Lab 5 Solutions

Enter Your Name and UNI Here

November 18, 2016

In today's lab we will use the Bet distribution to explore the probability of reaching a base safely in baseball. The Beta is a random variable bounded between 0 and 1 and often used to model the distribution of proportions. The probability distribution function for the Beta with parameters α and β is

$$p(x|\alpha, \beta) = \frac{\Gamma(\alpha + \beta)}{\Gamma(\alpha) + \Gamma(\beta)} x^{\alpha-1}(1-x)^{\beta-1}$$

where $\Gamma()$ is the Gamma function, the generalized version of the factorial. Thankfully, for this assignment, you need not know what the Gamma function is; you need only know that the mean of a Beta is $\frac{\alpha}{\alpha+\beta}$ and its variance is $\frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$.

For this assignment you will test the fit of the Beta distribution to the on-base percentages (OBPs) of hitters in the 2014 Major League Baseball season; each plate appearance (PA) results in the batter reaching base or not, and this measure is the fraction of successful attempts. This set has been pre-processed to remove those players with an insufficient number of opportunities for success.

Part I

1. Load the file `baseball.csv` into a variable of your choice in R. How many players have been included? What is the minimum number of plate appearances required to appear on this list? Who had the most plate appearances? What are the minimum, maximum, and mean OBP?

```
bb <- read.csv("baseball.csv", as.is = TRUE)
head(bb)
```

```
##               Name  PA   OBP
## 1       Mike Trout 705 0.377
## 2 Andrew McCutchen 648 0.410
## 3 Michael Brantley 676 0.385
## 4   Anthony Rendon 683 0.351
## 5      Alex Gordon 643 0.351
## 6   Josh Donaldson 695 0.342
```

```
num.players <- nrow(bb)
num.players
```

```
## [1] 441
```

```
min.PA <- min(bb$PA)
min.PA
```

```
## [1] 103
```

```r
bb$Name[which.max(bb$PA)]
```

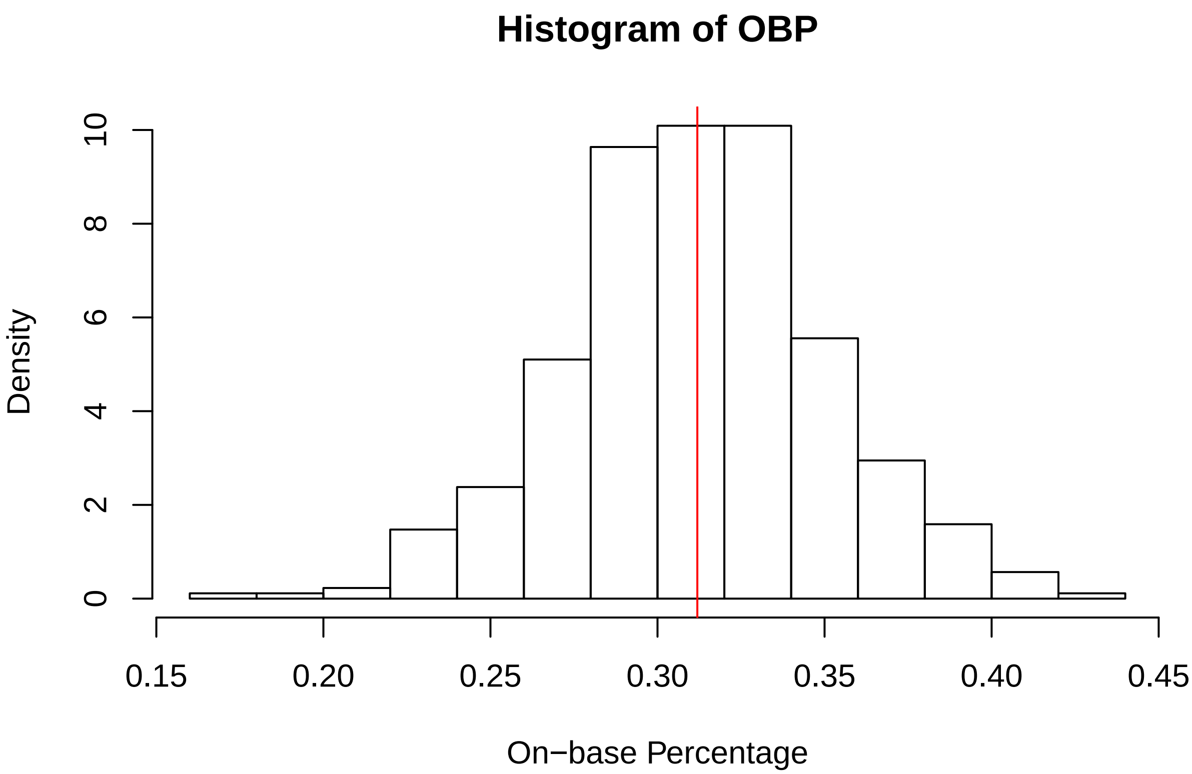
```
## [1] "Ian Kinsler"
```

```r
summary(bb$OBP)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##  0.1680  0.2870  0.3130  0.3119  0.3370  0.4320
```

2. Plot the OBP data as a histogram with the option `probability=TRUE`. Add a vertical line for the mean of the distribution. Does the mean coincide with the mode of the distribution?

```r
hist(bb$OBP, probability = TRUE, xlab = "On-base Percentage", main = "Histogram of OBP")
abline(v = mean(bb$OBP), col = "red")
```



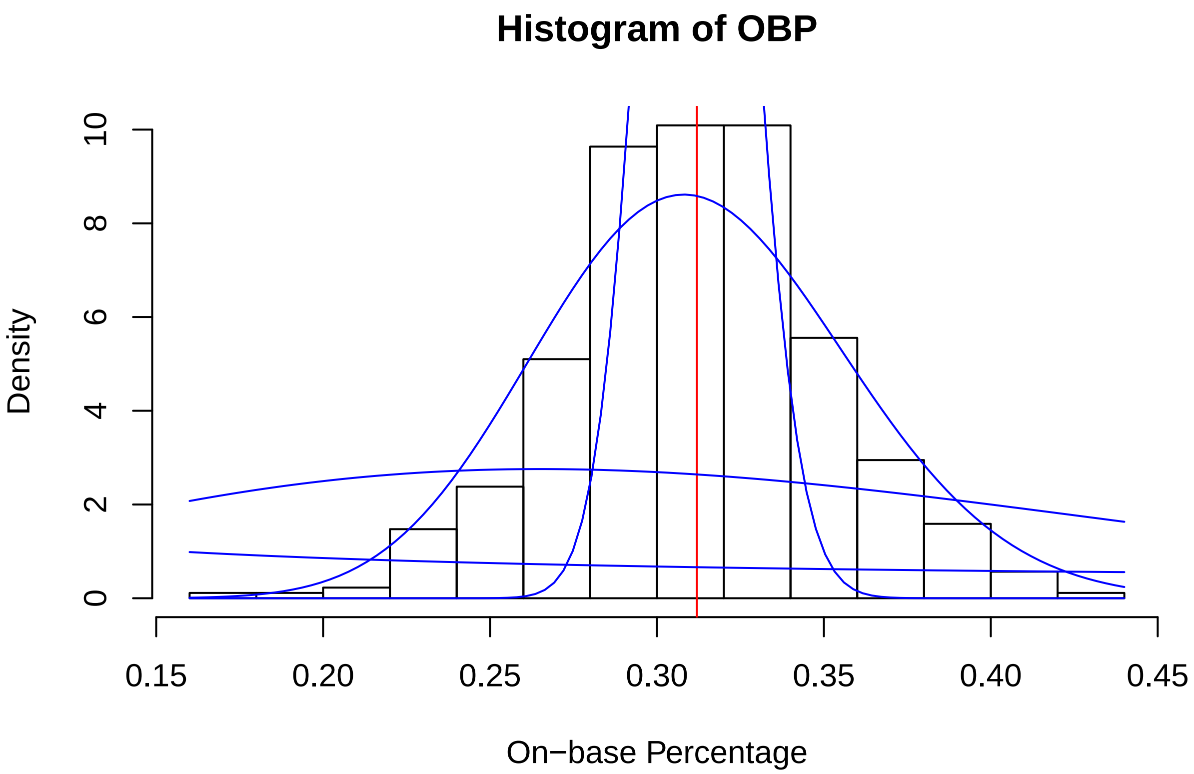
3. Eyeball fit. Add a `curve()` to the plot using the density function `dbeta()`. Pick parameters α and β that match the mean of the distribution but where their sum equals 1. Add three more `curve()`s to this plot where the sum of these parameters equals 10, 100 and 1000 respectively. Which of these is closest to the observed distribution?

```r
this.mean <- mean(bb$OBP)

params <- function(this.mean, sum) {
  a <- this.mean*sum
  return(c(alpha = a, beta = sum - a))
}
```

```
hist(bb$OBP, probability = TRUE, xlab = "On-base Percentage", main = "Histogram of OBP")
abline(v = mean(bb$OBP), col = "red")

curve(dbeta(x, shape1 = params(this.mean, sum = 1)[1], shape2 = params(this.mean, sum = 1)[2]), add = TR
curve(dbeta(x, shape1 = params(this.mean, sum = 10)[1], shape2 = params(this.mean, sum = 10)[2]), add = 
curve(dbeta(x, shape1 = params(this.mean, sum = 100)[1], shape2 = params(this.mean, sum = 100)[2]), add 
curve(dbeta(x, shape1 = params(this.mean, sum = 1000)[1], shape2 = params(this.mean, sum = 1000)[2]), ad
```

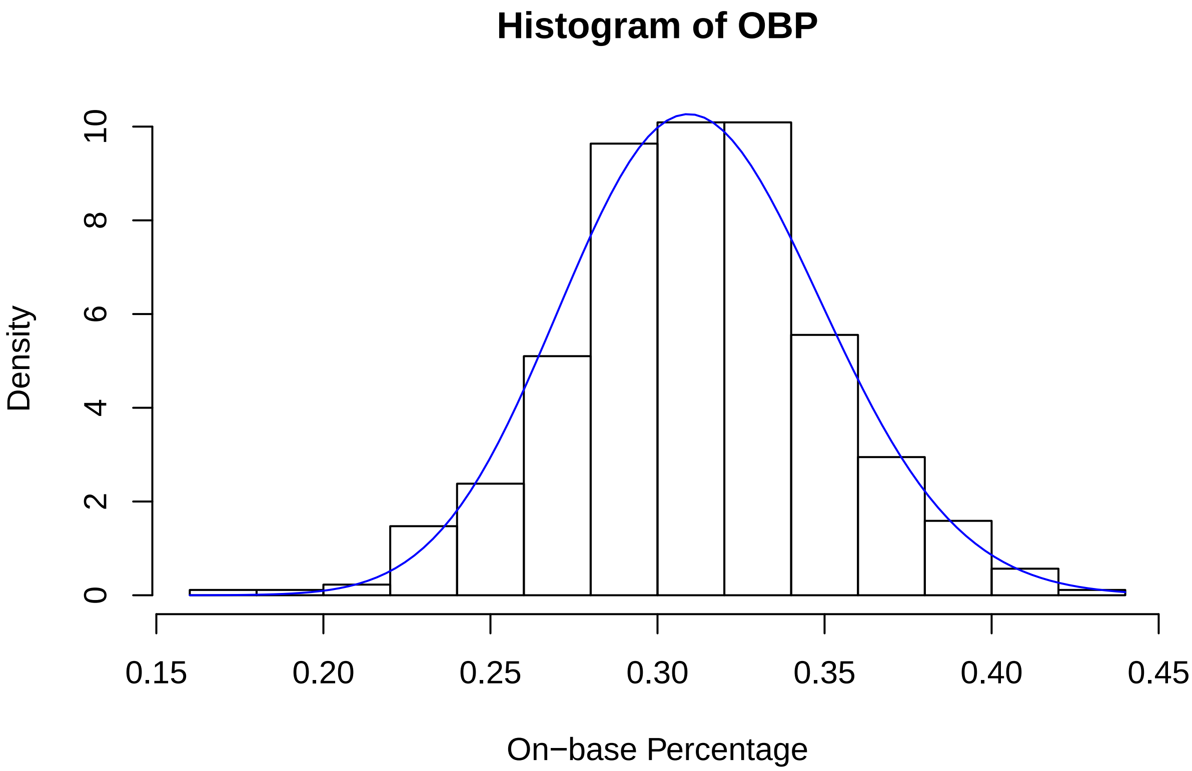


Part II

4. Method of moments fit. Find the calculation for the parameters from the mean and variance and solve for α and β. Create a new density histogram and add this `curve()` to the plot. How does it agree with the data?

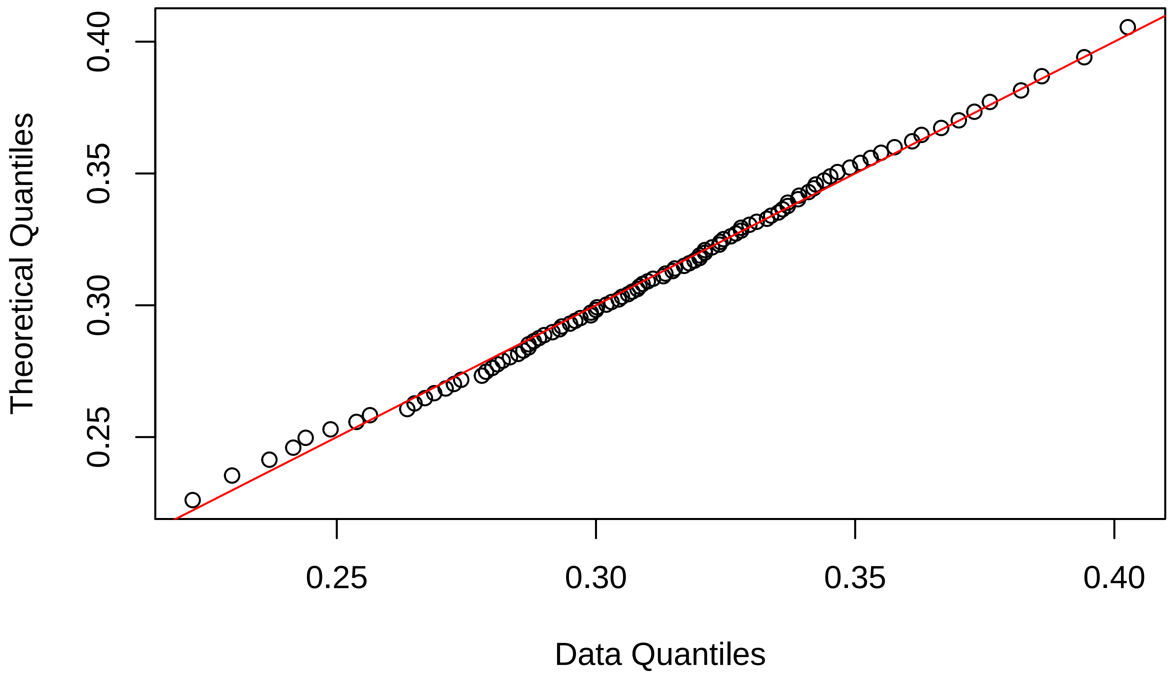
```
beta.MMest <- function(data) {
  m <- mean(data)
  v <- var(data)
  return(c(alpha = (m/v)*(m*(1-m) - v), beta = ((1-m)/v)*(m*(1-m) - v)))
}
MMest <- beta.MMest(bb$OBP)


hist(bb$OBP, probability = TRUE, xlab = "On-base Percentage", main = "Histogram of OBP")
curve(dbeta(x, shape1 = MMest[1], shape2 = MMest[2]), add = TRUE, col = "blue")
```



5. Calibration. Find the 99 percentiles of the actual distribution of the data using the `quantile()` function using `quantile(bb$OBP, probs = seq(1, 99)/100)` and plot them against the 99 percentiles of the beta distribution you just fit using `qbeta()`. How does the fit appear to you?

```
data.quant <- quantile(bb$OBP, probs = seq(1, 99)/100)
theory.quant <- qbeta(seq(1, 99)/100, shape1 = MMest[1], shape2 = MMest[2])
plot(data.quant, theory.quant, xlab = "Data Quantiles", ylab = "Theoretical Quantiles")
abline(a = 0, b = 1, col = "red")
```



6. Optional if you have time – MLE fit. Create a function for the log-likelihood of the distribution that calculates `-sum(dbeta(your.data.here, your.alpha, your.beta, log = TRUE))` and has one argument `params = c(your.alpha, your.beta)`. Use `nlm()` to find the minimum of the negative of the log-likelihood. Take the MOM fit for your starting position. How do these values compare?

```
logLik <- function(params) {
  return(-sum(dbeta(bb$OBP, params[1], params[2], log = TRUE)))
}

MLEest <- nlm(logLik, MMest)$est
MLEest
```

```
## [1] 43.73915 96.49892
```

```
MMest
```

```
##    alpha     beta
## 44.31690 97.76163
```