035
1	4	180.22897314	3.43078674	9.839196
2	2	178.01927468	2.20969846	2.829623
3	2	177.62775089	0.39152379	0.933919
4	2	177.56040942	0.06734147	0.301951
5	6	-292.9115503	470.47195972	3.496E17
6	100	-490.2217258	197.31017554	3.441E17
7	24	-554.2217253	63.99999944	3.441E17

Optimization routine cannot improve the function value.

Fitted Model

$$\lambda = \exp\{1.7434 + 0.03144*(\text{“no”}) - 0.65559*(\text{position} = F) - 0.6304*(\text{position} = G) + 0.4333*\text{yrs} + 0.01137*\text{age}\}$$

$$r = 1 / 0.7125 = 1.386001386$$

Analysis

The only significant predictor for this data set would be for positions played, specifically G or Guards position.

For this model though, since it is a very complex relation between the response and predictors, we omit interpretation entirely. I would like to say though, that due to the playstyle in the league currently, guards have been more sought out than ever before since the league has become very guard oriented.

Fitted Model Prediction Response

By hand:

$\lambda = \exp\{1.7434 - 0.6304*(\text{position} = G) + 0.4333*2 + 0.01137*20\} = 9.08841$

With R:

```
#using fitted model for prediction
print(predict(fitted.model, data.frame(pos="G", yrs=2, age = 20,
                                       tds.rel="yes"), type="response"))
    [,1]
9.655961
```

Franz Wagner

Round 1

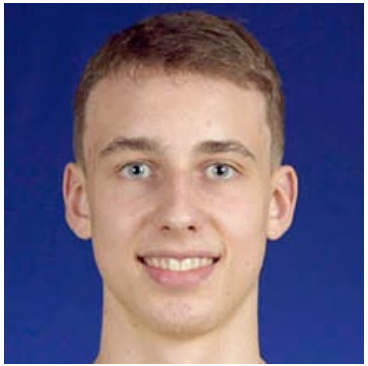
Picked 8

Michigan State

Plays Shooting Guard

Drafted at age 20

Played 2 years at Michigan



Thank you Dr. Olga and thank you to my fellow classmates. Good luck on the rest of your classes! Stay safe these holidays!