МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

МГТУ им Н.Э.Баумана

Факультет ФН
Кафедра вычислительной математики и математической физики

Соколов Арсений Андреевич

Домашнее задание №4 по математической статистике

3 курс, группа ФН11-53Б
Вариант 9

Преподаватель
____________ Т. В. Облакова
«___» _______ 2019 г.

Москва, 2019 г.

1 Построение доверительного интервала Клоппера-Пирсона

Используя выборку, сгенерированную нами в задаче 2 и считая параметр p неизвестным (k-дано), построим для уровней доверия $1-\alpha = 0.9, 0.95, 0.98$ симметричные интервальные оценки Клоппера-Пирсона ($\underline{p}; \bar{p}$) для вероятности успеха в одном испытании p.

Решение.

Рассмотрим выборку, сгенерированную нами во второй задаче:

```
> emp_sample
[1] 6 6 4 6 5 5 8 5 5 4 8 8 7 4 8 3 8 7 5 5 7 6 3
[24] 4 6 8 7 7 6 8 7 4 6 5 7 6 8 7 5 5 6 7 6 6 5 4
[47] 5 5 3 5 8 6 6 5 7 5 4 6 6 6 7 5 7 7 4 4 8 6 5
[70] 6 5 7 5 4 6 5 4 5 6 6 8 7 3 4 6 7 6 6 5 8 7 6
[93] 6 6 6 5 6 5 5 8 6 6 7 7 5 4 4 4 5 6 7 4 2 4 6
[116] 4 6 6 7 6 4 7 6 6 7 4 6 7 7 6 7 6 7 5 8 6 5 5
[139] 4 5
```

Интервальными оценками Клоппера-Пирсона для биномиального закона будут:

$$\underline{p} = qbeta\left(\frac{\alpha}{2}, \sum_{k=1}^{n} X_k, nk - \sum_{k=1}^{n} X_k + 1\right), \tag{1}$$

$$\bar{p} = qbeta\left(1 - \frac{\alpha}{2}, \sum_{k=1}^{n} X_k + 1, nk - \sum_{k=1}^{n} X_k\right), \tag{2}$$

где $qbeta(\cdots)$ – это квантильная функция (inverse CDF) бета распределения. Тогда согласно 1 и 2 имеем:

```
cp_ci_LB.01 <- qbeta(0.1/2, sum(emp_sample),
                     n*k - sum(emp_sample) + 1)
cp_ci_LB.005 <- qbeta(0.05/2, sum(emp_sample),
                      n*k - sum(emp_sample) + 1)
cp_ci_LB.002 <- qbeta(0.02/2, sum(emp_sample),
                      n*k - sum(emp_sample) + 1)

cp_ci_UB.01 <- qbeta(1 - 0.1/2, sum(emp_sample) + 1,
                     n*k - sum(emp_sample))
cp_ci_UB.005 <- qbeta(1 - 0.05/2, sum(emp_sample) + 1,
                      n*k - sum(emp_sample))
cp_ci_UB.002 <- qbeta(1 - 0.02/2, sum(emp_sample) + 1,
                      n*k - sum(emp_sample))
```

```
cp_cis <- as.data.frame( matrix(c(cp_ci_LB.01, cp_ci_UB.01,
                                  cp_ci_LB.005, cp_ci_UB.005,
                                  cp_ci_LB.002, cp_ci_UB.002),
                                byrow = T, ncol = 2))
colnames(cp_cis) <- c("Lower", "Upper")
rownames(cp_cis) <- c("0.1", "0.05", "0.02")
> cp_cis
alpha  Lower     Upper
0.1  0.6775675 0.7234332
0.05 0.6731287 0.7275974
0.02 0.6679422 0.7324052
```

2 Доверительные интервалы по ЦПТ

Для тех же уровней найдём по ЦПТ приближенные доверительные интервалы для p.

Решение.

Рассмотрим статистику $\sum_{j=1}^{n} X_j$, где X_j представляют собой значения выборки *emp_sample*.

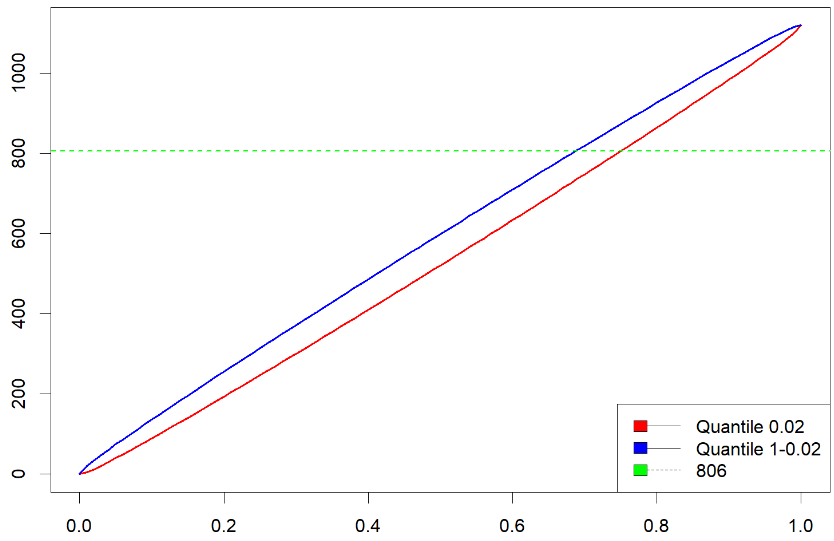
```
> stat1
[1] 806
```

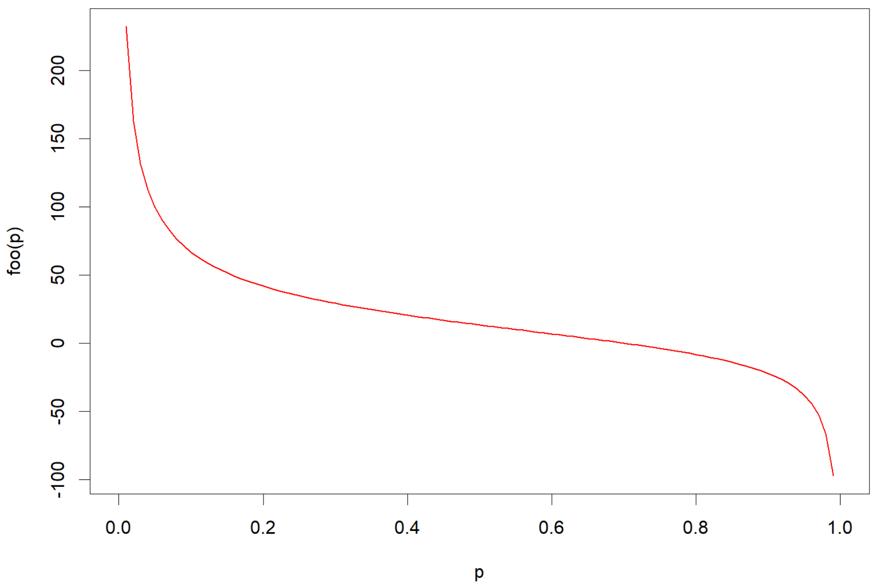
Рассмотрим квантили уровней $\frac{\alpha}{2}$ и $1 - \frac{\alpha}{2}$ для биномиального закона для $\alpha = 0.02$:

```
uup <- function(theta) qbinom(0.02/2, n*k, theta)
lou <- function(theta) qbinom(1 - 0.02/2, n*k, theta)
plot(uup, col = "red")
plot(lou, col = "blue", add = T)
abline(h = stat1, col = "green", lty = 2)
legend("bottomright",
c("Quantile 0.02", "Quantile 1-0.02", "Sum of sample values"),
lty=c(1,1,2),
fill=c("red", "blue", "green"))
```

Кроме того рассмотрим функцию:

$$foo(\theta) = \frac{\sum\limits_{k=1}^{n} X_j - nk\theta}{\sqrt{nk\theta \cdot (1-\theta)}}$$


1000
800
600
400
200
0
0.0
0.2
0.4
0.6
0.8
1.0
Quantile 0.02
Quantile 1-0.02
806


foo(p)
200
150
100
50
0
-50
-100
0.0
0.2
0.4
0.6
0.8
1.0
p

Верхние и нижние оценки найдём из решений соответствующих уравнений:

```
library(rootSolve)

clt_ci_LB.01_foo <- function(p) (sum(emp_sample) - n*k*p)/
(sqrt(n*k*p*(1-p))) - qnorm(1-0.1/2)
clt_ci_UB.01_foo <- function(p) (sum(emp_sample) - n*k*p)/
(sqrt(n*k*p*(1-p))) + qnorm(1-0.1/2)

clt_ci_LB.005_foo <- function(p) (sum(emp_sample) - n*k*p)/
(sqrt(n*k*p*(1-p))) - qnorm(1-0.05/2)
clt_ci_UB.005_foo <- function(p) (sum(emp_sample) - n*k*p)/
(sqrt(n*k*p*(1-p))) + qnorm(1-0.05/2)

clt_ci_LB.002_foo <- function(p) (sum(emp_sample) - n*k*p)/
(sqrt(n*k*p*(1-p))) - qnorm(1-0.02/2)
clt_ci_UB.002_foo <- function(p) (sum(emp_sample) - n*k*p)/
(sqrt(n*k*p*(1-p))) + qnorm(1-0.02/2)

clt_ci_LB.01 <- uniroot(clt_ci_LB.01_foo, c(0.1, 0.9))$root
clt_ci_UB.01 <- uniroot(clt_ci_UB.01_foo, c(0.1, 0.9))$root

clt_ci_LB.005 <- uniroot(clt_ci_LB.005_foo, c(0.1, 0.9))$root
clt_ci_UB.005 <- uniroot(clt_ci_UB.005_foo, c(0.1, 0.9))$root

clt_ci_LB.002 <- uniroot(clt_ci_LB.002_foo, c(0.1, 0.9))$root
clt_ci_UB.002 <- uniroot(clt_ci_UB.002_foo, c(0.1, 0.9))$root

clt_cis <- as.data.frame( matrix(c(clt_ci_LB.01, clt_ci_UB.01,
clt_ci_LB.005, clt_ci_UB.005,
clt_ci_LB.002, clt_ci_UB.002),
byrow = T, ncol = 2))
colnames(clt_cis) <- c("Lower", "Upper")
rownames(clt_cis) <- c("0.1", "0.05", "0.02")

> clt_cis
alpha  Lower     Upper
0.1  0.6779266 0.7228906
0.05 0.6734280 0.7269840
0.02 0.6681609 0.7316903
```

3 Сравнение полученных результатов

Сравним полученные в пп. 1 и 2 результаты для доверительных интервалов.
Решение.

```
compare_CI <- as.data.frame(matrix(c(clt_ci_LB.01, cp_ci_LB.01,
clt_ci_UB.01, cp_ci_UB.01,
clt_ci_LB.005, cp_ci_LB.005,
clt_ci_UB.005, cp_ci_UB.005,
clt_ci_LB.002, cp_ci_LB.002,
clt_ci_UB.002, cp_ci_UB.002),
ncol = 2, byrow = T))
```

Таблица 1: CLT vs. CP CIs

alpha	CI	V1	V2
alpha=0.1	L	0.6779266	0.6775675
	U	0.7228906	0.7234332
alpha=0.05	L	0.6734280	0.6731287
	U	0.7269840	0.7275974
alpha=0.02	L	0.6681609	0.6679422
	U	0.7316903	0.7324052

Мы видим, что различия несущественные и все полученные интервалы содержат истинное значение параметра $p = 0.7$.

4 Построение CDF с CP CIs

Для одного из значений α построим совмещённые графики функций распределения биномиальных законов $B(k, p)$, $B(k, \underline{p})$, $B(k, \bar{p})$. Рассмотрим $\alpha = 0.02$.

```
theor_distr <- rbinom(n, k, p1)
distr_UB <- rbinom(n, k, cp_ci_UB.002)
distr_LB <- rbinom(n, k, cp_ci_LB.002)

plot(ecdf(theor_distr),
col = "red", lwd = 3, verticals = T, axes = F,
xlim = c(0,k+1), ylim = c(0,1.2),
xlab = "Value", ylab = "CDF", main = "CDF with CIs")
```

```r
plot(ecdf(distr_LB),
col = "black", lwd = 2, verticals = T, add = T)
plot(ecdf(distr_UB),
col = "blue", lwd = 2, verticals = T, add = T)
axis(1, c(1:k))
axis(2, seq(0.0, 1.2, 0.2), las = 1)
grid(nx = k+1, ny = 1.2 / 0.2)
legend("bottomright", c("Theoretical", "Lower", "Upper"),
lty=c(1,1,1),
fill=c("red", "black", "blue"))
```