Métodos Numéricos y Simulación(Final)

Felipe de Mendiburu

Friday, June 28, 2014

Pregunta 1.

Pruebe que los vectores caracteristicos son ortogonales, Para este fin utilice la siguiente matriz de correlación de dos variables aleatorias X e Y, donde la correlación X,Y es 0.65. Utilice:

```r
A<-cbind(c(1,0.65),c(0.65,1))
```

a. Polinomio caracteristico.

$$A = \begin{bmatrix} 1 & 0.65 \\ 0.65 & 1 \end{bmatrix}$$

polinomio: $(1-\lambda)^2 - 0.65^2 = (0.35-\lambda)(1.65-\lambda)$

$\lambda_1 = 1.65$, $\lambda_2 = 0.35$

Para $\lambda_1 = 1.65$, se calcula $\phi_1 = [x_1, x_2]$ del siguiente sistema de ecuaciones:

$x_1 + 0.65x_2 = 1.65x_1$

$0.65x_1 + x_2 = 1.65x_2$

Resulta:

$-0.65x_1 + 0.65x_2 = 0$

$0.65x_1 - 0.65x_2 = 0$

si $x_1 = 1$, entonces $x_2 = 1$; asi $\phi_1 = [1, 1]$

Para $\lambda_2 = 0.35$, se calcula $\phi_2 = [x_1, x_2]$ del siguiente sistema de ecuaciones:

$x_1 + 0.65x_2 = 0.35x_1$

$0.65x_1 + x_2 = 0.35x_2$

Resulta:

$0.65 * x_1 + 0.65x_2 = 0$

$0.65 * x_1 + 0.65x_2 = 0$

si $x_1 = 1$, entonces $x_2 = -1$; asi $\phi_1 = [1, -1]$

La matriz Q formada es:

```r
Q<-cbind(c(1,1),c(1,-1))
```

$$Q = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

El producto escalar: $\phi_1 * \phi_2 = 0$, por lo tanto es ortogonal

b. Método de Jacobi. Segun la matriz definida A:

```
print(A)
```

```
##      [,1] [,2]
## [1,] 1.00 0.65
## [2,] 0.65 1.00
```

Para hacer $a_{12} = 0$, se utiliza la matriz de rotación plana P, para un ángulo de rotación θ, la matriz P estaria formada por:

$$P = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

donde $\theta = 0.5(\arctan(\frac{2a_{12}}{a_{11}-a_{22}}))$

para $a_{11} = 1$, $a_{22} = 1$ y $a_{12} = 0.65$ se tendria el ángulo θ y la matriz P

```
theta <- 0.5*atan(2*A[1,2]/(A[1,1] - A[2,2]))
grados <-round(theta*180/pi,1)
cat("angulo =",theta,"radianes que equivales a ",grados," grados\n")
```

```
## angulo = 0.7854 radianes que equivales a  45  grados
```

```
c<-cos(theta)
s<-sin(theta)
P<-matrix(rep(0,4),c(2,2))
P[1,1]<-c
P[2,2]<-c
P[1,2]<--s
P[2,1]<- s
```

Donde, la matriz ϕ :

```
print(P)
```

```
##        [,1]    [,2]
## [1,] 0.7071 -0.7071
## [2,] 0.7071  0.7071
```

y los valores propios λ_i corresponden a la diagonal de la matriz obtenida por el producto matricial $P'AP$:

```
lambda<-round(t(P)%*%A%*%P,2)
print(lambda)
```

```
##      [,1] [,2]
## [1,] 1.65 0.00
## [2,] 0.00 0.35
```

Resultan: $\lambda_1 = 1.65$ y $\lambda_2 = 0.35$

Pregunta 2.

Se dispone de una tabla trigonometrica con dos decimales, correspondiente al $\arctan(x)$ para valores enteros de x=1 a 10. Usted necesita hallar un valor intermedio, exclusivamente para x=4.5. Aplique un método de interpolación con diferencias finitas de orden 2 y otra de orden 3. halle el valor y su correspondiente error. Utilice el menor número de datos de la tabla. $(0.79, 1.11, 1.25, 1.33, 1.37, 1.41, 1.43, 1.45, 1.46, 1.47)$

El valor del arctan(4.5) es:

```
print(atan(4.5))
```

```
## [1] 1.352
```

Se requiere hallar por interpolación; la tabla es:

```
x<-1:10
y<-c(0.79, 1.11, 1.25, 1.33, 1.37, 1.41, 1.43, 1.45, 1.46, 1.47)
print(data.frame(x,y),row.names=FALSE)
```

```
##   x    y
##   1 0.79
##   2 1.11
##   3 1.25
##   4 1.33
##   5 1.37
##   6 1.41
##   7 1.43
##   8 1.45
##   9 1.46
##  10 1.47
```

Para interpolar por diferencias finitas hacia adelante seria suficiente uns subconjunto de 4 fila con sus diferencias de segundo y tercer orden.

```
x<- 4:7
f<-y[x]
n<-length(f)
nombres<-c("f",paste("D",1:(n-1),sep="") )
diff.ad <-rep(NA,n*n)
dim(diff.ad)<-c(n,n)
diff.ad[,1]<-f
dimnames(diff.ad)<-list(0:(n-1),nombres)
for (j in 2:n) {
for (i in 1:(n-j+1)) {
diff.ad[i,j] <- diff.ad[i+1,j-1] - diff.ad[i,j-1]
}
}
tabla<-round(as.matrix(data.frame(x=x,diff.ad)),2)
print(tabla,na.print = "")
```

```
##   x    f   D1    D2    D3
## 0 4 1.33 0.04  0.00 -0.02
```

```
## 1 5 1.37 0.04 -0.02
## 2 6 1.41 0.02
## 3 7 1.43
```

Aplicando interpolación de segundo y tercer orden:

$f(x_0 + \alpha h) = f_0 + \alpha\Delta f_0 + \frac{1}{2}\Delta^2 f_0 + \frac{1}{6}\Delta^3 f_0$

$x_0 = 4$ y $h = 1$ resulta $\alpha = 0.5$

Para segundo orden:

```
f2<-1.33+0.5*0.04+0.5*(0.5-1)*0.0/2
f3<-f2+0.5*(0.5-1)*(0.5-2)*(-0.02)/6
```

$f(4.5) = 1.33 + 0.5 * 0.04 + 0.5 * (0.5 - 1) * 0.0/2$

```
cat("f(4.5) = ",f2,"\n")
```

```
## f(4.5) =  1.35
```

```
cat("Error absoluto =", abs(f2-1.352))
```

```
## Error absoluto = 0.002
```

Para tercer orden

$f(4.5) = 1.33 + 0.5 * 0.04 + 0.5 * (0.5 - 1) * 0.0/2 + 0.5 * (0.5 - 1) * (0.5 - 2) * (-0.02)/6$

```
cat("f(4.5) = ",f3,"\n")
```

```
## f(4.5) =  1.349
```

```
cat("Error absoluto =", abs(f3-1.352))
```

```
## Error absoluto = 0.00325
```

Pregunta 3.

Calcule el error absoluto de la integral I, si utiliza el método de Simpson para $h = 1,\ 2$ y extrapolación.

$$I = \int_1^5 (x^2 - x^{1/3} + 0.3)dx$$

$f(x) = x^2 - x^{1/3} + 0.3$

La integral exacta es: $I = |_1^5 \frac{1}{3}x^3 - \frac{3}{4}X^{4/3} + 0.3x$

```
f<-function(x) x^2-x^{1/3}+0.3
I<-integrate(f,1,5)$value
cat("I = ",I,"\n")
```

```
## I =  36.87
```

Las formulas para h=1 y h=2 aplicado:

```
I1<-1/3 *(f(1)+4*f(2)+f(3))+1/3 *(f(3)+4*f(4)+f(5))
I2<-2/3 *(f(1)+4*f(3)+f(5))
```

$I_{h=1} = \frac{1}{3}(f(1) + 4f(2) + f(3)) + \frac{1}{3}(f(3) + 4f(4) + f(5))$

```
error<-abs(I1-I)
cat("Integral, h=1: ",I1,"\n")
```

```
## Integral, h=1:  36.87
```

```
cat("Error absoluto =", error)
```

```
## Error absoluto = 0.001155
```

$I_{h=2} = \frac{2}{3}(f(1) + 4f(3) + f(5))$

```
error<-abs(I2-I)
cat("Integral, h=2: ",I2,"\n")
```

```
## Integral, h=2:  36.88
```

```
cat("Error absoluto =", error)
```

```
## Error absoluto = 0.00976
```

Pregunta 4.

Dada la ecuación diferencial de primer orden: $Y' = 2X * Y + Y$; con la condición inicial $Y(1) = 0.5$

Si se utiliza Runge-Kutta (m=2) con las constantes $C1 = C2 = 1/2$

Hallar la solución Para $X = 1, 1.5, 2$ para $h = 0.5$

Segun Runge Kutta m:2, el calculo de la ecuación diferencia se determina como:

$Y_{i+1} = Y_i + C_1R_1 + C_2R_2$

$R_1 = hf(x_i, Y_i)$

$R_2 = hf(x_i + ah, Y_i + bR_1)$

Como $C_2 = 0.5$ implica $a = b = 1$

$Y_{i+1} = Y_i + 0.5R_1 + 0.5R_2$

$R_1 = 0.5(2X_i * Y_i + Y_i)$

$R_2 = 0.5(2(X_i + 0.5) * (Y_i + R_1) + (Y_i + R_1))$

```
f<-function(x,y)2*x*y+y
x<-seq(1,2,0.5); y<-NULL;R1<-NULL;R2<-NULL
y[1]<-0.5
for(i in 1:3){
R1[i]<-0.5*f(x[i],y[i])
```

```
R2[i]<-0.5*f(x[i]+0.5,y[i]+R1[i])
y[i+1]<-y[i]+0.5*R1[i]+0.5*R2[i]
}
y<-y[-4]
print(data.frame(x,y,R1,R2),row.names=FALSE)
```

```
##    x      y    R1     R2
##  1.0  0.500  0.75   2.50
##  1.5  2.125  4.25  15.94
##  2.0 12.219 30.55 128.30
```

Pregunta 5.

Considere la siguiente ecuación:

$Y'' = Y' + X * Y - Y$, las condiciones iniciales: $Y(0) = 15$, $Y'(0) = 5$

Hallar $Y(x)$, para $x = 0, 0.5, ..., 2$ con h=0.5

Con las condiciones iniciales y la transformación:

$Z_1 = Y$, $Z_2 = Y'$

Resulta el siguiente sistema de ecuaciones diferenciales de primer orden:

$Z_1' = Z_2$ $Z_2' = Z_2 + X * Z_1 - Z_1$

condiciones iniciales: $Z_1(0) = 15$, $Z_2(0) = 5$

Segun Euler:

$Z_{1,i+1} = Z_{1,i} + h * Z_{2,i}$

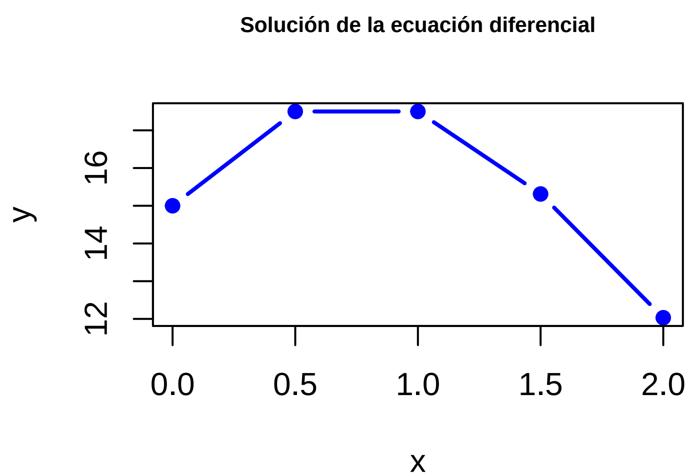
$Z_{2,i+1} = Z_{2,i} + h * (Z_{2,i} + X * Z_{1,i} - Z_{1,i})$

$Y = Z_1$

```
f1 <- function(X,Z1,Z2) Z2
f2 <- function(X,Z1,Z2) Z2+X*Z1 -Z1
# Euler
# Y1+h*f1 -> Y1, i+1
# Y2+h*f2 -> Y2, i+1
x<-NULL # el vector x
z1<-NULL ; z2<-NULL # vectores solución y1 y y2
# valor inicial
x[1] <- 0
z1[1]<- 15
z2[1]<- 5
h<-0.5
j<-1
while (x[j]<2) {
  z1[j+1]<-z1[j]+h*f1(x[j],z1[j],z2[j])
  z2[j+1]<-z2[j]+h*f2(x[j],z1[j],z2[j])
  x[j+1]<-x[j]+h
  j<-j+1
}
y<-z1
solucion<-data.frame(x,z1,z2)
print(solucion,row.names=FALSE)
```

```
##     x    z1     z2
##   0.0 15.00  5.000
##   0.5 17.50  0.000
##   1.0 17.50 -4.375
##   1.5 15.31 -6.562
##   2.0 12.03 -6.016
```

```
plot(x,y,type="b",pch=19,col=c("blue"),cex=0.8,main="Solución de la ecuación diferencial",lwd=2,lty=c(1
text(0.5,40,"Y'' = Y' + XY - Y",cex=0.8)
```



Puntaje 4 c/u