

```
#####  
#####
```

```
#####          EXERCÍCIO I
```

```
#####
```

```
#####  
#####
```

```
#
```

```
# Exercício sobre máxima verosimilhança com dados de  
presença-ausência
```

```
# Objetivos:
```

```
# Obter estimativa de parametros psi e p com base em  
método de máxima verosimilhança
```

```
# Representar superfície de verosimilhança em gráfico
```

```
# Construir base de dados fictícia
```

```
DADOS <- rbind(c(0,0,0,0,0),  
              c(0,0,0,0,0),  
              c(1,1,1,1,0),  
              c(0,1,1,1,1),  
              c(1,0,1,1,1),  
              c(0,1,1,1,1),  
              c(0,0,0,0,0),  
              c(1,1,1,1,0),  
              c(0,0,0,0,0),  
              c(0,0,0,0,0))
```

```
r <- dim(DADOS)[1]
```

```
c <- dim(DADOS)[2]
```

```
# declarar vetor com possíveis valores de psi
```

```

psis <- seq(0.01,0.99,0.01)
# declarar vetor com possíveis valores de p
ps <- seq(0.01,0.99,0.01)
# Declarar matriz para guardar valores de verosimilhança
LIKE <-
matrix(rep(0,length(psis)*length(ps)),nrow=length(psis))

# Calcular valores de verosimilhança para cada par de
valores de psi e p
# o modelo assume p constante no tempo (colunas) e psi
igual para todos os
# lugares (linhas)
for (i in 1:length(psis)) {
  for (j in 1:length(ps)) {
    # obter valores de psi e p para esta iteração
    psi <- psis[i]
    p <- ps[j]
    # declarar vetor para guardar verosimilhança das linhas
de dados
    llikerow <- rep(0,r)
    # calcular verosimilhança das linhas de dados
(loglikelihood)
    for (k in 1:r) {
      viu <- sum(DADOS[k,])
      if(viu==0) {
        llikerow[k] <- -log((1-psi)+psi*(1-p)^5)
      } else {
        llikerow[k] <- -log(psi*(p^viu)*((1-p)^(c-viu)))
      }
    }
  }
}

```

```
    }  
    LIKE[i,j] <- sum(llikerow)  
  }  
}  
  
# Obter estimativa de psi e p de máxima verosimilhança  
estpsi <- psis[which(LIKE==min(LIKE),arr.ind=TRUE)[1]]  
estp <- ps[which(LIKE==min(LIKE),arr.ind=TRUE)[2]]  
  
# Representação gráfica  
contour(LIKE,xlab="psi",ylab="p")  
abline(v=estpsi,lty=2)  
abline(h=estp,lty=2)
```