Lab 5

Autumn Li and UNI ql2280

November 12, 2016

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("~/Desktop/baseball.csv")
player_num = length(baseball$Name);player_num
```

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

```
plate = baseball$PA
min(plate)
```

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

```
max(plate)
```

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

```
min(baseball$OBP)
```

```
## [1] 0.168
```

```
max(baseball$OBP)
```

```
## [1] 0.432
```

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

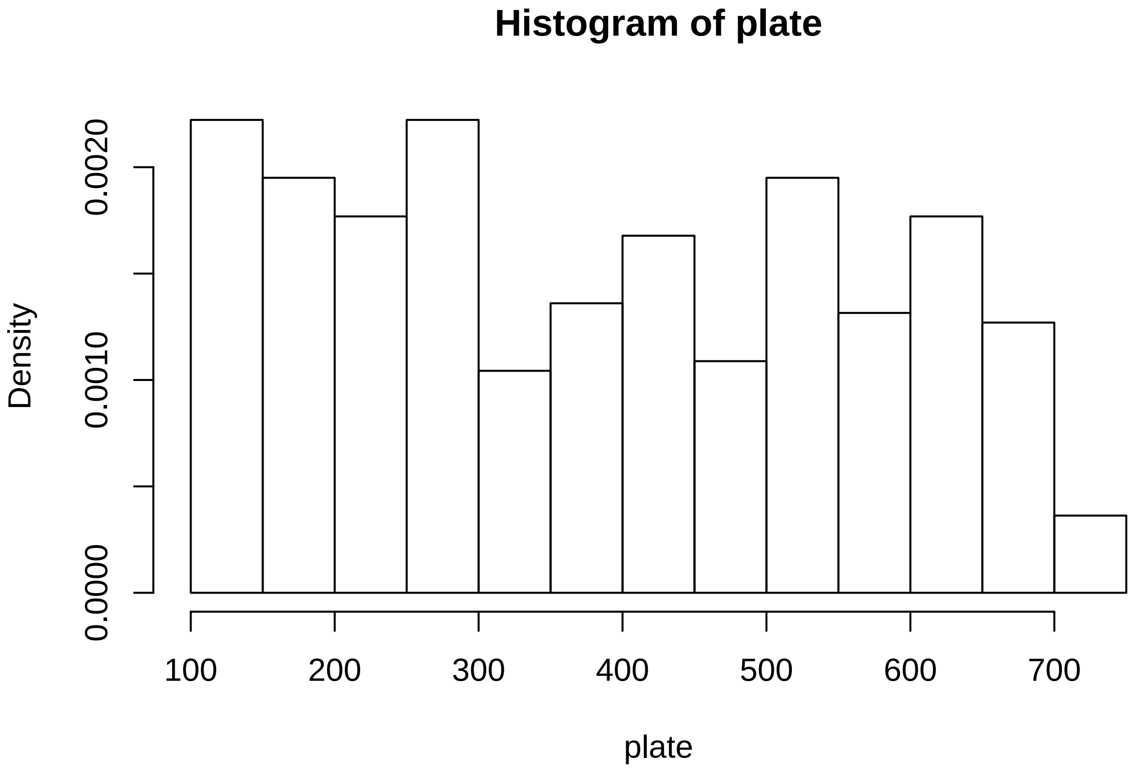
```
## [1] 0.3119184
```

```r
sd(baseball$OBP)
```

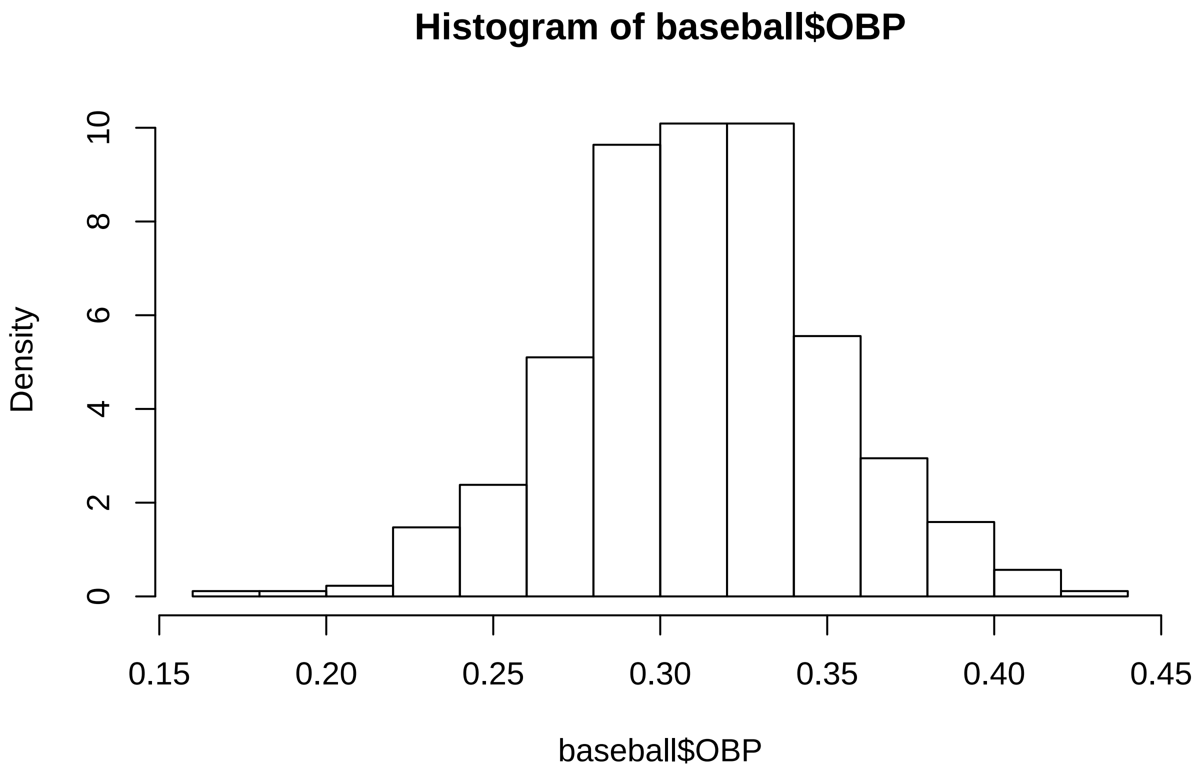
```
## [1] 0.03873051
```

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?

```r
hist(plate,probability=TRUE)
```



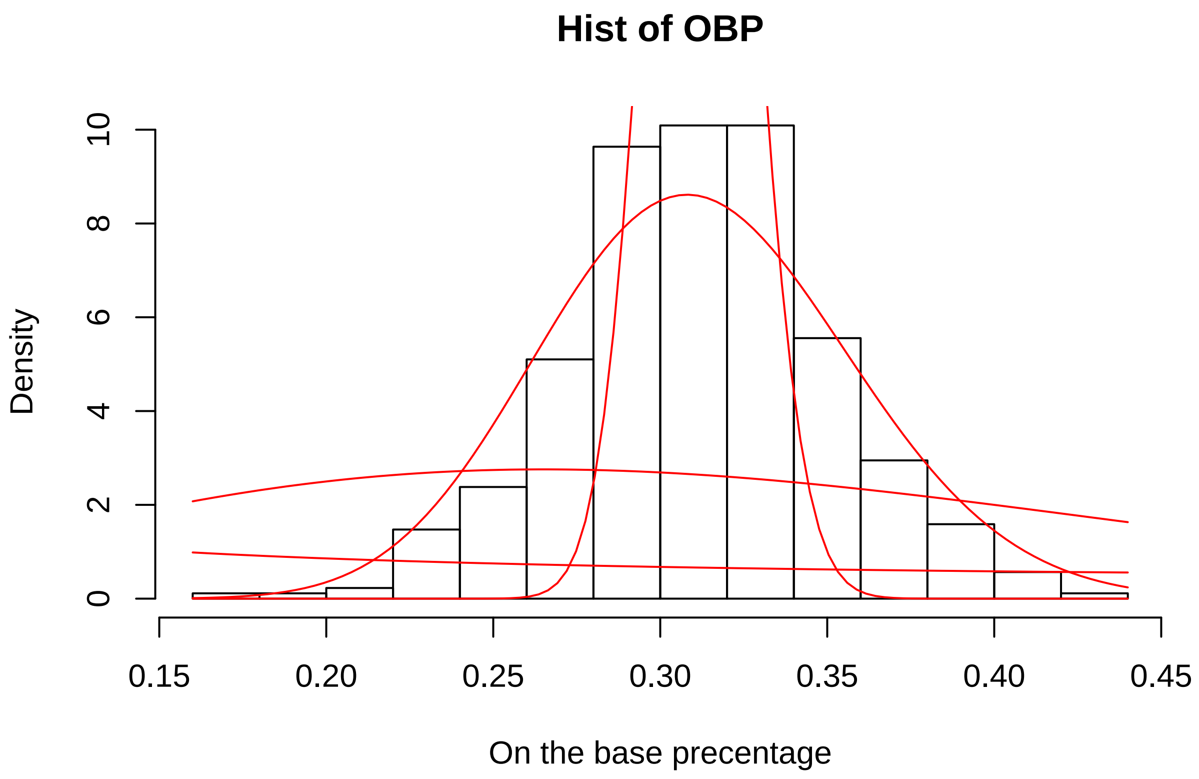
```r
hist(baseball$OBP,probability=TRUE)
```



```
#The distribution of OBP seems normal.
```

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?

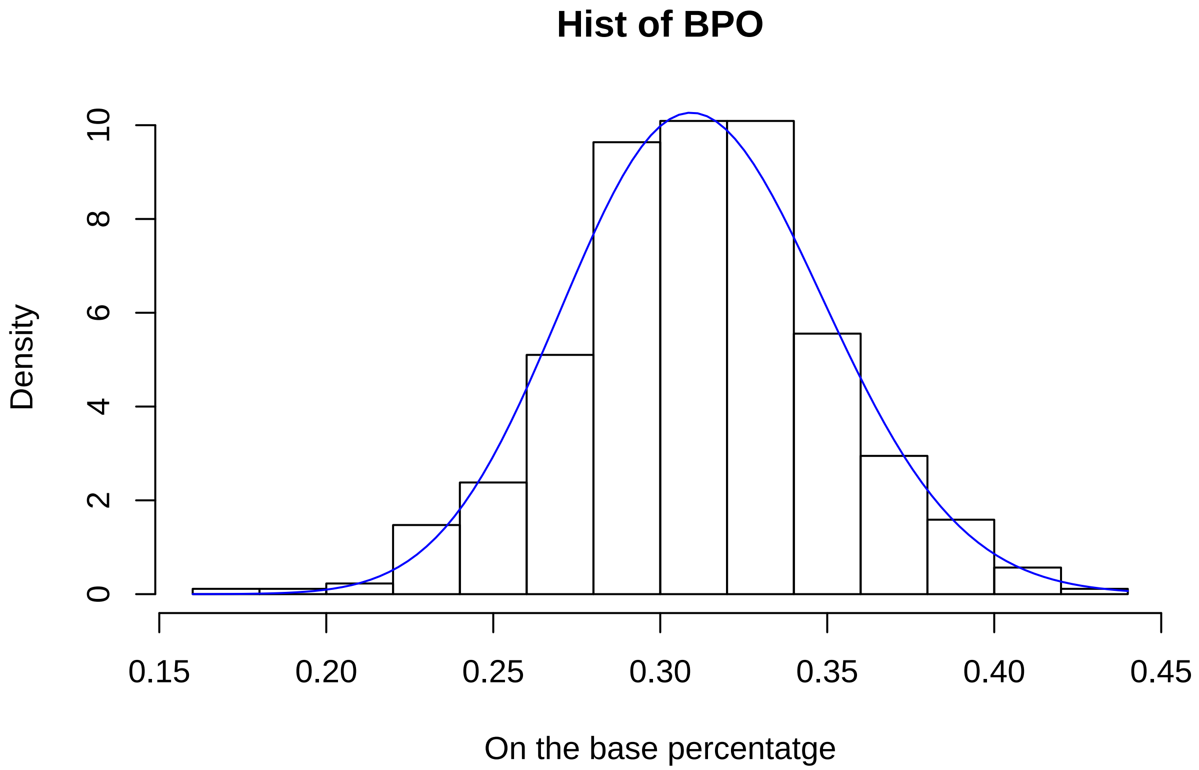
```
mean_OBP = mean(baseball$OBP)
parameters = function(mean_OBP,sum) {
  num = mean_OBP*sum
  result = c(alpha = num,beta = sum - num)
  return(result)
}
hist(baseball$OBP,probability = T, xlab = "On the base precentage", main = "Hist of OBP")
curve(dbeta(x,shape1 = parameters(mean_OBP, sum = 1)[1],shape2 = parameters(mean_OBP,sum=1)[2]),add = T
curve(dbeta(x,shape1 = parameters(mean_OBP, sum = 10)[1],shape2 = parameters(mean_OBP,sum=10)[2]),add =
curve(dbeta(x,shape1 = parameters(mean_OBP, sum = 100)[1],shape2 = parameters(mean_OBP,sum=100)[2]),add
curve(dbeta(x,shape1 = parameters(mean_OBP, sum = 1000)[1],shape2 = parameters(mean_OBP,sum=1000)[2]),a
```



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?

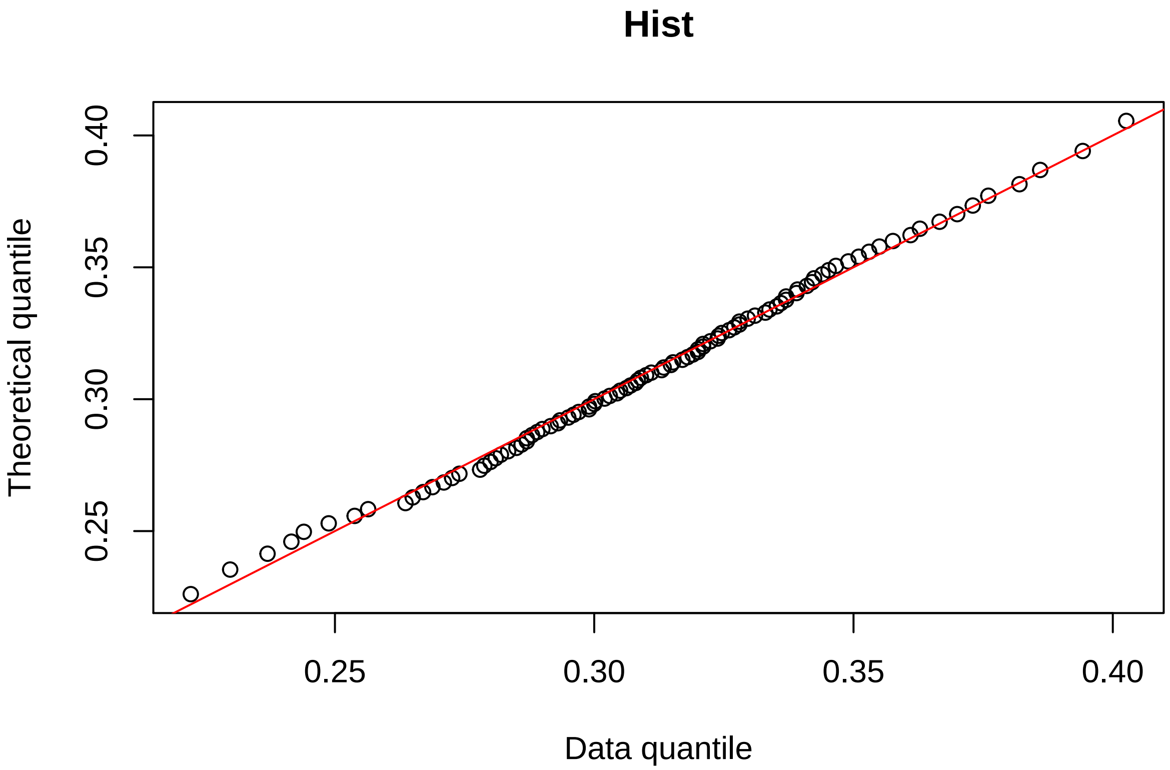
```
beta_momoent_fit = function(data) {
  mean_data = mean(data)
  var_data = var(data)
  result = c(alpha = (mean_data/var_data)*(mean_data*(1-mean_data)-var_data),beta = ((1-mean_data)/var_d
  return(result)
  }
moment_fit = beta_momoent_fit(baseball$OBP)
hist(baseball$OBP,probability = TRUE,xlab = "On the base percentatge",main = "Hist of BPO")
curve(dbeta(x,shape1 = moment_fit[1],shape2 = moment_fit[2]), add = TRUE,col = "blue")
```



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?

```
quan = quantile(baseball$OBP, probs = seq(1, 99)/100)
beta_quant = qbeta(seq(1,99)/100,shape1 = moment_fit[1],shape2 = moment_fit[2])

plot(quan,beta_quant, xlab = "Data quantile", ylab = "Theoretical quantile", main = "Hist")
abline(0,1,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 Method of Moments fit for your starting position. How do these values compare?

```
MMest = moment_fit
logLik = function(params){
  return(-sum(dbeta(baseball$OBP,params[1],params[2],log = TRUE)))
}
MLEest = nlm(logLik,MMest)$est
MLEest
```

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