HwK#6

Euchie Jn Pierre

2022-11-09

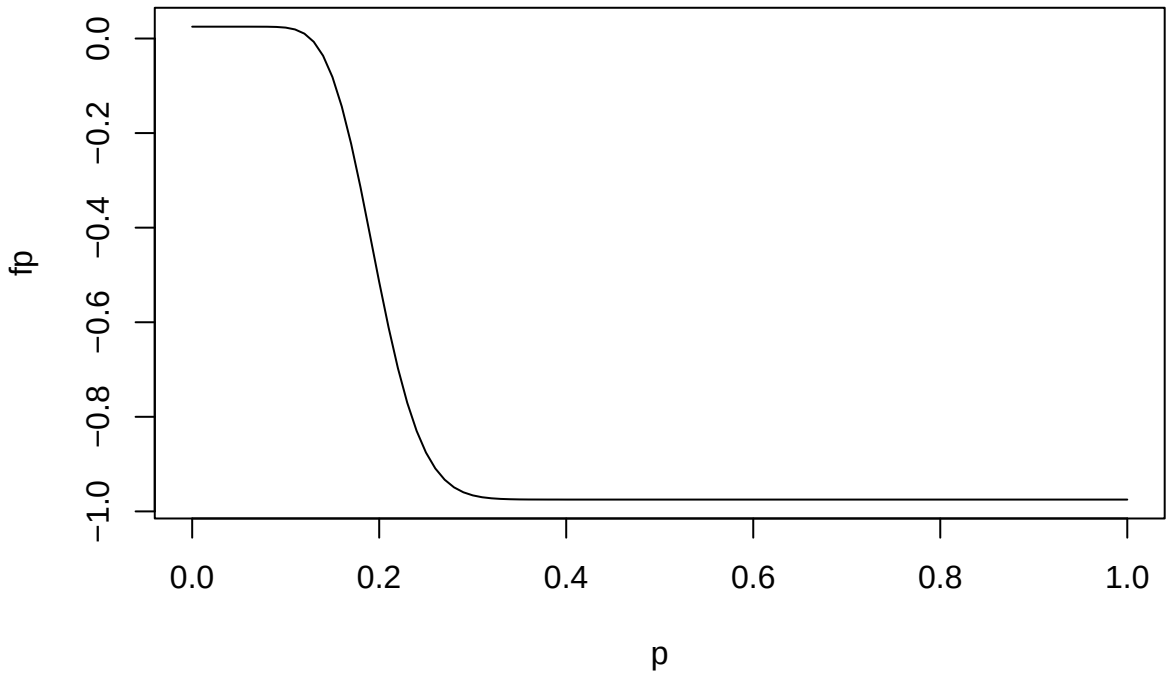
This homework is to practice finding the root for a function and exact confidence intervals.

NewtonRaphson function

```
#newtonraphson function
newtonraphson <- function(ftn, x0, tol = 1e-9, max.iter = 100) {
  x <- x0 # x0: the initial value
  fx <- ftn(x)
  iter <- 0
  while ((abs(fx[1]) > tol) & (iter < max.iter)) {
    x <- x - fx[1]/fx[2]
    fx <- ftn(x)
    iter <- iter + 1
    cat("At iteration", iter, "value of x is:", x, "\n")
  }
  if (abs(fx[1]) > tol) {
    cat("Algorithm failed to converge\n")
    return(NULL)
  } else { # abs(fx[1]) <= tol
    cat("Algorithm converged\n")
    return(x)
  }
}
```

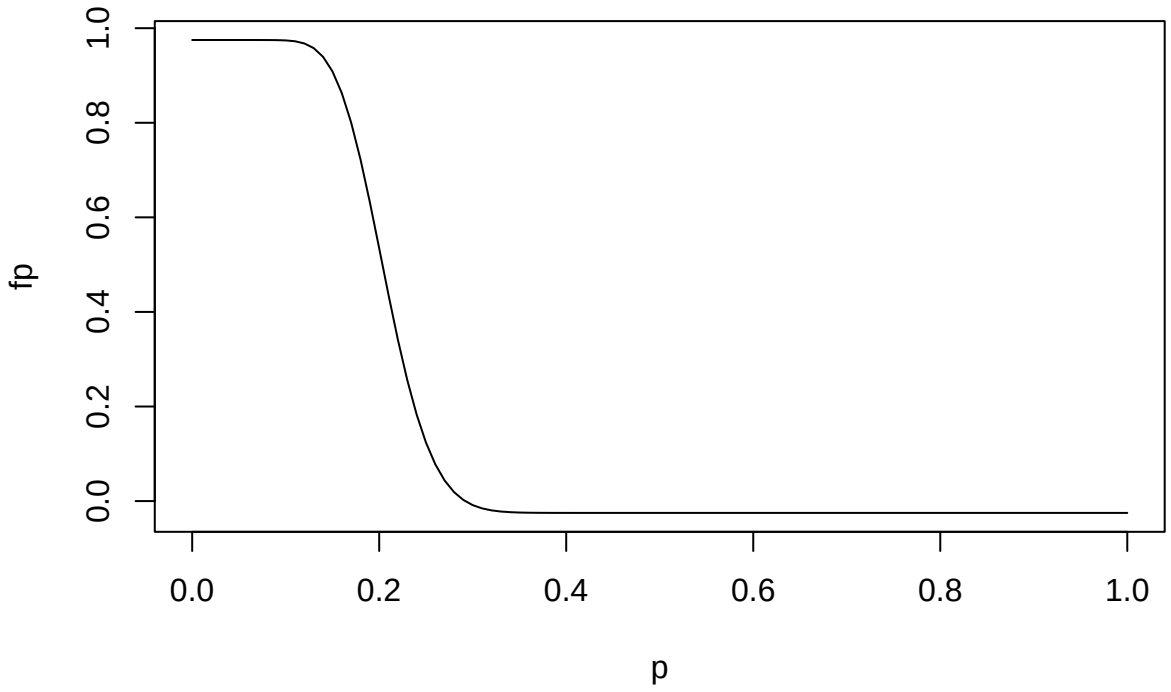
Ex 17-1 (Exact confidence intervals)

```
#pL
p <- seq(0,1,0.01)
fp <- (-0.975)
for (k in 0:19) {
    fp <- fp+choose(100,k)*(p^k)*((1-p)^(100-k))
}
plot(p,fp, type = "l")
```




```r
ftn1 <- function(p) {
  fp <- (-0.975)
  dfp <- 0
  for (k in 0:19) {
    fp <- fp+choose(100,k)*(p^k)*((1-p)^(100-k))
    dfp <- dfp+choose(100,k)*(k*(p^(k-1))*((1-p)^(100-k))-(p^k)*(100-k)*((1-p)^(99-k)))
  }
  return(c(fp, dfp))
}

#pU
p <- seq(0,1,0.01)
fp <- (-0.025)
for (k in 0:20) {
  fp <- fp+choose(100,k)*(p^k)*((1-p)^(100-k))
}
plot(p,fp, type = "l")
```




```
ftn2 <- function(p) {
  fp <- (-0.025)
  dfp <- 0
  for (k in 0:20) {
    fp <- fp+choose(100,k)*(p^k)*((1-p)^(100-k))
    dfp <- dfp+choose(100,k)*(k*(p^(k-1))*((1-p)^(100-k))-(p^k)*(100-k)*((1-p)^(99-k)))
  }
  return(c(fp, dfp))
}
#pL
newtonraphson(ftn1, 0.19, 1e-9)
```

```
## At iteration 1 value of x is: 0.1490968
## At iteration 2 value of x is: 0.1343497
## At iteration 3 value of x is: 0.1280073
## At iteration 4 value of x is: 0.1267065
## At iteration 5 value of x is: 0.1266556
## At iteration 6 value of x is: 0.1266556
## Algorithm converged
```

```
## [1] 0.1266556
```

```
#pU
newtonraphson(ftn2, 0.25, 1e-9)
```

```
## At iteration 1 value of x is: 0.2735477
## At iteration 2 value of x is: 0.2867346
## At iteration 3 value of x is: 0.2913356
## At iteration 4 value of x is: 0.2918372
## At iteration 5 value of x is: 0.2918427
## Algorithm converged
```

```
## [1] 0.2918427
```

```r
binom.confint(20, 100, conf.level = 0.95, methods = "all")
```

```
##           method  x   n      mean     lower     upper
## 1  agresti-coull 20 100 0.2000000 0.1326077 0.2895884
## 2     asymptotic 20 100 0.2000000 0.1216014 0.2783986
## 3          bayes 20 100 0.2029703 0.1272666 0.2819852
## 4        cloglog 20 100 0.2000000 0.1283119 0.2832200
## 5          exact 20 100 0.2000000 0.1266556 0.2918427
## 6          logit 20 100 0.2000000 0.1328161 0.2898107
## 7         probit 20 100 0.2000000 0.1310048 0.2871983
## 8        profile 20 100 0.2000000 0.1297551 0.2854280
## 9            lrt 20 100 0.2000000 0.1297488 0.2854267
## 10     prop.test 20 100 0.2000000 0.1292482 0.2943230
## 11        wilson 20 100 0.2000000 0.1333669 0.2888292
```

Ex 17-1 (asymptotic confidence intervals)

```r
#asymptotic confidence intervals
aci <- 0.2
aci-qnorm(0.975)*sqrt((aci*(1-aci))/100)
```

```
## [1] 0.1216014
```

```r
aci+qnorm(0.975)*sqrt((aci*(1-aci))/100)
```

```
## [1] 0.2783986
```