

Taller Repaso  
Análisis Numérico

Por:  
Nicolai Barrera  
Diego Burgos

Para:  
Profesora Eddy Herrera Daza

1. Suponga que un dispositivo solo puede almacenar únicamente los cuatro primeros dígitos decimales de cada número real, y trunca los restantes (esto es redondeo inferior). Calcule el error de redondeo si se quiere almacenar el número 536.78

```
redondear <- function (numero, dig){ #Recibe como parametro Numero a redondear y digitos deseados
  contador=0
  while(numero>1){ #divide y cuenta cuantas divisiones
    numero<-numero/10
    contador=contador+1
  }

  E<-(numero-trunc(numero*10^4)/10^4) # obtiene el numero redondeado

  izq<-1*10^(contador-dig)
  der<-1*10^(contador-dig)
  cat("El error es de ",izq,"<",E,"Y",E,"<",der)
}
redondear(536.78,4)
```

```
## El error es de 0.1 < 8e-05 Y 8e-05 < 0.1
```

2. Implemente en cualquier lenguaje el siguiente algoritmo que sirve para calcular la raíz cuadrada. Aplíquelo para evaluar la raíz cuadrada de 7, analice su precisión, como podría evaluar la convergencia y validez del algoritmo.

```
raiz <- function(numero, valor, tolerancia){ #La funcion recibe el numero a calcular
                                             #junto con un valor inicial y el error que se puede permitir
  resul<-((0.5)*(valor+(numero/valor))) #valor aproximado del calculo raiz de 7

  while(abs(valor-resul) > tolerancia) #calculando el valor hasta que no exeda el error permitido
  {
    valor<- resul
    resul<- (0.5)*(valor+(numero/valor))
  }
  return(cat("La raiz da:", resul, ", el error permitido es de:", tolerancia))
}

raiz(7,3,0.000001)
```

```
## La raiz da: 2.645751 , el error permitido es de: 1e-06
```

3. Utilizando el teorema de Taylor hallar la aproximación de  $e^{(0.5)}$  con cinco cifras significativas

```
PolinomioTaylor <- function(exponente,cifras){ #la funcion recibe un exponente y las cifras que quiero mostrar (
5)
  iterador<-cifras-1
  sumaTotal <-1

  while(iterador>0){
    sumaTotal <- 1+(exponente*sumaTotal)/iterador
    iterador<-iterador-1
  }
  resultado <- signif(sumaTotal,digits=cifras) # (signif)acota el resultado a la cantidad de cifras que necesita
  cat("La aproximacion de e^0.5 es:",resultado)
}

PolinomioTaylor(0.5,5)
```

```
## La aproximacion de e^0.5 es: 1.6484
```

4. Calcule el error de la solución de el siguiente problema: La velocidad de una partícula es constante e igual a 4 m/s, media con un error de 0.1 m/s durante el tiempo de recorrido de 5 seg. Medido con un error de 0.1 seg. Determinemos el error absoluto y el error relativo en el valor de la distancia recorrida  $d = vt$

```
DistanciaError<- function (velocidad,errorVel,tiempo,errorTiem) #recibe los datos para calcular distancia y sus r
espectivos errores
{
  distancia<- velocidad*tiempo #calcula la distancia
  errorRela = (errorVel/velocidad)+(errorTiem/tiempo) #calcula error relativo
  errorAbso = (velocidad*errorVel)+(tiempo*errorTiem) #calcula error absoluto
  cat("La distancia calculada es:" ,distancia, "\n","Error absoluto: ",errorAbso, " \n Error RELativo: ",errorRe
la)
}

DistanciaError(4,0.1,5,0.1)
```

```
## La distancia calculada es: 20
## Error absoluto: 0.9
## Error RELativo: 0.045
```

5. Evaluar el valor de un polinomio es una tarea que involucra para la máquina realizar un número de operaciones la cual debe ser mínimas. Como se puede evaluar el siguiente polinomio con el número mínimo de multiplicaciones  $P(x)=2x^4-3x^2+3x-4$  en  $x_0=-2$

```
polinomio<- function (x0)
{
  resultado <- (2*x0^4)-(3*x0^2)+(3*x0) - 4 #polinomio dado en el enunciado reemplazado en x
  cat("El resultado de reemplazar X0 fue:",resultado)
}

polinomio(-2)
```

```
## El resultado de reemplazar X0 fue: 10
```

6. Reconstruir la silueta del perrito utilizando la menor cantidad de puntos para reproducir el dibujo del contorno completo del perrito sin bigotes, con la información dada:

```
library(PolynomF)

require(PolynomF)

x=c(1,2,5,6,7.5,8.1,10,13,17.6,20,23.5,24.5,25,26.5,27.5,28,29,30)
y=c(3,3.7,3.9,4.5,5.7,6.69,7.12,6.7,4.45,7,6.1,5.6,5.87,5.15,4.1,4.3,4.1,3) # se definen los puntos que seran gr
aficados en la grafica
```

```

plot(x,y, pch=19, cex=0.5, col = "red", asp=1)
i= 1
Minimos = i
Maximos = i+1
bool = 0
cont = 1

repeat
{
  m = (y[i+1]-y[i])/(x[i+1]-x[i])#
  cont = cont +1
  if (i == 1 && m > 0)
  {
    bool = 1}
  else if(i == 1 && m < 0)
  {
    bool = 0}
  if (m < 0)
  {
    if (bool == 0)
    {
      j = i + 1
      Maximos= j
      i = j
    }
    else
    {

      datx = x[Minimos:Maximos]; daty = y[Minimos:Maximos]
      polyAjuste = poly_calc(datx, daty)
      cat("bool", bool, "Minimos", Minimos, "Maximos", Maximos, "cont", cont-1,"<0\n")
      curve(polyAjuste,from=x[Minimos],to=x[Maximos],add=T, lwd=1,col="blue")
      Minimos = Maximos
      cont = 1
      bool = 0
      i = Maximos
      j = i + 1
      Maximos= j
    }
  }
}
else
{
  if (bool == 0)
  {
    datx = x[Minimos:Maximos]; daty = y[Minimos:Maximos]
    polyAjuste = poly_calc(datx, daty)
    cat("bool", bool, "Minimos", Minimos, "Maximos", Maximos, "cont", cont-1,">0\n")
    curve(polyAjuste,from=x[Minimos],to=x[Maximos],add=T, lwd=1,col="blue")
    Minimos = Maximos
    cont = 1
    bool = 1
    i = Maximos
    j = i + 1
    Maximos= j
  }
  else
  {
    j = i + 1
    Maximos= j
    i = j
  }
}
if (cont == 3)
{
  datx = x[Minimos:Maximos]; daty = y[Minimos:Maximos]
  polyAjuste = poly_calc(datx, daty)
  cat("bool", bool, "Minimos", Minimos, "Maximos", Maximos, "cont", cont-1,"<>0\n")
  curve(polyAjuste,from=x[Minimos],to=x[Maximos],add=T, lwd=1,col="blue")
  Minimos = Maximos
  cont = 1
  i = Maximos
  j = i + 1
  Maximos= j

  if(m < 0)

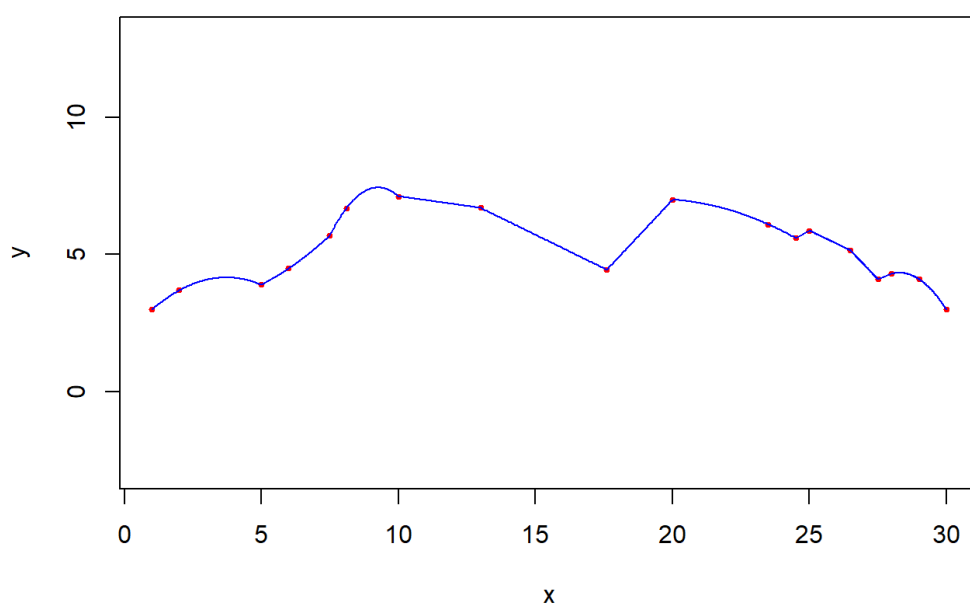
```

```

    {
        bool = 0
    }
    else
    {
        bool = 1
    }
}

if (i==length(x))
{
    Maximos = i
    datx = x[Minimos:Maximos]; daty = y[Minimos:Maximos]
    polyAjuste = poly_calc(datx, daty)
    cat("bool", bool, "Minimos", Minimos, "Maximos", Maximos, "cont", cont-1, "<>0\n")
    curve(polyAjuste, from=x[Minimos], to=x[Maximos], add=T, lwd=1, col="blue")
    break;
}
}

```



```

## bool 1 Minimos 1 Maximos 3 cont 2 <>0
## bool 1 Minimos 3 Maximos 5 cont 2 <>0
## bool 1 Minimos 5 Maximos 7 cont 2 <>0
## bool 1 Minimos 7 Maximos 8 cont 1 <0
## bool 0 Minimos 8 Maximos 9 cont 2 >0
## bool 1 Minimos 9 Maximos 10 cont 2 <0
## bool 0 Minimos 10 Maximos 12 cont 2 <>0
## bool 0 Minimos 12 Maximos 13 cont 1 >0
## bool 1 Minimos 13 Maximos 14 cont 1 <0
## bool 0 Minimos 14 Maximos 15 cont 2 >0
## bool 1 Minimos 15 Maximos 16 cont 2 <0
## bool 0 Minimos 16 Maximos 18 cont 2 <>0
## bool 0 Minimos 18 Maximos 18 cont 0 <>0

```