

```

#####
#####

#####   EXERCÍCIO III
#####

#####
#####

#

# Parte 1: Exercício sobre crescimento logístico discreto e contínuo (determinístico)

#

# testar com r=0.05, K=80 depois com r=2.8

#

cresclog <- function(No,r,K,tmax) {
  # Declarar matriz de três colunas que vai receber resultados
  PROJLOG <- matrix(rep(0,3*tmax),nrow=tmax)
  # Colocar tempo na primeira coluna
  PROJLOG[,1] <- seq(0,tmax-1)
  # Colocar tamanho inicial na primeira linha da segunda e terceira colunas
  PROJLOG[1,2:3] <- No
  # Colocar crescimento contínuo e crescimento discreto na segunda e terceira colunas
  for (t in 2:tmax) {
    cat("t=",t-1,"\n")
    # usar integração para o crescimento contínuo
    PROJLOG[t,2] <- K / (1 + ((K-No)/No)*exp(-r*(t-1)))
    # usar fator de crescimento discreto rd=r para crescimento discreto
    lastN <- PROJLOG[t-1,3]
    PROJLOG[t,3] <- lastN + r*lastN*(1-lastN/K)
  }
  # Representação gráfica

```

```
plot(PROJLOG[,1],PROJLOG[,3],type="l",lty=2,xlab="tempo (t)", ylab="tamanho da
população (N)")
```

```
lines(PROJLOG[,1],PROJLOG[,2])
```

```
x <- 0.58*tmax
```

```
y <- min(PROJLOG[,2:3])+ 0.2*(max(PROJLOG[,2:3])-min(PROJLOG[,2:3]))
```

```
legend(x,y,c("cresc. discreto","cresc. contínuo"),lty=2:1)
```

```
# Apresentar matriz com os valores
```

```
return(PROJLOG)
```

```
}
```

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

```
#
```

```
# Parte 2: Um atrator no crescimento logístico discreto
```

```
#
```

```
# Usar o crescimento logístico discreto da projeção produzida na Parte 1
```

```
# e representar  $N(t+1)$  contra  $N(t)$ 
```

```
par(mfrow=c(1,2)) # informar que os gráficos vão aparecer em dois painéis em uma  
linha
```

```
# preencher o primeiro painel
```

```
plot(PROJLOG[,1],PROJLOG[,3],type="l",lty=2,xlab="tempo (t)", ylab="tamanho da  
população (N)")
```

```
# preencher o segundo painel
```

```
tmax <- dim(PROJLOG)[1] # colocar o valor tmax numa variável, com base no número  
de linhas da projeção
```

```
plot(PROJLOG[1:tmax-  
1,3],PROJLOG[2:tmax,3],type="l",col="gray",xlab="N[t]",ylab="N[t+1]")
```

```
points(PROJLOG[1:tmax-1,3],PROJLOG[2:tmax,3],pch=20)
```

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

```
#
```

```
# Parte 3: Comparação do crescimento logístico discreto com o crescimento  
exponencial
```

```
# estocástico discreto
```

```
#
```

```
# Tentar valores No=610, r=2.99, K=600, l=1.007, varl=0.03, tmax=100
```

```
#
```

```
caosvstoc <- function(No,r,K,l,varl,tmax) {
```

```
  # Declarar matriz de três colunas para receber resultados
```

```
  CSTOC <- matrix(rep(0,3*tmax),nrow=tmax)
```

```
  # Colocar tempo na primeira coluna
```

```
  CSTOC[,1] <- seq(0,tmax-1)
```

```
  # Colocar tamanho inicial na primeira linha da segunda e terceira colunas
```

```
  CSTOC[1,2:3] <- No
```

```
  # Colocar crescimento logístico discreto na segunda coluna e crescimento exponencial
```

```
  # estocástico (também discreto) na terceira
```

```
  for (t in 2:tmax) {
```

```
    # usar fator de crescimento discreto rd=r para crescimento discreto
```

```
    lastN <- CSTOC[t-1,2]
```

```
    CSTOC[t,2] <- lastN + r*lastN*(1-lastN/K)
```

```
    # usar lambda estocástico para o crescimento exponencial
```

```
    le <- rnorm(1,1,sqrt(varl)) # obter o lambda estocástico
```

```
    cat("t=",t-1,"\n")
```

```
    cat("lambda estocástico = ",le,"\n")
```

```
    CSTOC[t,3] <- CSTOC[t-1,3]*le # projetar a população
```

```

if (CSTOC[t,3]<1) {CSTOC[t,3] <- 0} # substituir por zero caso dê N menor que 1
}

# Representação gráfica

par(mfrow=c(2,2)) # gráficos em dois por dois, preencher por linhas

# plot 1: projeção exponencial

plot(CSTOC[,1],CSTOC[,3],type="l",lty=1,xlab="tempo (t)", ylab="tamanho da
população (N)",main="Exponencial estocástico")

# plot 2: busca de atractor na projeção exponencial

tmax <- dim(CSTOC)[1] # colocar o valor tmax numa variável, com base no número
de linhas da projeção

plot(CSTOC[1:tmax-
1,3],CSTOC[2:tmax,3],type="l",col="gray",xlab="N[t]",ylab="N[t+1]")

points(CSTOC[1:tmax-1,3],CSTOC[2:tmax,3],pch=20)

# plot 3: projeção logística

plot(CSTOC[,1],CSTOC[,2],type="l",lty=1,xlab="tempo (t)", ylab="tamanho da
população (N)", main="Logístico determinístico")

# plot 4:

plot(CSTOC[1:tmax-
1,2],CSTOC[2:tmax,2],type="l",col="gray",xlab="N[t]",ylab="N[t+1]")

points(CSTOC[1:tmax-1,2],CSTOC[2:tmax,2],pch=20)

# Apresentar matriz com os valores

return(CSTOC)

}

```

```

caosvstoc (610,2.99,600,1.007,0.03,100)

```