Lab 7

Christine Chong cc4190

June 21, 2017

In today's lab we will use the Beta 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?

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

```
##               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
```

```
nrow(baseball)
```

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

```
min(baseball$PA)
```

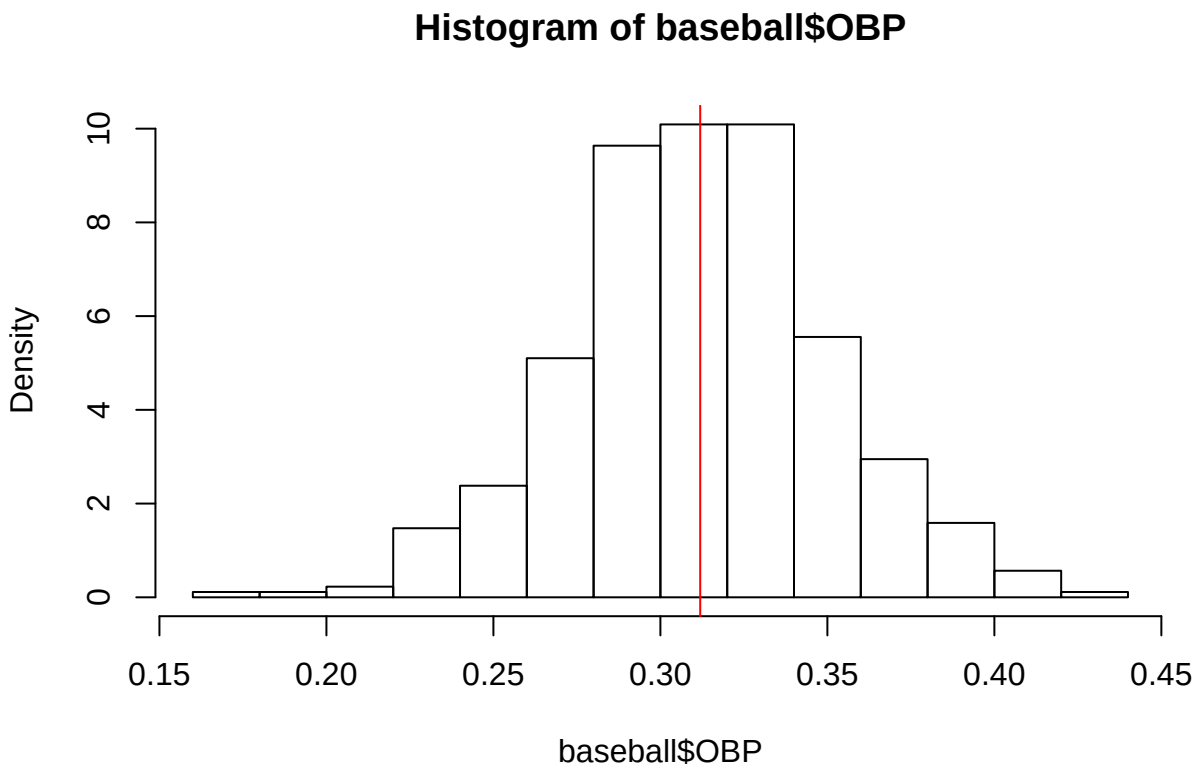
```
## [1] 103
```

```
max(baseball$PA)
```

```
## [1] 726
```

The number of players is 441. The minimum plate appearances is 103. The maximum plate appearances is 727. 2. Plot the 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?

```
hist(baseball$OBP, probability = TRUE)
this.mean <- mean(baseball$OBP)
abline(v=this.mean, col ="red")
```




```
getmode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}
getmode(baseball$OBP)
```

```
## [1] 0.321
```

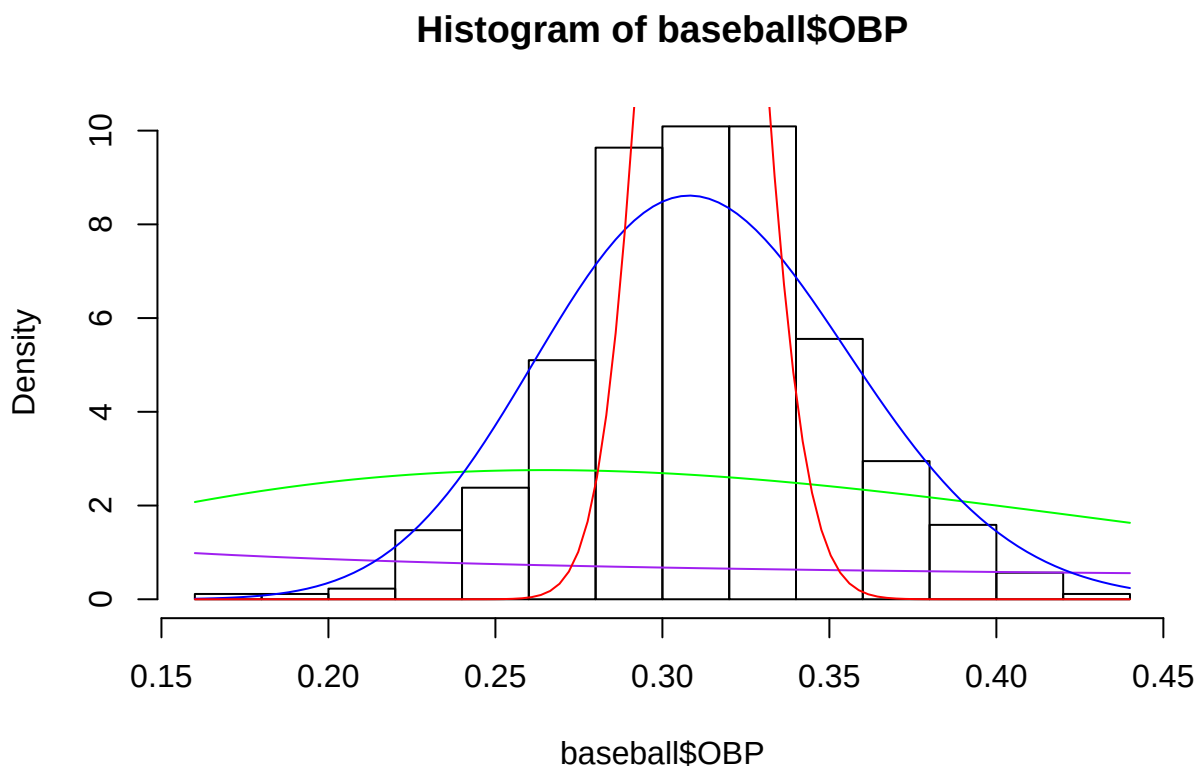
```
mean(baseball$OBP)
```

```
## [1] 0.3119184
```

The mean does seem to closely coincide with the mode 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?

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

```
hist(baseball$OBP, probability = TRUE)
curve(dbeta(x,shape1=params(this.mean, sum=1)[1],shape2=params(this.mean,sum=1)[2]), add=TRUE, col = "pu
curve(dbeta(x,shape1=params(this.mean, sum=10)[1],shape2=params(this.mean,sum=10)[2]), add=TRUE, col =
curve(dbeta(x,shape1=params(this.mean, sum=100)[1],shape2=params(this.mean,sum=100)[2]), add=TRUE, col =
curve(dbeta(x,shape1=params(this.mean, sum=1000)[1],shape2=params(this.mean,sum=1000)[2]), add=TRUE, col
```



The closest to the observed distribution seems to be the blue line. (Sum = 100). Maybe using a sum that is higher would help

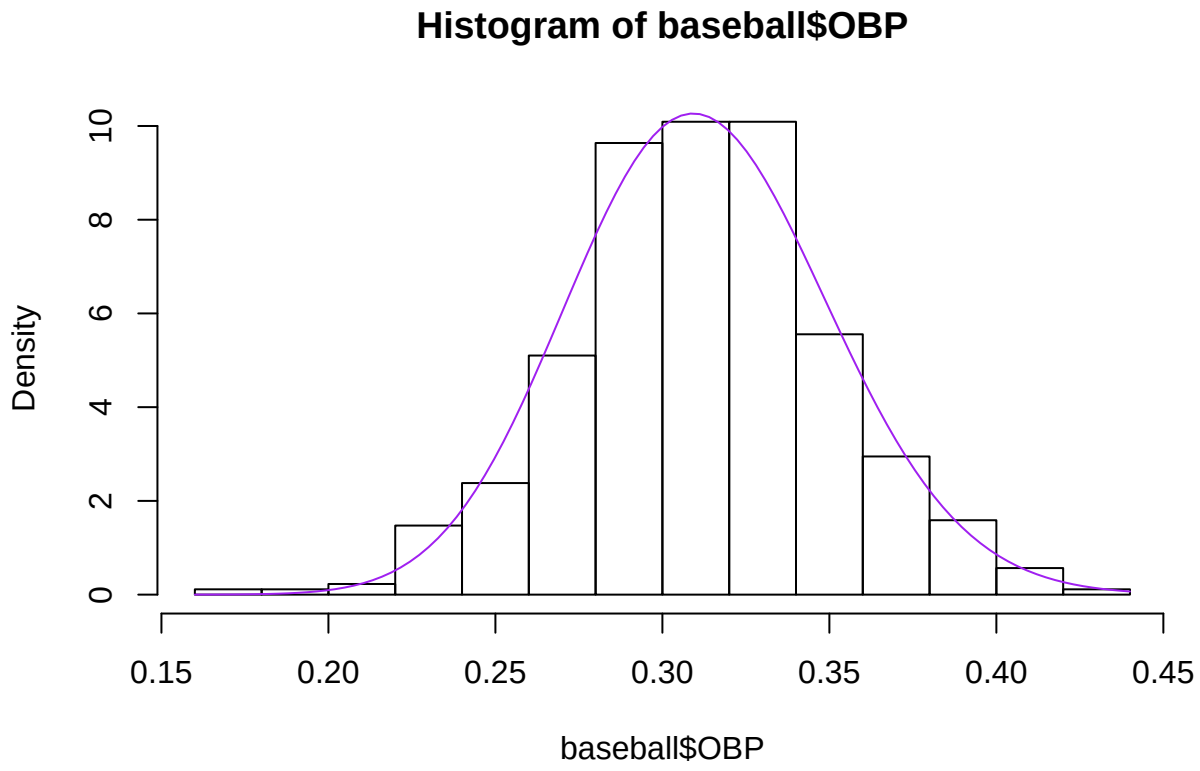
Part II

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)
  shape2<-(m*(1-m)^2-v+m*v)/v
  shape1<-(m/(1-m))*shape2
  return (c(alpha=shape1, beta=shape2))
}
MMest <- beta.MMest(baseball$OBP)
MMest
```

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

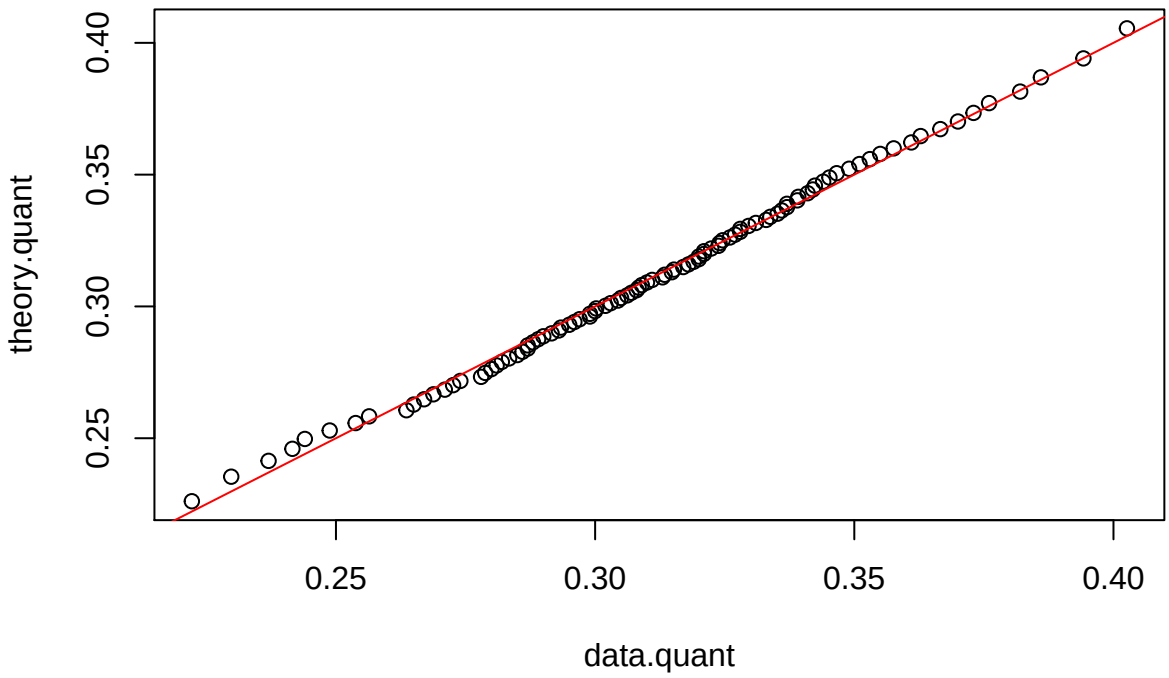
```
hist(baseball$OBP, probability = TRUE)
curve(dbeta(x,shape1= MMest[1],shape2=MMest[2]), add=TRUE, col = "purple")
```



The curve seems to agree well with the histrogram. This should mean that the alpha and beta estimates are pretty spot on.

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

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



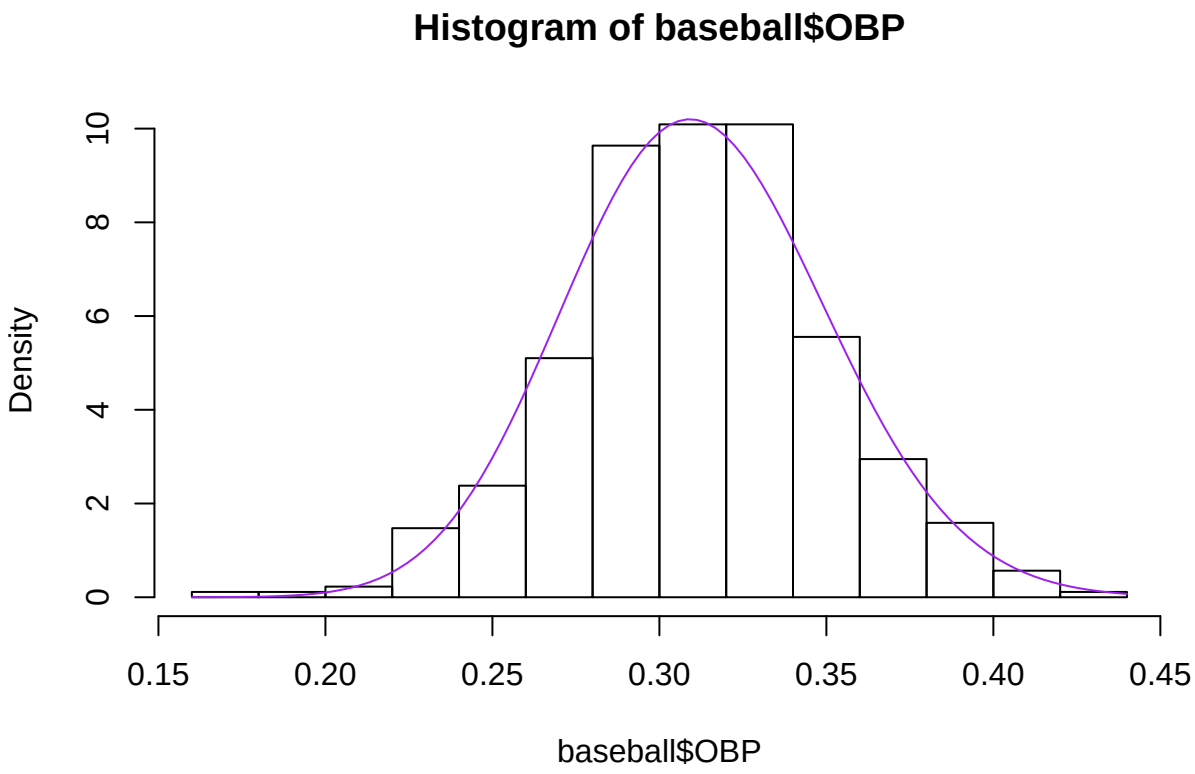
The fit looks good! Every point seems to be close to or on the AB line.

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 Method of Moments fit for your starting position. How do these values compare?

```
neglog<- function(theta, data=baseball$OBP){
  return(-sum(dbeta(data, shape1=theta[1], shape2=theta[2],log=TRUE)))
}
MLEest<- nlm(neglog, MMest,data = baseball$OBP )$estimate
MLEest
```

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

```
hist(baseball$OBP, probability = TRUE)
curve(dbeta(x,shape1= MLEest[1],shape2=MLEest[2]), add=TRUE, col = "purple")
```



The values of the MLE and MoM are pretty close to each other. For this case I don't know if there is something that is necessarily better. The curves on the histrograms seem very similar