Power functions

STAT 346/446

Lecture 10

Poisson Example

Let $X_1, \ldots, X_n$ be a random sample from Poisson(λ) We want to test the hypotheses

$$H_0 : \lambda \leq 5.3 \qquad \text{versus} \qquad H_1 : \lambda > 5.3$$

We will reject H_0 if $\overline{X} \geq c$. The power function for this test is

$$\beta(\lambda) = P_\lambda(\overline{X} \geq c) = P_\lambda\left(\sum_{i=1}^{n} X_i \geq cn\right)$$

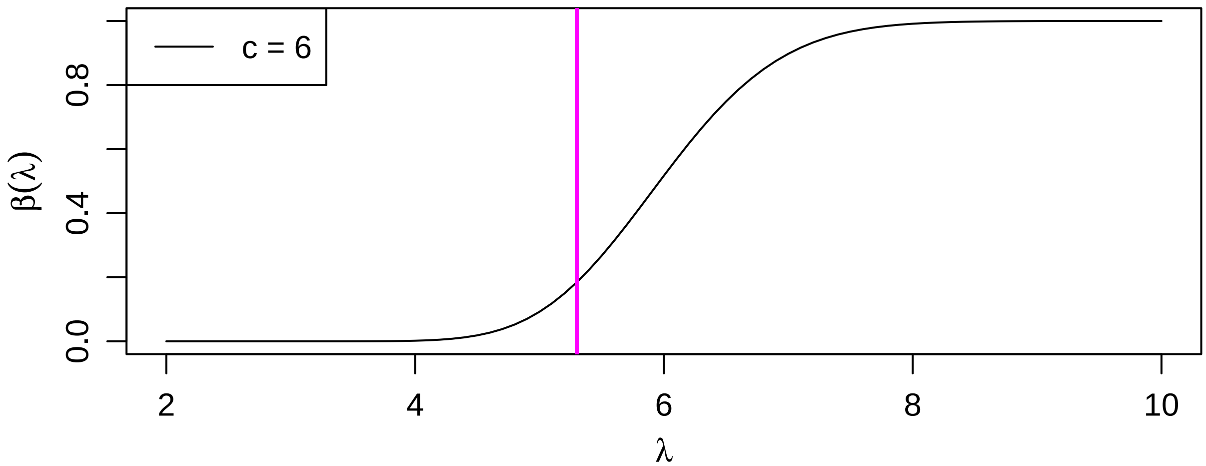
where $\sum_{i=1}^{n} X_i \sim \text{Poisson}(n\lambda)$. The following is a code that calculates this function for any given sample size n and any value of c. We use the `ceiling` function to account for possibly non-integer values of c.

```
BetaFun <- function(lambda, c, n = 10){
  tmp <- c()
  for (i in 1:length(lambda)){
    tmp[i] <- 1 - sum(dpois(0:ceiling(n*c - 1), lambda = n*lambda[i]))
  }
  return(tmp)
}

plotBeta <- function(cc, n = 10){
  par(mar=c(3.5,3.5,1,1), mgp=c(2,0.8,0))
  litir <- 1:length(cc)
  lambda.grid <- seq(2.0,10, by = 0.1)
  plot(lambda.grid, BetaFun(lambda.grid , c = cc[1], n=n), type = 'l', col = litir[1],
     xlab = expression(lambda), ylab = expression(beta(lambda)))
  abline(v=5.3, col='magenta', lwd=2)
  if(length(cc)>1){
    for(i in 2:length(cc)){
      lines(lambda.grid, BetaFun(lambda.grid , c = cc[i], n=n), col=litir[i])
  }}
  legend("topleft", legend = paste("c =", cc), col = litir, lty=1 )
}
```

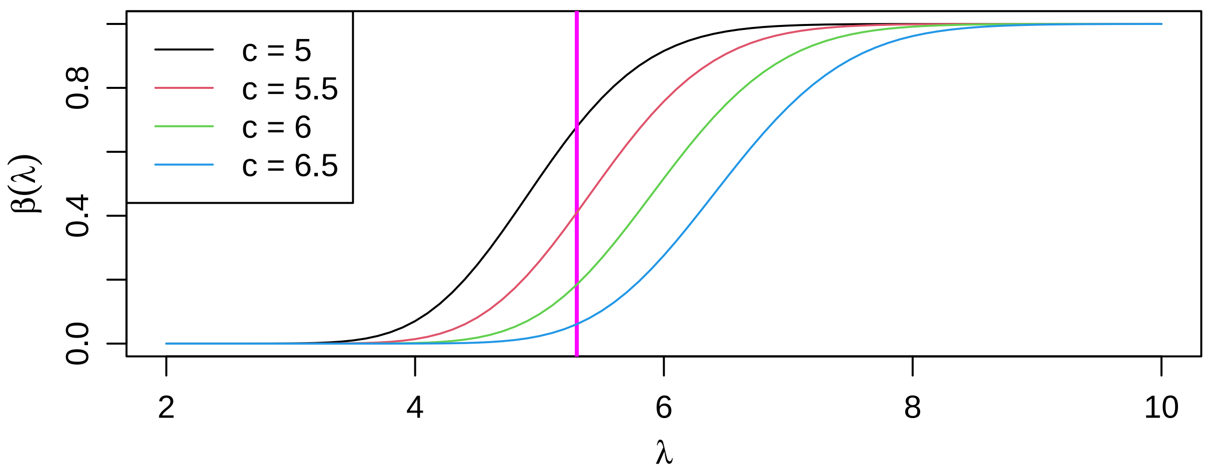
The power function $\beta(\lambda)$ for $n = 10$ and $c = 6$.

```
plotBeta(cc=6)
```



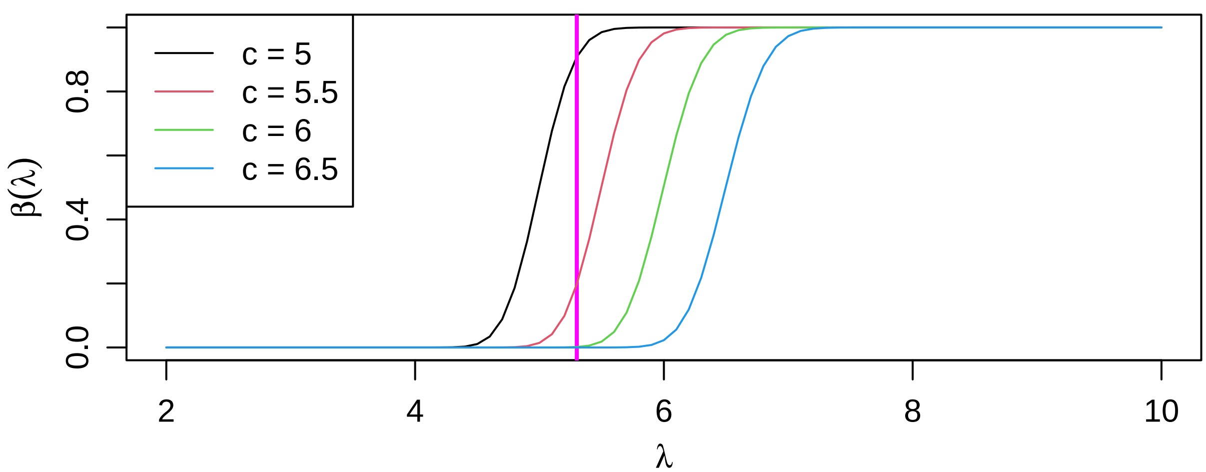
The power function for $n = 10$ and different values of c

```
plotBeta(cc = c(5, 5.5, 6, 6.5))
```



The power function for $n = 100$ and different values of c

```
plotBeta(cc = c(5, 5.5, 6, 6.5), n=100)
```



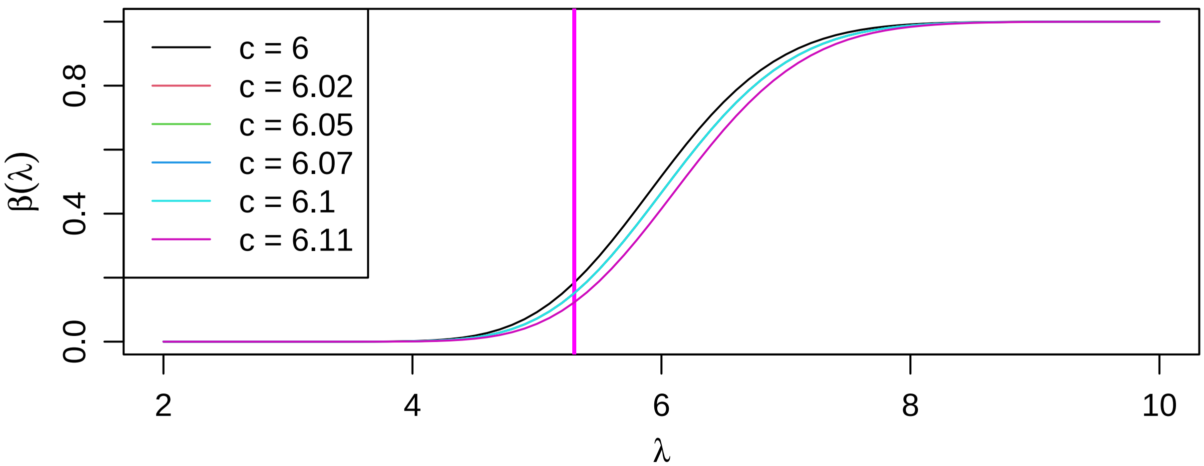
Size / level for Poisson example

Note that we get the same power function for several c values

$$\beta(\lambda) = P_\lambda\left(\sum_{i=1}^{n} X_i \geq cn\right) = 1 - P_\lambda\left(\sum_{i=1}^{n} X_i \leq \lceil cn - 1 \rceil\right)$$

The power function for $n = 10$ and different values of c that give the same power function

```
plotBeta(cc = c(6.00, 6.02, 6.05, 6.07, 6.10, 6.11), n=10)
```



Note that $c \in (6, 6.1]$ all give the same power function - the curves are plotted over the previous ones. This means that we can't obtain all α sizes - only levels.

Size of the test for different n and c.

```
BetaFun(lambda = 5.3, c = 6, n=10)
```

```
## [1] 0.1846032
```

```
BetaFun(lambda = 5.3, c = 6, n=100)
```

```
## [1] 0.001521488
```

```
BetaFun(lambda = 5.3, c = 6.5, n=10)
```

```
## [1] 0.06065293
```

Suppose $n = 10$. Can we find a c so that we get a size 0.05 test?

```
cc <- seq(6.4,6.9, by = 0.1)
Prob <- c()
for (k in 1:length(cc)){
  Prob[k] <- BetaFun(lambda = 5.3, c = cc[k], n=10)
}
rbind(cc,Prob)
```

```
##             [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
## cc   6.40000000 6.50000000 6.60000000 6.70000000 6.80000000 6.90000000
## Prob 0.07773857 0.06065293 0.04672156 0.03553425 0.02668459 0.01978706
```

We see that $\beta(5.3) = 0.06065$ when $c = 6.6$ and $\beta(5.3) = 0.0467$ when $c = 6.6$. Any other c value between 6.5 and 6.6 will also give $\beta(5.3) = 0.0467$:

```r
cc <- seq(6.5, 6.6, by = 0.01)
Prob <- c()
for (k in 1:length(cc)){
  Prob[k] <- BetaFun(lambda = 5.3, c = cc[k], n=10)
}
rbind(cc,Prob)
```

```
##            [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
## cc   6.50000000 6.51000000 6.52000000 6.53000000 6.54000000 6.55000000
## Prob 0.06065293 0.04672156 0.04672156 0.04672156 0.04672156 0.04672156
##            [,7]       [,8]       [,9]      [,10]      [,11]
## cc   6.56000000 6.57000000 6.58000000 6.59000000 6.60000000
## Prob 0.04672156 0.04672156 0.04672156 0.04672156 0.04672156
```

Therefore, to get a **level** 0.05 test we can pick any c in the range $(6.5, 6.6]$. But we cannot get a **size** 0.05 test.

Normal example

```
## One-sided z-test H0 theta >= mu0
powerZ <- function(theta, theta0, cstar, n=100, sigma = 1){
  return(pnorm( cstar - (theta-theta0)*sqrt(n)/sigma))
}
sigma <- 6; n <- 220; theta0 <- 22
cval <- c(-2, -1, 0, 1, 2)

par(mfrow=c(1,1), mar=c(3.5,3.5,2,1), mgp=c(2,0.8,0))
plot(c(0,0), ylim = c(0,1), xlim=c(20, 27), type="n",
     ylab=expression(beta(theta)), xlab=expression(theta),
     main = "Power function for school example" )
for(k in 1:5){
  curve(powerZ(x, theta0, cval[k], n, sigma), from = 20, to=27, add=T,
        col=k, lwd=2)
}
abline(v=22, col="magenta")
legend("topright", legend=paste("c.star = ", cval), col=1:5, lwd=2)
```