

Parcial, Mayo del 2016

1. Dada la función: $\sqrt{5-x} + 2x^2 + 2x - 10 = 0$

(a) Hallar una aplicación contraída $g(x)$ en el intervalo $(1,2)$ para hallar el punto fijo.

```
> x<-NULL
> g<-function(x)sqrt(5-x-sqrt(5-x))/2
> x[1]<-1.5
> x[2]<-g(x[1])
> x[3]<-g(x[2])
> x[4]<-g(x[3])
> x[5]<-g(x[4])
> print(x)
[1] 1.500000 1.601432 1.573788 1.581364 1.579291
> cat("\nPuntos que pertenecen al intervalo I(1,2)")
Puntos que pertenecen al intervalo I(1,2)
```

(b) Probar que cumple la condición de Lipschitz.

```
> cat("\nCondicion de Lipschitz\n")
Condicion de Lipschitz
> abs(g(x[1])-g(x[2])) ; abs(x[1]-x[2])
[1] 0.02764478
[1] 0.1014324
> k <- abs(g(x[1])-g(x[2])) / abs(x[1]-x[2])
> print(k)
[1] 0.2725439
> cat("\nnuevos puntos\n")
nuevos puntos
> abs(g(x[2])-g(x[3])) ; abs(x[2]-x[3])
[1] 0.007576011
[1] 0.02764478
> k <- abs(g(x[2])-g(x[3])) / abs(x[2]-x[3])
> print(k)
[1] 0.2740485
> cat("\nnuevos puntos\n")
nuevos puntos
> abs(g(x[3])-g(x[4])) ; abs(x[3]-x[4])
[1] 0.002073056
```

```

[1] 0.007576011
> k <- abs(g(x[3])-g(x[4])) / abs(x[3]-x[4])
> print(k)
[1] 0.2736343
> cat("\nCumple la condicion de Lipschitz\n")
Cumple la condicion de Lipschitz

```

(c) Hallar el punto fijo mediante el algoritmo $X_{i+1} = g(X_i)$

```
> cat("\nAplicando el algoritmo\n")
```

Aplicando el algoritmo

```

> for (i in 1:5){
+   x[i+1]<-g(x[i])
+ }
> cbind(x)

```

```

      x
[1,] 1.500000
[2,] 1.601432
[3,] 1.573788
[4,] 1.581364
[5,] 1.579291
[6,] 1.579858

```

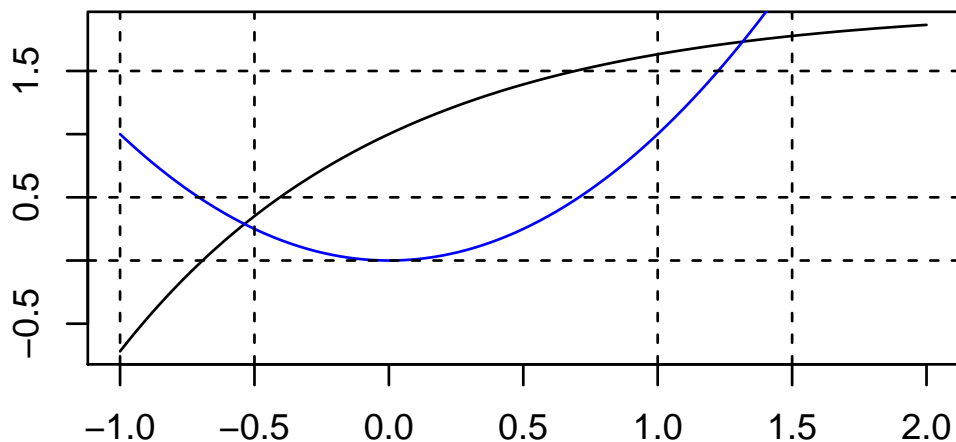
2. Considere el siguiente sistema de ecuaciones: $\exp(-x) + y - 2 = 0$ y la ecuacion $y - x^2 = 0$. Utilice el método aproximacion sucesiva

(a) Localizar graficamente las raices.

```

> par(mar=c(2,2,2,2),cex=0.8)
> f<-function(x)2-exp(-x)
> g<-function(x)x^2
> curve(f,from=-1,to=2)
> curve(g,from=-1,to=2,add=TRUE,col="blue")
> abline(v=c(-1,-0.5),h=c(0,0.5),lty=2)
> abline(v=c(1,1.5),h=c(1.5,2),lty=2)

```



Las raices estan en las regiones:

$$R_1 (-1,-0.5),(0,0.5)$$

$$R_2 (1,1.5), (1.5,2)$$

(b) Buscar el algoritmo que satisfaga la condicion de convergencia.

Posibilidad: $x=1, y=2$ cerca a una raiz

$$x = F(x, y) = \sqrt{y}$$

$$y = G(x, y) = 2 - \exp(-x)$$

$$F_x = 0$$

$$F_y = \frac{1}{2\sqrt{y}}$$

$$G_x = \exp(-x)$$

$$G_y = 0$$

Las relaciones que deben satisfacer para la convergencia:

$$|F_x| + |F_y| \leq k$$

$$|G_x| + |G_y| \leq k$$

$$|F_y| \leq k$$

$$|G_x| \leq k$$

Para valores $0 < k < 1$ para todos los valores en la vecindad R

```
> F<-function(x) sqrt(y)
> G<-function(x) 2-exp(-x)
> Fx<-0
> Fy<-function(y) 0.5/sqrt(y)
> Gx<-function(x) exp(-x)
> Gy<-0
> x<-NULL; y<-NULL
> x[1]<- 1
> y[1]<- 2
> # primera iteracion
> x[2]<-sqrt(y[1])
> y[2]<-2-exp(-x[1])
> # segunda iteracion
> x[3]<-sqrt(y[2])
> y[3]<-2-exp(-x[2])
> # tercera iteracion
> x[4]<-sqrt(y[3])
> y[4]<-2-exp(-x[3])
> cat("\nEstan en la region de interes y satisfacen:\n")
```

Estan en la region de interes y satisfacen:

```
> abs(Fy(y))
[1] 0.3535534 0.3913757 0.3772233 0.3811047
> abs(Gx(x))
[1] 0.3678794 0.2431167 0.2787208 0.2656768
> cat("\nSon valores entre 0 y 1:\n")
```

Son valores entre 0 y 1:

(c) Error relativo de cada aproximación.

```
> # Asumiendo 3 iteraciones
> Ex<-abs(x[4]- x[3])/abs(x[4])
> Ey<-abs(y[4]- y[3])/abs(y[4])
> cat("\nError absoluto para x, y es:",Ex,",",Ey)

Error absoluto para x, y es: 0.03616062 , 0.02068465
```

3. Indicar las condiciones para que un sistema de ecuaciones lineales tenga o no solución en todos los casos. Poner ejemplos de 2 ecuaciones con 2 incógnitas para ilustrar cada caso.

Para que un sistema $Ax = b$ tenga solución, el rango de la matriz aumentada $|Ab|$ debe ser igual al rango de A.

$$\rho(A \ b) = \rho(A)$$

si es diferente no tiene solución.

Ejemplos:

(a) $\rho(A \ b) = \rho(A) = 3$

$$A = \begin{pmatrix} 2 & 4 & 2 \\ 2 & 1 & 3 \\ 4 & 1 & 2 \end{pmatrix} \quad b = \begin{pmatrix} 14 \\ 14 \\ 15 \end{pmatrix}$$

```
> A<-rbind(c(2,4,2),c(2,1,3),c(4,1,2))
> b<-c(14,14,15)
> solve(A,b)

[1] 2 1 3
```

(b) $\rho(A \ b) = 3 \neq \rho(A) = 2$

$$A = \begin{pmatrix} 2 & 4 & 2 \\ 2 & 1 & 3 \\ 4 & 5 & 5 \end{pmatrix} \quad b = \begin{pmatrix} 14 \\ 14 \\ 15 \end{pmatrix}$$

```
> A<-rbind(c(2,4,2),c(2,1,3),c(4,5,5))
> b<-c(14,14,15)
> #solve(A,b)

> solve(A,b)
```

```
Error in solve.default(A, b) :
Lapack routine dgesv: system is exactly singular: U[3,3] = 0
```

4. Descomponer algebraicamente la matriz A en LU, dado por:

$$A = \begin{pmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad L = \begin{pmatrix} 1 & 0 & 0 \\ x & 1 & 0 \\ y & z & 1 \end{pmatrix} \quad U = \begin{pmatrix} a & b & 0 \\ 0 & u & v \\ 0 & 0 & w \end{pmatrix}$$

$$ax = c, \text{ entonces } x = \frac{c}{a}$$

$$bx + u = d, \text{ entonces } u = d - \frac{bc}{a}$$

$$v = 0, y = 0, z = 0, \text{ entonces } w = 1$$

Por lo tanto.

$$L = \begin{pmatrix} 1 & 0 & 0 \\ c/a & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad U = \begin{pmatrix} a & b & 0 \\ 0 & d - bc/a & v \\ 0 & 0 & 1 \end{pmatrix}$$

5. Resolver el siguiente sistemas de ecuaciones $Ax=b$, dado por:

$$A = \begin{pmatrix} 1 & 4 \\ 2 & 0 \\ 4 & 1 \\ 3 & 1 \end{pmatrix} \quad b = \begin{pmatrix} 4 \\ 2 \\ 1 \\ 4 \end{pmatrix}$$

```
> A<-rbind(c(1,4),c(2,0),c(4,1),c(3,1))
> b<-c(4,2,1,4)
> print(data.frame(A,b),row.names=FALSE)

X1 X2 b
  1  4  4
  2  0  2
  4  1  1
  3  1  4

> # La transpuesta de A
> tA<-t(A)
> # construyendo la relacion optima
> # tA%%A = tA%%b
> AA<-tA%%A
> Ab<-tA%%b
> print(data.frame(AA,Ab),row.names=FALSE)

X1 X2 Ab
30 11 24
11 18 21

> #Resolviendo el sistema
> solve(AA,Ab)

      [,1]
[1,] 0.4797136
[2,] 0.8735084
```

Puntaje 6,8,2,2,2