

1. Write a `pois.prob()` function that computes  $P(X = x)$ ,  $P(X \neq x)$ ,  $P(X < x)$ ,  $P(X \leq x)$ ,  $P(X > x)$ , and  $P(X \geq x)$ . Enable the user to specify the rate parameter  $\lambda$ .

```
pois.prob <- function(x, size, prob, type="<="){
  # Use dpois and ppois to conditionally return the correct probability
  #dpois is for PMF
  #ppois is for CDF
  # prob is lambda

  #cdf of anything negative should just be zero
  if (x-1 < 0){
    sub <- 0
  }
  else{
    sub=prob-1
  }

  #prob that x== is just the pmf(==)
  equal <- dpois(x,prob)
  #prob that x!= is 1-pmf(==)
  not.equal <- 1- dpois(x,prob)
  #prob that x< is cdf(== - 1)
  less <- ppois(sub,prob,TRUE)
  #prob that x <= is cdf(==)
  less.or.equal <- ppois(x,prob,TRUE)
  #prob that x> is 1-cdf(==)
  greater <- 1- ppois(x,prob,TRUE)
  #prob that x>= is 1-cdf(==-1)
  greater.or.equal <- 1-ppois(sub,prob,TRUE)

  to.return <- data.frame(c("Equal", "Not Equal", "Less Than", "Less than or Equal to",
    "Greater Than", "Greater Than or Equal to"),
    c(equal,not.equal,less,less.or.equal,greater,greater.or.equal))

  return(to.return)
}
```

2. Write a `beta.prob()` function that computes  $P(X = x)$ ,  $P(X \neq x)$ ,  $P(X < x)$ ,  $P(X \leq x)$ ,  $P(X > x)$ , and  $P(X \geq x)$  for a beta distribution. Enable the user to specify the shape parameters  $\alpha$  and  $\beta$ .

```
beta.prob <- function(x, size, prob, type="<="){
  # Use dbeta and pbeta to conditionally return the correct probability
  #dbeta is for PMF
  #pbeta is for CDF
  # prob is lambda

  #prob that x== is just 0
  equal <-0
  #prob that x!= is 1
  not.equal <- 1
  #prob that x< is cdf(==)
  less <- ppois(x,prob,TRUE)
  #prob that x <= is cdf(==)
  less.or.equal <- ppois(x,prob,TRUE)
  #prob that x> is 1-cdf(==)
  greater <- 1- ppois(x,prob,TRUE)
  #prob that x>= is 1-cdf(==)
  greater.or.equal <- 1-ppois(x,prob,TRUE)

  to.return <- data.frame(c("Equal", "Not Equal", "Less Than", "Less than or Equal to",
    "Greater Than", "Greater Than or Equal to"),
    c(equal,not.equal,less,less.or.equal,greater,greater.or.equal))

  return(to.return)
}
```