Applying Bayesian Statistics in Predicting NBA Basketball Prospect Outcomes Using Aggregate Scouting Information

Ted Henson

8/8/2020

Intro

There are an infinite number of causes or variables that could potentially predict NBA success if one had access to the information: work ethic, receptiveness to coaching, wingspan, vertical leap, top speed, collegiate or international basketball success. Some of these variables such as collegiate success (measured by statistics) are available to the public. Some such as vertical leap may be available to a team after a workout or NBA combine. This analysis is not meant to create the best possible predictions for a prospect as that would require a lot of time and resources. The goal is to apply Bayesian techniques in a basketball setting and to see how much influence a player's high school recruiting ranking plays in NBA success.

Data

The input data is aggregate scouting information from the Recruiting Services Consensus Index (RSCI). The data includes rankings from a variety of recruiting services for the top 100 high school prospects going back to 1998. Rivals is the only service that has been constant through the years. ESPN has been in the index since 2013, but their initial creator, Dave Telep, has been ranking prospects since 1998 so they are treated synonymous for this project. Other services that are still active include 247Sports and Hot100Hoops. All other services in the index no longer rank prospects. Regressions run with the composite ratings and the other active services produced posterior predictor distributions that were collinear. As a result only the composite rating was included in the final regression. It was also standardized to more easily set the priors and interpret the posterior distribution. Also included in the the predictor matrix was a player's age and position. The response data is the advanced metrics table from basketball reference. This table includes

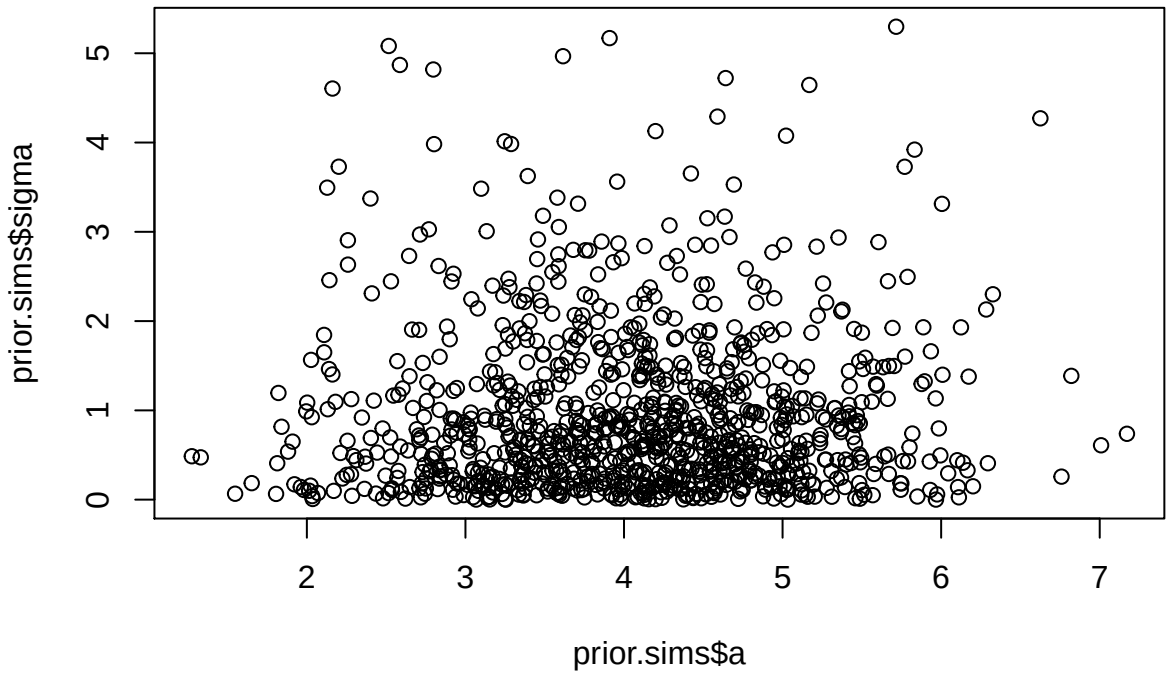
statistics such as Value Over Replacement Player (VORP), Win Shares (WS), and many others. This analysis will attempt to first predict Win Shares Per 48 Minutes (WS/48) for a player's rookie season. Rookie seasons were chosen to simplify the analysis. WS/48 was chosen as opposed to VORP as VORP penalized rookies who played a lot of minutes on bad teams. In order to control for outliers who played very few minutes, only rookies who played over 100 minutes were considered. If a player did not play over 100 minutes they either were injured, were not given a chance, or were ineffective when they played. In practice one would not exclude them and either include them, regress their stats towards the mean or minimum, or something else, but for simplicity they are excluded here. The definition of the model and priors will be shown below.

Analysis

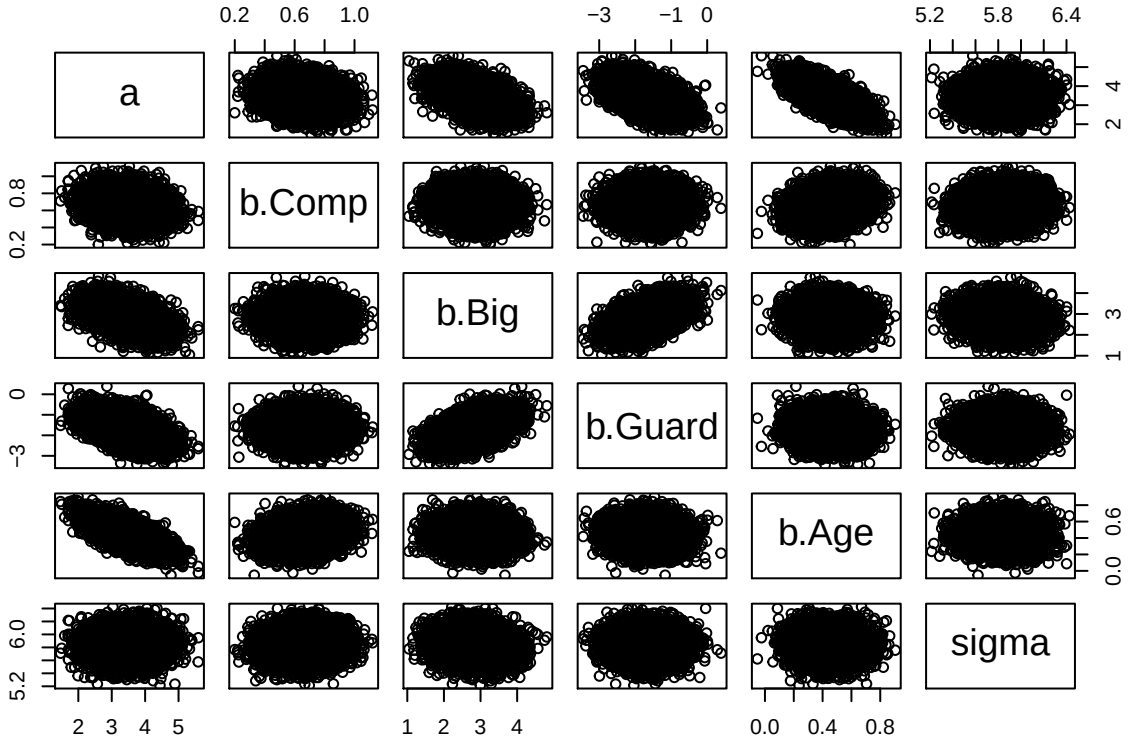
Definition of Model and Priors

```
##
## Quadratic approximate posterior distribution
##
## Formula:
## y ~ dnorm(mu, sigma)
## mu <- a + b.Comp * Composite.Rating + b.Big * Is.Big + b.Guard *
##     Is.Guard + b.Age * Age
## a ~ dnorm(4, 1)
## b.Comp ~ dlnorm(0, 0.25)
## b.Big ~ dnorm(0, 1)
## b.Guard ~ dnorm(0, 1)
## b.Age ~ dnorm(2, 1)
## sigma ~ dexp(1)
##
## Posterior means:
##          a     b.Comp      b.Big    b.Guard      b.Age      sigma
##  3.4846877  0.6665989  2.8238724 -1.6719284  0.4405826  5.8413733
##
## Log-likelihood: -1976.93
```

Plots of Simulation of Priors



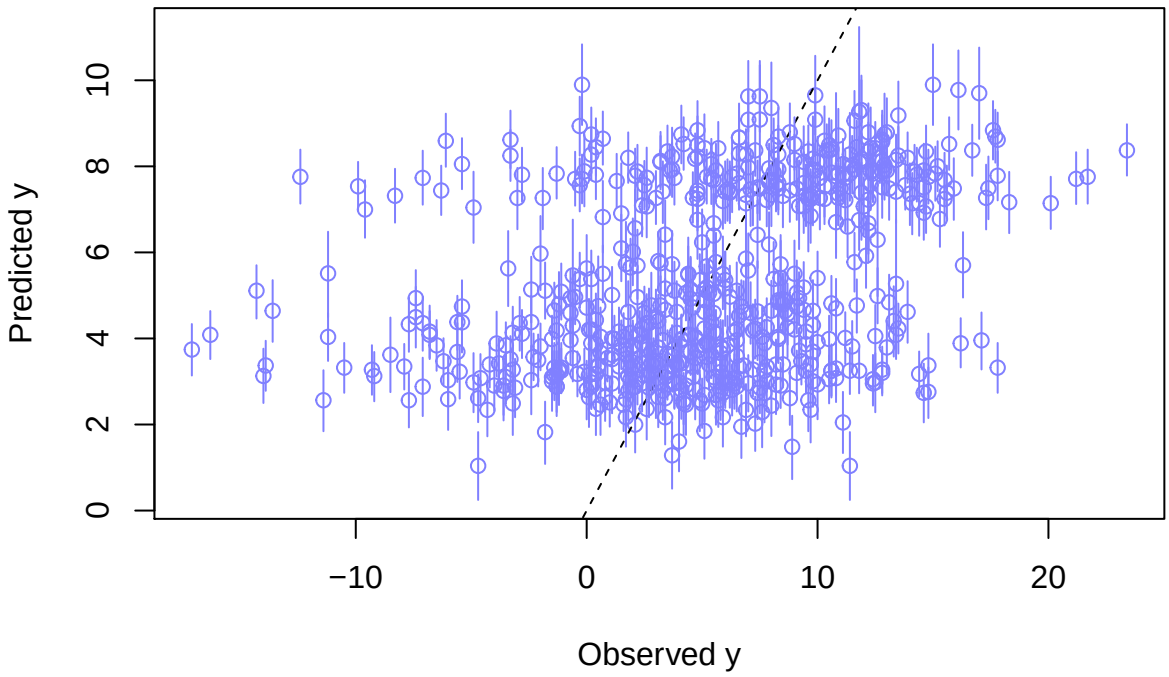
Plots of Simulated Samples from Posterior



Summary of Model Posterior

```
##                mean         sd       5.5%      94.5%
## a         3.4846877 0.5368172  2.6267502  4.3426253
## b.Comp    0.6665989 0.1262544  0.4648200  0.8683778
## b.Big     2.8238724 0.5062742  2.0147484  3.6329965
## b.Guard -1.6719284 0.4895676 -2.4543520 -0.8895049
## b.Age     0.4405826 0.1174662  0.2528488  0.6283163
## sigma     5.8413733 0.1654938  5.5768822  6.1058643
```

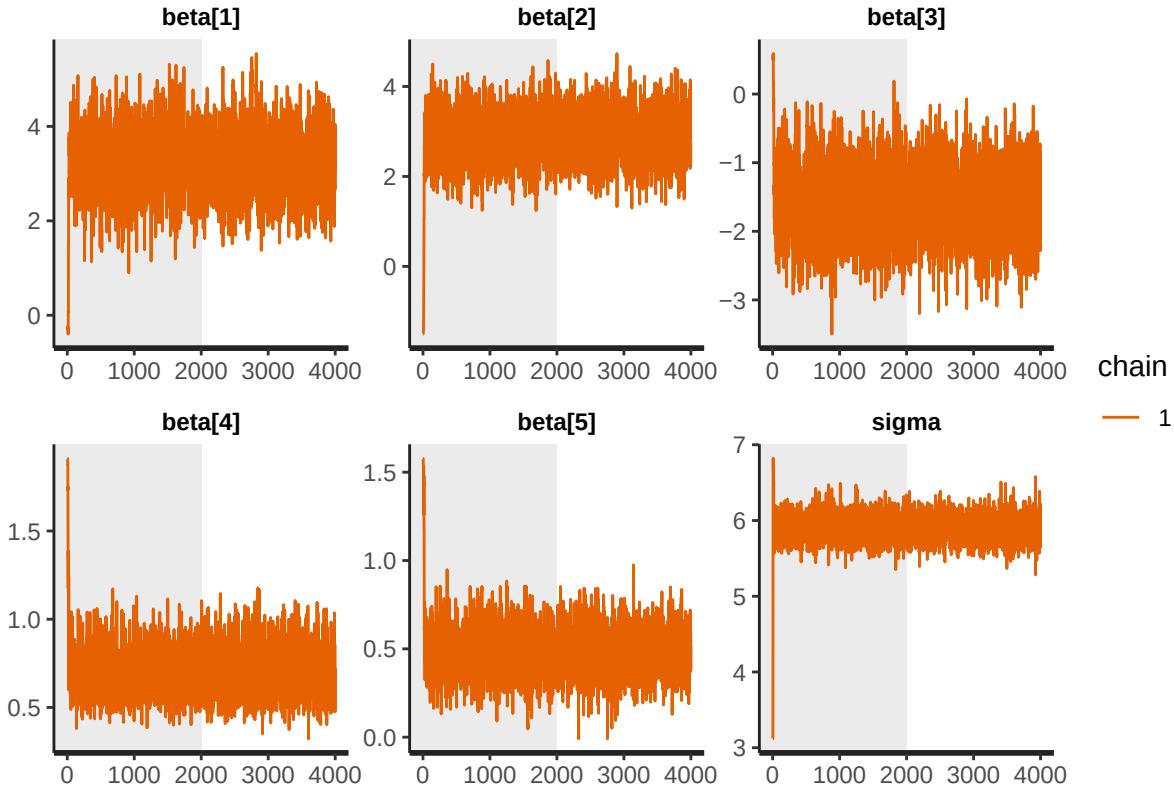
Plot of Observed versus Expected WS/48 with Predictive Intervals



Markov Chain Monte Carlo

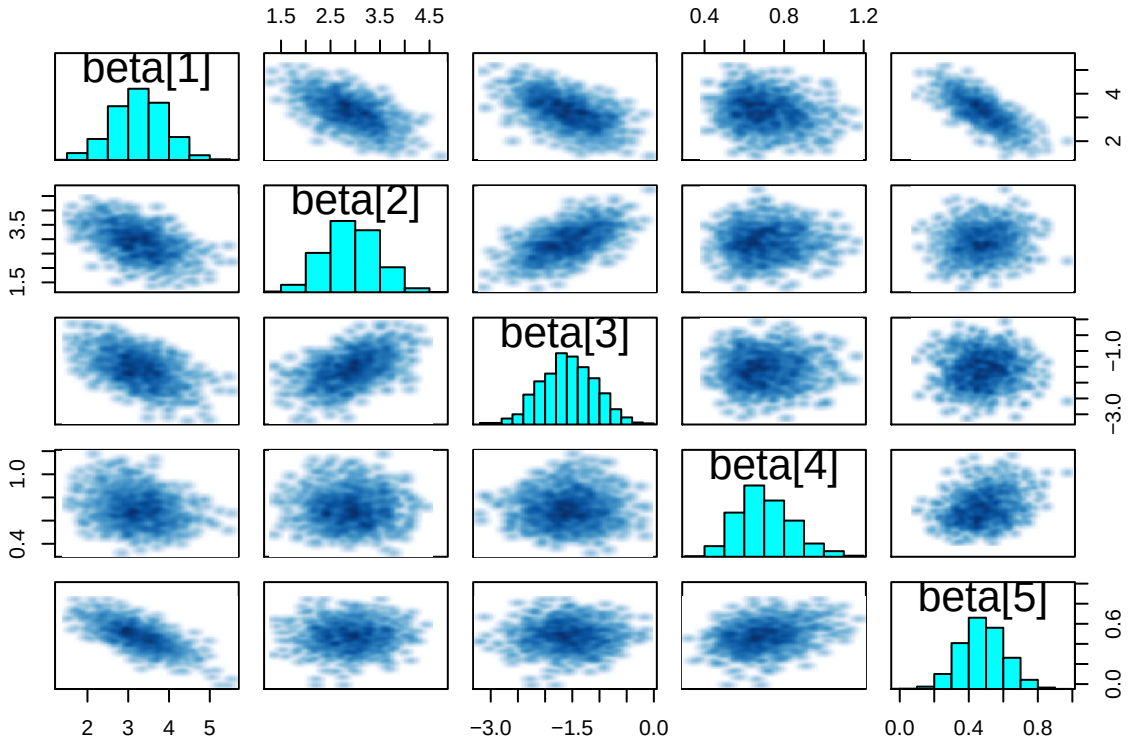
The same model definitions above were applied, but Markov Chain Monte Carlo methods were applied to estimate the posterior distribution for the parameters. The process was run with 4000 iterations per chain including warmup iterations. The model was initially ran with 4 chains to check the posterior estimations across

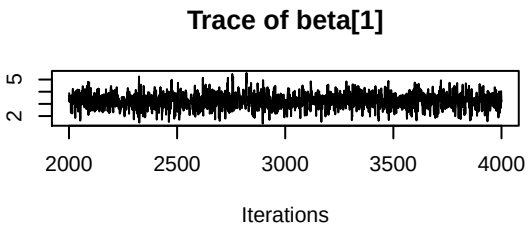
the chains, but since they all converged to the same values one chain was used for the final model. Below beta[1] is the mean of the response, beta[2] for Big, beta[3] for Guard, beta[4] for composite rating, and beta[5] for age.

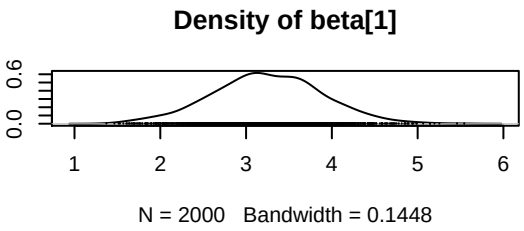


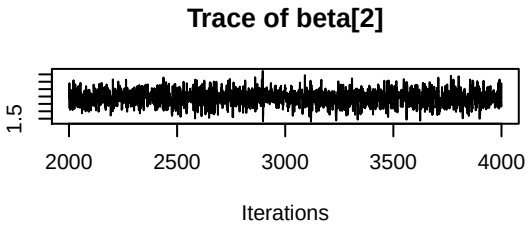

```
## Inference for Stan model: c4eed859a77bcf0d2b9709cd93ee41c2.
## 1 chains, each with iter=4000; warmup=2000; thin=1;
## post-warmup draws per chain=2000, total post-warmup draws=2000.
##
##          mean se_mean   sd  2.5%   50% 97.5% n_eff Rhat
## beta[1]  3.25    0.02 0.64  1.97  3.23  4.49  1131    1
## beta[2]  2.91    0.01 0.52  1.86  2.90  3.93  1573    1
## beta[3] -1.58    0.01 0.50 -2.56 -1.58 -0.62  1485    1
## beta[4]  0.70    0.00 0.14  0.47  0.69  1.01  1561    1
## beta[5]  0.48    0.00 0.13  0.24  0.48  0.73  1500    1
##
## Samples were drawn using NUTS(diag_e) at Wed Aug 19 19:59:46 2020.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
```

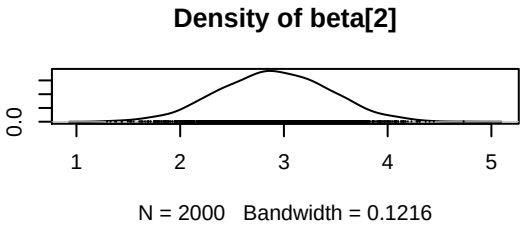
```
## convergence, Rhat=1).
```

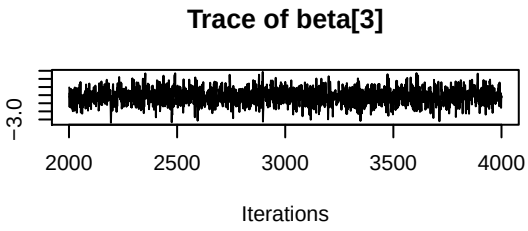


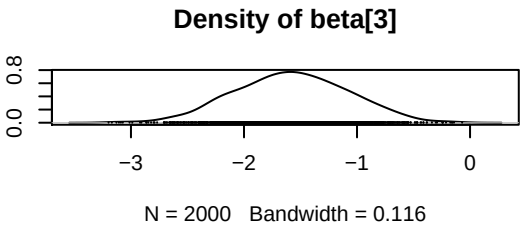

Trace of beta[1]
2 5
2000 2500 3000 3500 4000
Iterations

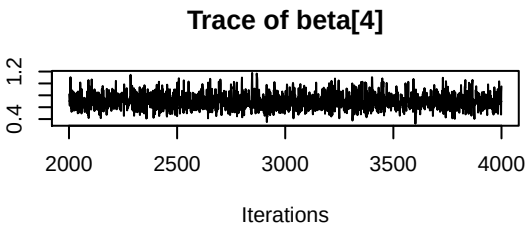

Density of beta[1]
0.0 0.6
1 2 3 4 5 6
N = 2000 Bandwidth = 0.1448

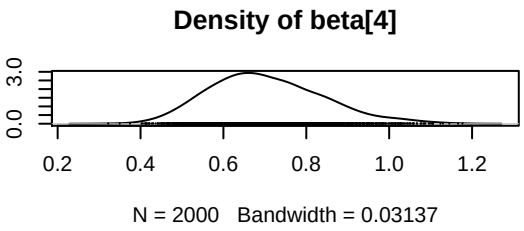

Trace of beta[2]
1.5
2000 2500 3000 3500 4000
Iterations

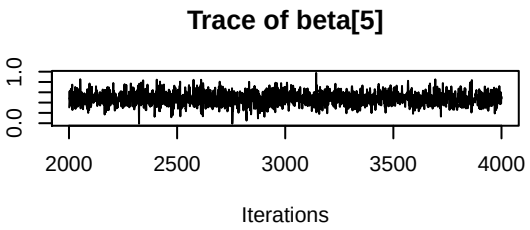

Density of beta[2]
0.0
1 2 3 4 5
N = 2000 Bandwidth = 0.1216

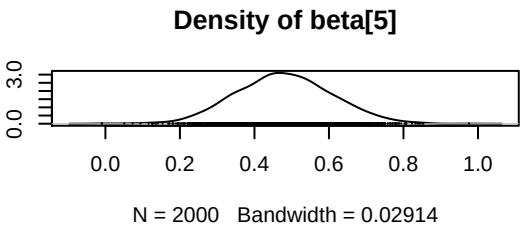

Trace of beta[3]
−3.0
2000 2500 3000 3500 4000
Iterations


Density of beta[3]
0.0 0.8
−3 −2 −1 0
N = 2000 Bandwidth = 0.116


Trace of beta[4]
0.4 1.2
2000 2500 3000 3500 4000
Iterations


Density of beta[4]
0.0 3.0
0.2 0.4 0.6 0.8 1.0 1.2
N = 2000 Bandwidth = 0.03137


Trace of beta[5]
0.0 1.0
2000 2500 3000 3500 4000
Iterations


Density of beta[5]
0.0 3.0
0.0 0.2 0.4 0.6 0.8 1.0
N = 2000 Bandwidth = 0.02914