

# Busqueda de raiz

Agustin Huczok

11/9/2021

#Algoritmos

```
#Biseccion-----
raiz_biseccion<-function(func,a,b,tol,N){
  i<-1
  FA<-func(a)
  while(i<=N){
    p=a+(b-a)/2
    FP=func(p)
    if(FP==0 | (b-a)/2<tol){
      return(p)
    }
    i=i+1
    if(FA*FP>0){
      a=p
      FA=FP
    }
    else
      {b=p}
  }
  return(paste('El metodo fracaso luego de N',N,'intentos'))
}

#Punto fijo-----
iteracion_punto_fijo<-function(func,p0,tol,N){
  i<-1
  while(i<=N){
    p=func(p0)
    if(abs(p-p0) < tol){
      return(p)
    }
    i<-i+1
    p0=p
  }
  paste('El metodo fallo despues de ',N,'iteraciones')
}

#Newton-----
f=function(x){
  f= sin(x)-exp(-x)
}
df=function(x){
  df= cos(x)+exp(x)
}
```

```

}

metodo_newton<-function(func,dfunc,p0,tol,N){
  i=1
  while (i<=N){
    p=p0-func(p0)/dfunc(p0)
    if(abs(p-p0)<tol){
      return(p)
    }
    i=i+1
    p0=p
  }
  paste('El metodo fallo despues de ', N,'iteraciones')
}

#Secante-----
#Necesito primero aproximar p1

f<-function(x){
  f= x^3 - 2*x^2-5
}

metodo_secante<-function(func,p0,p1,tol,N){
  i=2
  q0=func(p0)
  q1=func(p1)
  while(i<=N){
    p=p1-q1*(p1-p0)/(q1-q0)
    if(abs(p-p1)<tol){
      return(p)
    }
    i=i+1
    p0=p1
    q0=q1
    p1=p
    q1=func(p)
  }
  paste('El metodo fallo despues de', N, 'iteraciones')
}

#Posicion falsa-----
#Necesito primero aproximar p1

posicion_falsa<-function(func,p0,p1,tol,N){
  i=2
  q0=func(p0)
  q1=func(p1)
  while(i<=N){
    p=p1-q1*(p1-p0)/(q1-q0)
    if(abs(p-p1)<tol){
      return(p)
    }
    i=i+1
  }

```

```

q=func(p)
if(q*q1<0){
  p0=p
  q1=q
}
p1=p
q1=q
}
}

```

#Metodo de Biseccion

##Ejercicio 1

Hallar la solucion de:  $\cos(x) = \sqrt{x}$  Defino la funcion como  $f(x) = 0$ :  $\cos(x) - \sqrt{x} = 0$  Grafico la funcion para observar el intervalo donde se encuentra la raiz

```

f <- function(x) {
  f = cos(x)- x^(0.5)
}

x<-seq(0,10,0.1)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

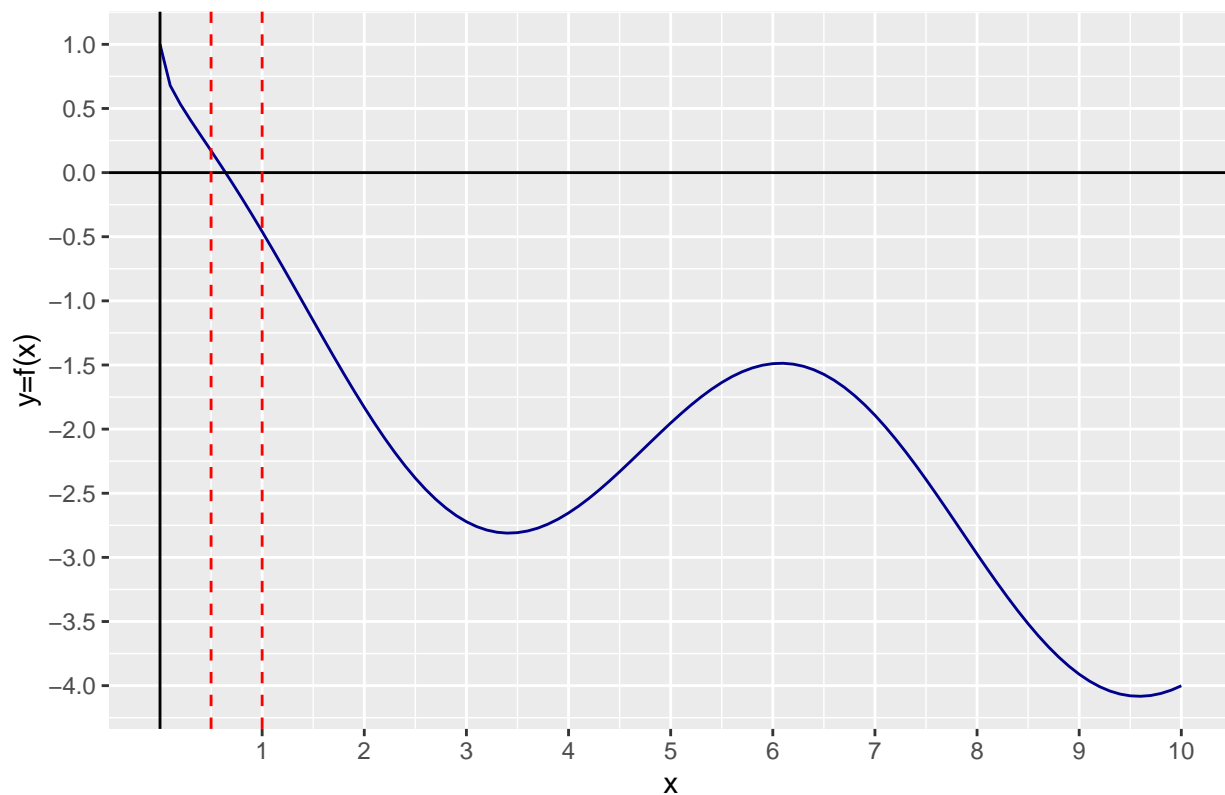
ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(1,10,1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-4.5,1,0.5)) #Cambio escala eje Y
ggfx=ggfx+ggtitle("Funcion con raiz entre 0.5 y 1") #Agrego titulo

#Ahora se que la raiz esta entre 0.5 y 1. Grafico puntos en ese area
ggfx=ggfx+geom_vline(xintercept=c(0.5,1),linetype=2,colour="red")

ggfx

```

## Funcion con raiz entre 0.5 y 1



Aplico metodo de biseccion

```
raiz_biseccion(f,0.5,1,0.0001,100)
```

```
## [1] 0.6416626
```

La ecuacion  $\cos(x) = \sqrt{x}$  tiene raiz en 0.6416626

##Ejercicio 2

Hallar la solucion de:  $x^3 + 4x^2 - 10 = 0$

Grafico la funcion para observar el intervalo donde se encuentra la raiz

```
f=function(x){
  f=x^3+4*x^2-10
}

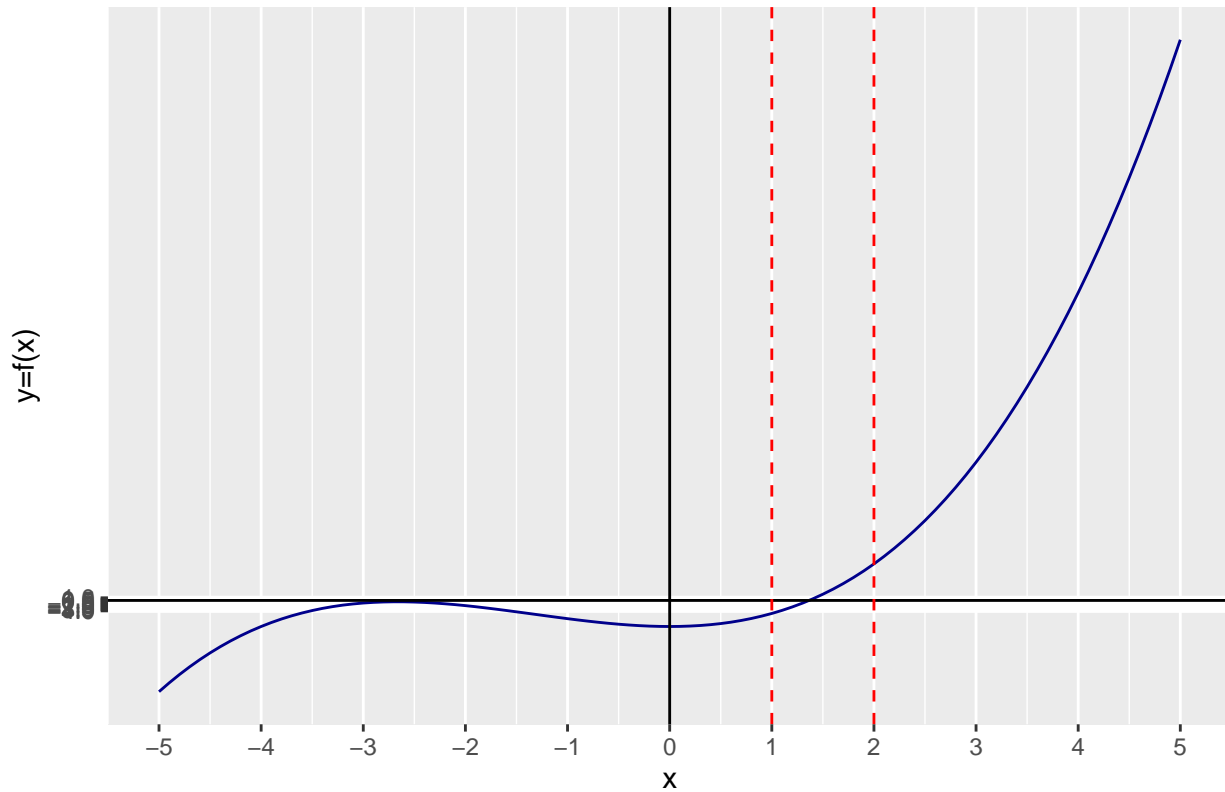
x<-seq(-5,5,0.1)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe
ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-5,5,1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-4,1,0.5)) #Cambio escala eje Y
```

```
ggfx=ggfx+ggtitle("Funcion con raiz entre 1 y 2") #Agrego titulo

#Ahora se que la raiz esta entre 1 y 2. Grafico puntos en ese area
ggfx=ggfx+geom_vline(xintercept=c(1,2),linetype=2,colour="red")

ggfx
```

Funcion con raiz entre 1 y 2



Aplico metodo de biseccion

```
raiz_biseccion(f,1,2,0.0001,100)
```

```
## [1] 1.365173
```

La funcion  $x^3 + 4x^2 - 10 = 0$  tiene raiz en 1.365173

##Ejercicio 3 Hallar la solucion de:  $2 + \cos(e^x - 2) - e^x = 0$

Grafico la funcion para observar el intervalo donde se encuentra la raiz

```
f=function(x){
  f=2+cos(exp(x)-2)-exp(x)
}

x<-seq(-1,2,0.1)#Genero vector para graficar f(x)
fx<-f(x)
```

```
df<-data.frame(x,fx) #Creo dataframe
ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-10,10,1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-10,10,1)) #Cambio escala eje Y
ggfx=ggfx+ggtitle("Funcion con raiz entre 0.5 y 2") #Agrego titulo

#Ahora se que la raiz esta entre 0.5 y 2. Grafico puntos en ese area
ggfx=ggfx+geom_vline(xintercept=c(0.5,2),linetype=2,colour="red")

ggfx
```



Aplico metodo de biseccion

```
round(raiz_biseccion(f,0.5,2,0.0001,100),8)
```

```
## [1] 1.00766
```

La ecuacion  $2 + \cos(e^x - 2) - e^x = 0$  tiene raiz en 1.00766

##Ejercicio 4

Hallar la solucion de:  $x^3 - 7x^2 + 14x - 6 = 0$

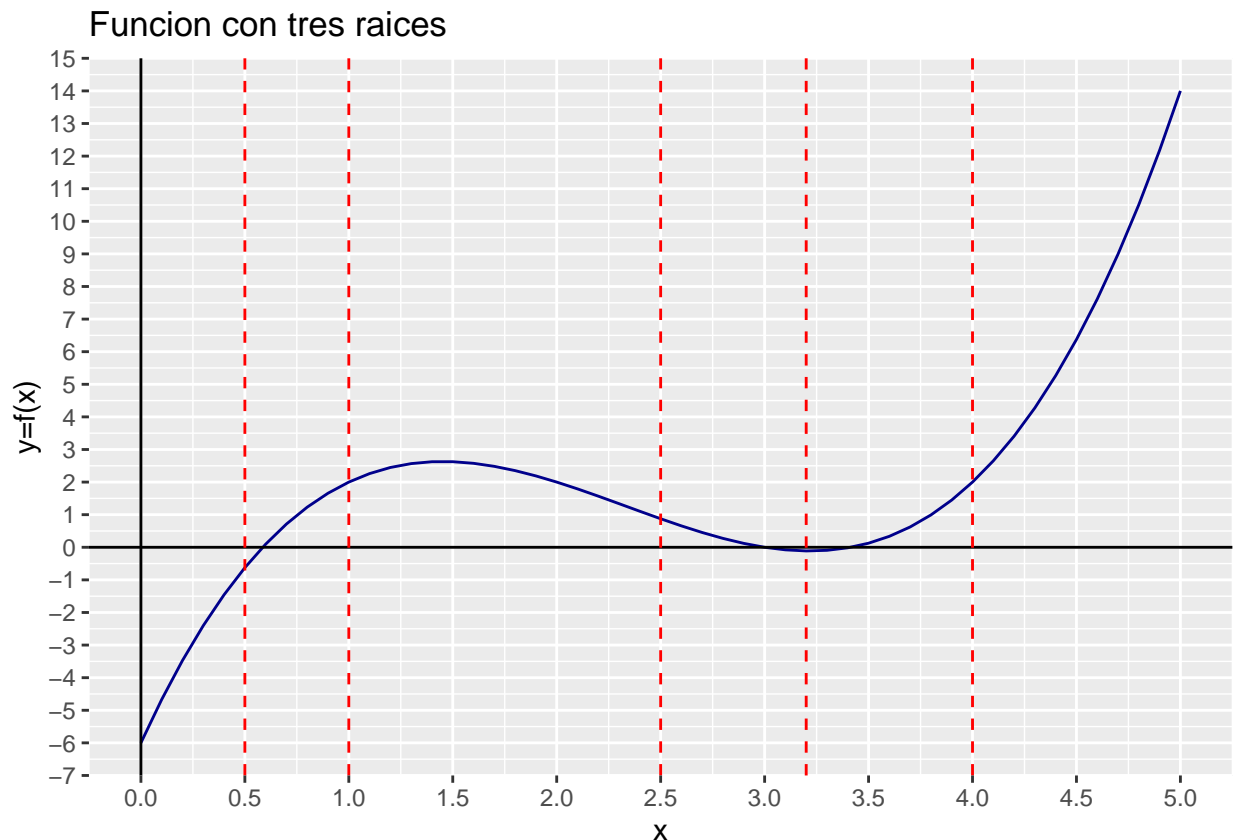
Grafico la funcion para observar el intervalo donde se encuentra la raiz

```
f=function(x){
  f=x^3-7*x^2+14*x-6
}

x<-seq(0,5,0.1)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe
ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,10,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-8,15,1)) #Cambio escala eje Y
ggfx=ggfx+ggtitle("Funcion con tres raices") #Agrego titulo

#Ahora se que una raiz esta entre 0.5 y 1 , otra entre 2.5 y 3.2 y otra entre 3.2 y 4. Grafico puntos e
ggfx=ggfx+geom_vline(xintercept=c(0.5,1,2.5,3.2,4),linetype=2,colour="red")

ggfx
```



Aplico metodo de biseccion

```
raiz_biseccion(f,0.5,1,0.0001,100)
```

```
## [1] 0.5857544
```

```
raiz_biseccion(f,2.5,3.2,0.0001,100)
```

```
## [1] 2.999963
```

```
raiz_biseccion(f,3.2,4,0.0001,100)
```

```
## [1] 3.41416
```

Las raices de la ecuacion  $x^3 - 7x^2 + 14x - 6 = 0$  se encuentran en 0.5857544, 2.999963 y en 3.41416

#Metodo punto fijo

##Ejercicio 1

```
f=function(x){  
  x-x^3 -4*x^2+10  
}
```

Grafico

```
x<-seq(-4,1.8,0.001)#Genero vector para graficar f(x)
```

```
fx<-f(x)
```

```
df<-data.frame(x,fx) #Creo dataframe
```

```
ggfx=ggplot(data=df) #carga los datos
```

```
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
```

```
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
```

```
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
```

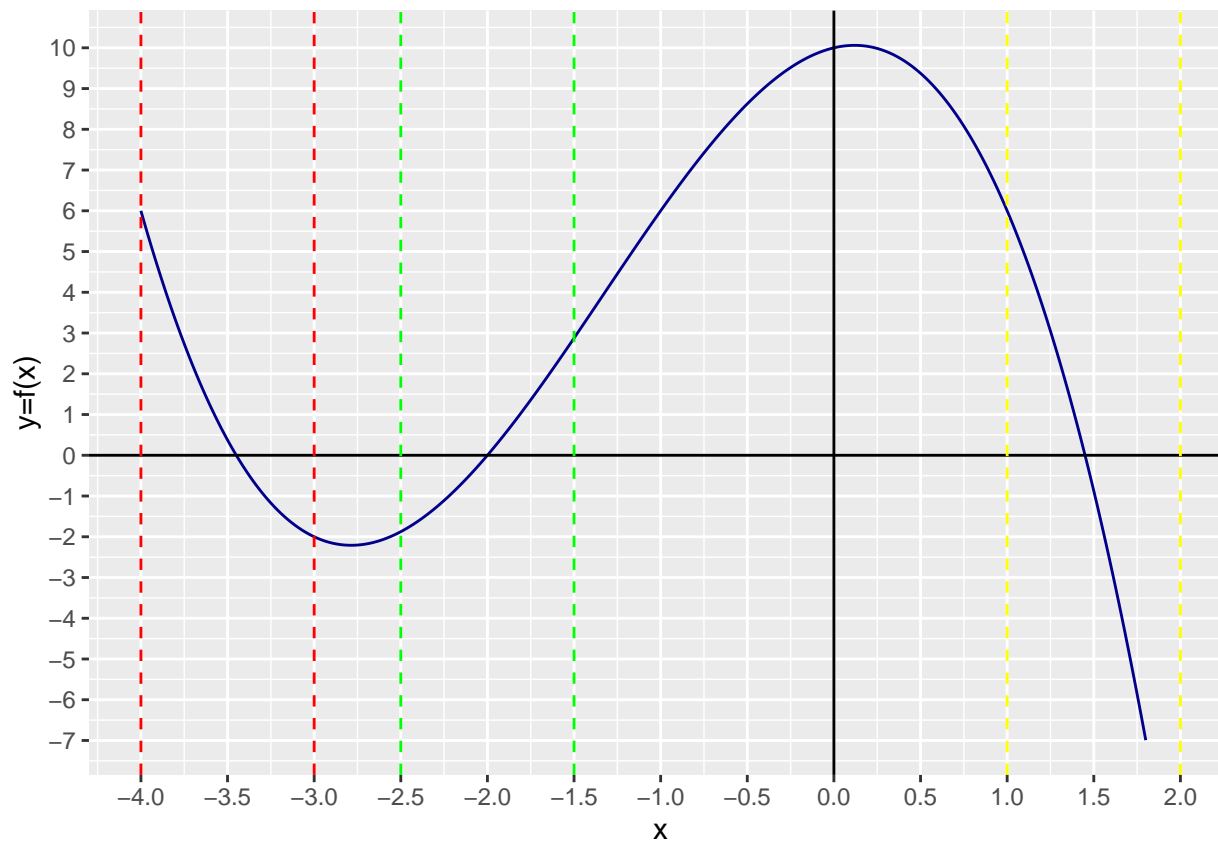
```
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-4,3,0.5)) #cambio escala eje X
```

```
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-10,10,1)) #Cambio escala eje Y
```

```
ggfx=ggfx+geom_vline(xintercept=c(-4,-3),linetype=2,colour="red")+geom_vline(xintercept = c(-2.5,-1.5),
```

```
ggfx
```





Aplico el metodo de punto fijo en todos los intervalos

```
puede_fallar<- tryCatch(print("Las raices se encuentran en: ",iteracion_punto_fijo(f,-3.5,0.0001,100),"
```

```
## [1] "Error"
```

No se pueden obtener las raices mediante este metodo

##Ejercicio 2

```
f=function(x){
  sqrt((10/x)-4*x)
}
```

grafico

```
x<-seq(-4,2,0.001)#Genero vector para graficar f(x)
fx<-f(x)
```

```
## Warning in sqrt((10/x) - 4 * x): Se han producido NaNs
```

```
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
```

```

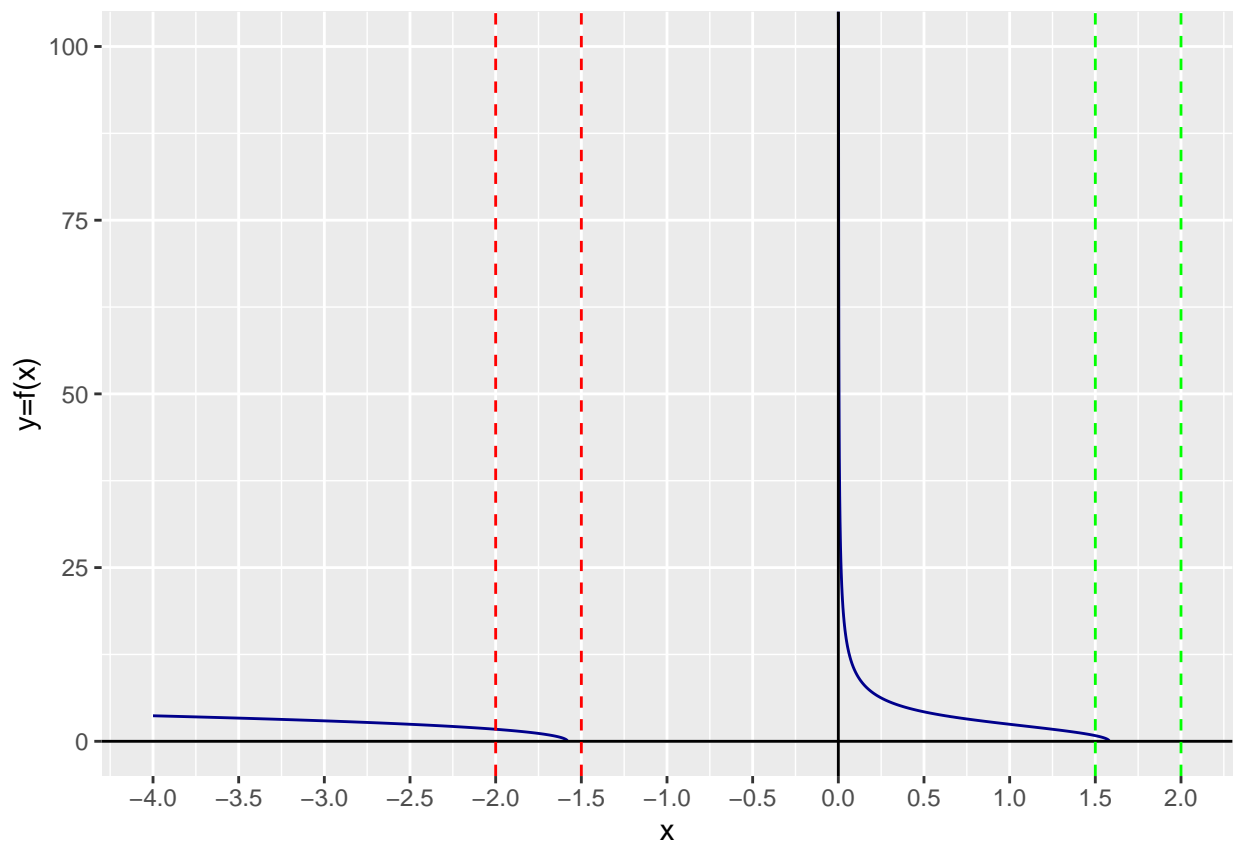
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-4,2,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(0,100,25)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(-2,-1.5),linetype=2,colour="red")+geom_vline(xintercept = c(1.5,2),linetype=2,colour="green")

ggfx

```

```
## Warning: Removed 419 row(s) containing missing values (geom_path).
```



Aplico el metodo de punto fijo en todos los intervalos

```

puede_fallar<- tryCatch(print("Las raices se encuentran en: ",iteracion_punto_fijo(f,-1.75,0.0001,100),
  ", y en : ",iteracion_punto_fijo(f,1.5,0.0001,100)),error = function(e){print("Error")})

```

```
## Warning in sqrt((10/x) - 4 * x): Se han producido NaNs
```

```
## [1] "Error"
```

La funcion no cumple con la condicion de ser continua

##Ejercicio 3

```
f=function(x){
  0.5*(10-x^3)^0.5
}
g=function(x){
  -x+0.5*(10-x^3)^0.5
}
```

grafico f y g

```
x <- seq(0, 3, by = 0.0001) #Genero vector para graficar
fx <- f(x)
df <- data.frame(x, fx) #Creo dataframe
gggx <- ggplot(data = df) #Cargo los datos
gggx <- gggx + aes(x = x, y = fx) #Cargo las variables
gggx <- gggx + geom_line(linetype = 1, colour = "darkblue")
#Gráfico x = y
gggx <- gggx + geom_line(aes(y = x),linetype=4, colour = "black")
#Gráfico la función del ejercicio donde esta la raiz
gggx <- gggx + geom_line(aes(x = x, y = g(x)), colour = "steelblue")
#Agrego el eje X e Y
gggx <- gggx + geom_vline(xintercept = 0, linetype = 1)+
  geom_hline(yintercept = 0, linetype = 1)
#Mapeo las escalas
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,3,0.5)) +scale_y_continuous(name="y=f(x)",breaks=seq
```

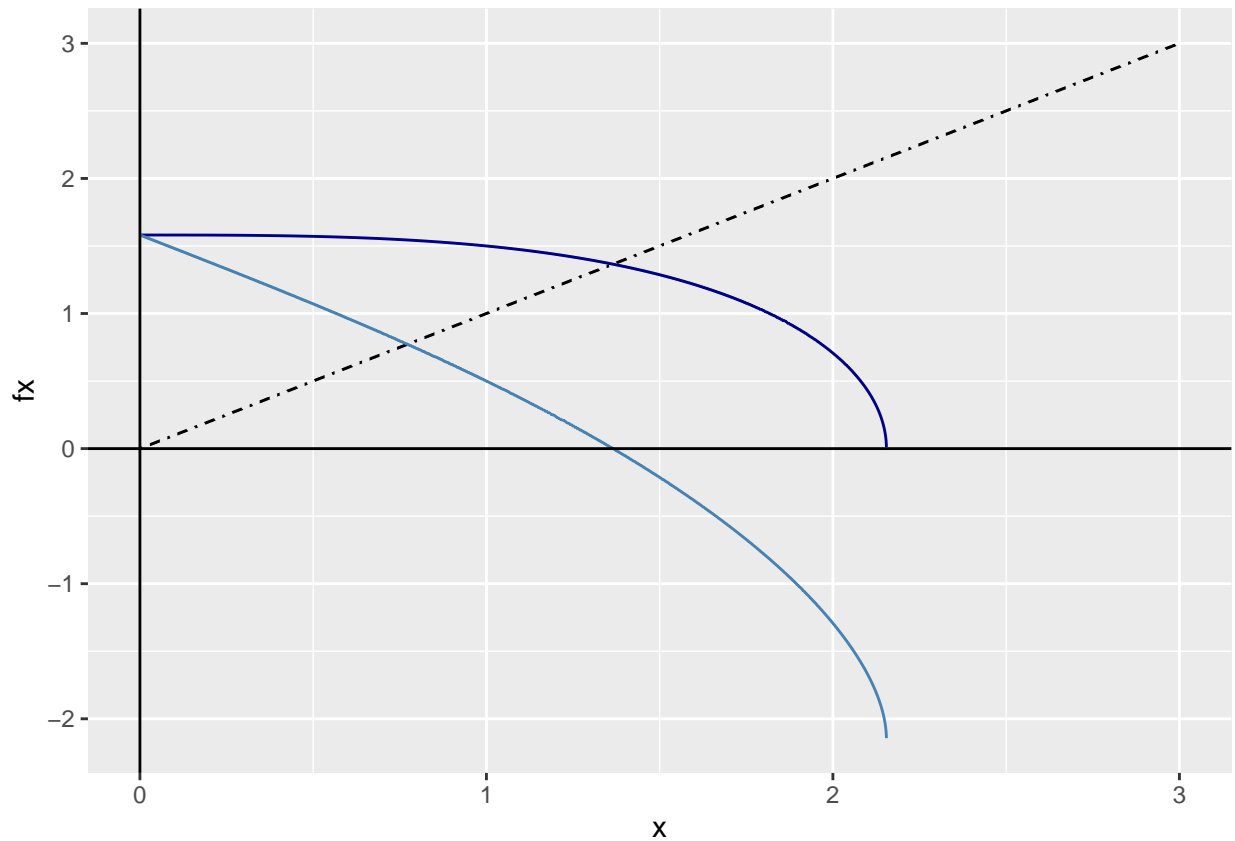
```
## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.
```

```
## Scale for 'y' is already present. Adding another scale for 'y', which will
## replace the existing scale.
```

```
gggx
```

```
## Warning: Removed 8456 row(s) containing missing values (geom_path).
```

```
## Warning: Removed 8456 row(s) containing missing values (geom_path).
```



Calculo el punto fijo y verifico graficamente que coincida con la raiz

```
iteracion_punto_fijo(f,1.5,0.0001,100)
```

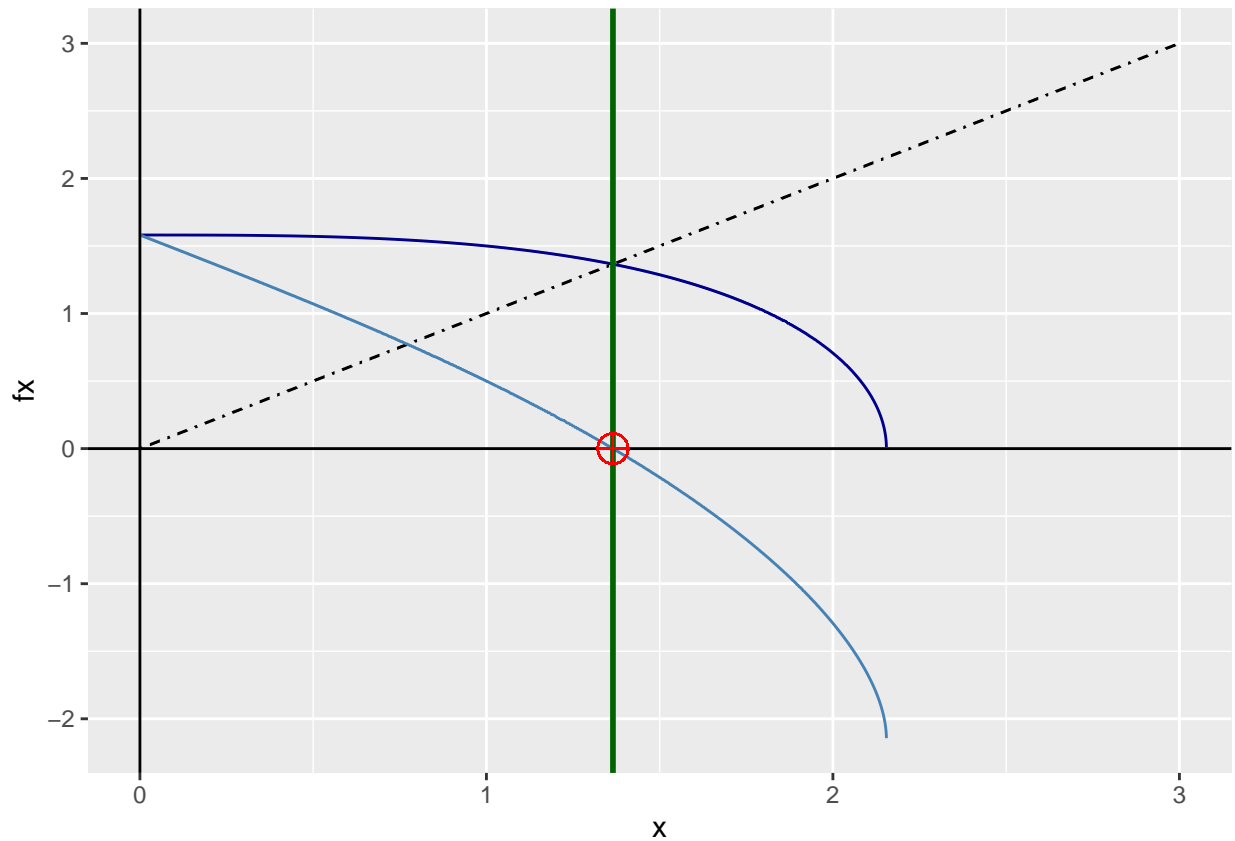
```
## [1] 1.365206
```

```
gggx=gggx+geom_vline(xintercept=iteracion_punto_fijo(f,1.5,0.0001,100),
                      linetype=1,size=1,colour="darkgreen")+
  geom_point(aes(x=iteracion_punto_fijo(f,1.5,0.0001,100),
                  y=g(iteracion_punto_fijo(f,1.5,0.0001,100))),pch=10,size=5,col="red")

gggx
```

```
## Warning: Removed 8456 row(s) containing missing values (geom_path).
```

```
## Warning: Removed 8456 row(s) containing missing values (geom_path).
```



```
paste("La raiz se encuentra en ", iteracion_punto_fijo(f,1.5,0.0001,100))
```

```
## [1] "La raiz se encuentra en 1.36520585029705"
```

```
paste("La raiz se encuentra en ", iteracion_punto_fijo(f,1.5,0.0001,100))
```

```
## [1] "La raiz se encuentra en 1.36520585029705"
```

##Ejercicio 4

```
#La función del ejercicio
f <- function(x){
  return((10/(4+x))^(1/2))
}
#La función para graficar la raiz
g <- function(x){
  return(-x+(10/(4+x))^(1/2))
}
```

Grafico  $f(x)$  y  $g(x)$

```
graphics.off()
x <- seq(0, 2, by = 0.0001) #Genero vector para graficar
fx <- f(x)
```

```

df <- data.frame(x, fx) #Creo dataframe
gggx <- ggplot(data = df) #Cargo los datos
gggx <- gggx + aes(x = x, y = fx) #Cargo las variables
gggx <- gggx + geom_line(linetype = 1, colour = "darkblue")
#Gráfico x = y
gggx <- gggx + geom_line(aes(y = x),linetype=4, colour = "black")
#Gráfico la función del ejercicio donde esta la raiz
gggx <- gggx + geom_line(aes(x = x, y = g(x)), colour = "steelblue")
#Agrego el eje X e Y
gggx <- gggx + geom_vline(xintercept = 0, linetype = 1)+
  geom_hline(yintercept = 0, linetype = 1)
#Mapeo las escalas
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,3,0.5)) +scale_y_continuous(name="y=f(x)",breaks=seq

```

```

## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.

```

```

## Scale for 'y' is already present. Adding another scale for 'y', which will
## replace the existing scale.

```

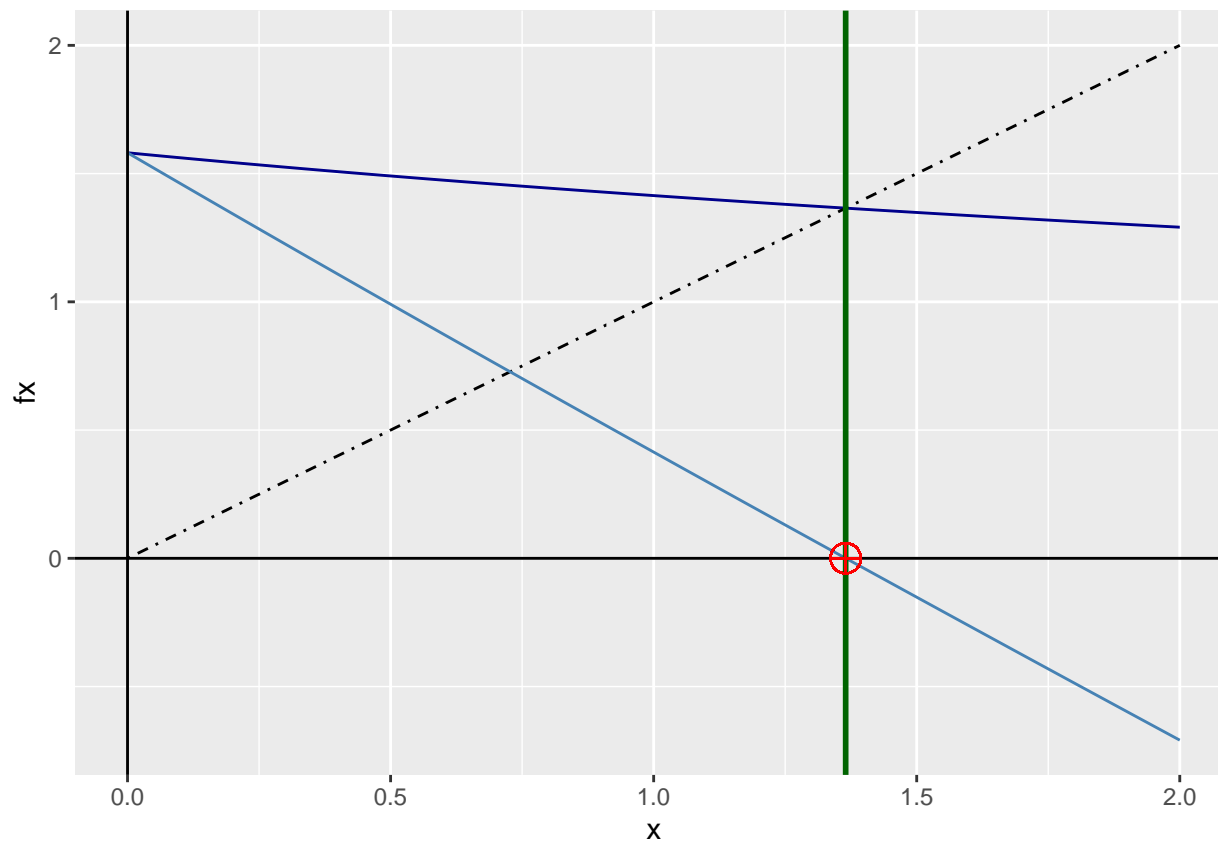
```
gggx
```

Calculo el punto fijo y verifico graficamente que coincida con la raiz

```

gggx=gggx+geom_vline(xintercept=iteracion_punto_fijo(f,1.5,0.01,300),linetype=1,size=1,colour="darkgreen")
gggx

```



```
paste("La raiz se encuentra en ", iteracion_punto_fijo(f,1.5,0.1,100))
```

```
## [1] "La raiz se encuentra en 1.36737637199128"
```

```
##Ejercicio 5
```

```
f=function(x){
  x-(x^3+4*x^2-10)/(3*x^2+8*x)
}
g=function(x){
  -(x^3+4*x^2-10)/(3*x^2+8*x)
}
```

Grafico  $f(x)$  y  $g(x)$

```
graphics.off()
x <- seq(-10, 10, by = 0.1) #Genero vector para graficar
fx <- f(x)
df <- data.frame(x, fx) #Creo dataframe
gggx <- ggplot(data = df) #Cargo los datos
gggx <- gggx + aes(x = x, y = fx) #Cargo las variables
gggx <- gggx + geom_line(linetype = 1, colour = "darkblue")
#Gráfico  $x = y$ 
gggx <- gggx + geom_line(aes(y = x), linetype=4, colour = "black")
```

```
#Gráfico la función del ejercicio donde esta la raíz
gggx <- gggx + geom_line(aes(x = x, y = g(x)), colour = "steelblue")
#Agrego el eje X e Y
gggx <- gggx + geom_vline(xintercept = 0, linetype = 1)+
  geom_hline(yintercept = 0, linetype = 1)
```

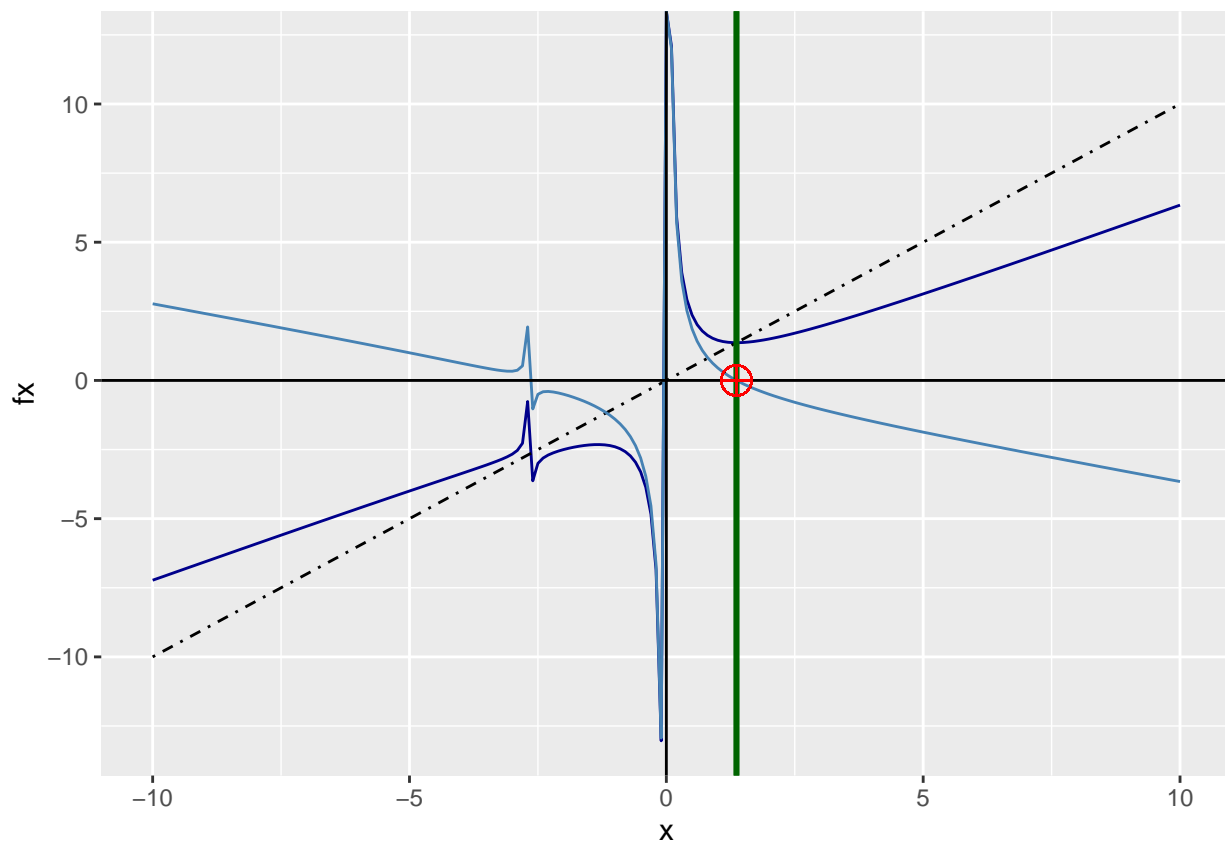
```
gggx
```

Calculo el punto fijo y verifico graficamente que coincida con la raíz

```
gggx=gggx+geom_vline(xintercept=iteracion_punto_fijo(f,-1.5,0.0001,100),
  linetype=1,size=1,colour="darkgreen")+
  geom_point(aes(x=iteracion_punto_fijo(f,-1.5,0.0001,100),
    y=g(iteracion_punto_fijo(f,-1.5,0.0001,100))),pch=10,size=5,col="red")

gggx=gggx+geom_vline(xintercept=iteracion_punto_fijo(f,1.5,0.0001,100),
  linetype=1,size=1,colour="darkgreen")+
  geom_point(aes(x=iteracion_punto_fijo(f,1.5,0.0001,100),
    y=g(iteracion_punto_fijo(f,1.5,0.0001,100))),pch=10,size=5,col="red")
```

```
gggx
```





```
paste("La raices se enecenuntran en ", iteracion_punto_fijo(f,1.5,0.0001,100)," y en: ",iteracion_punto_f
```

```
## [1] "La raices se enecenuntran en 1.36523001391615 y en: 1.36523001342257"
```

##Ejercicio 6

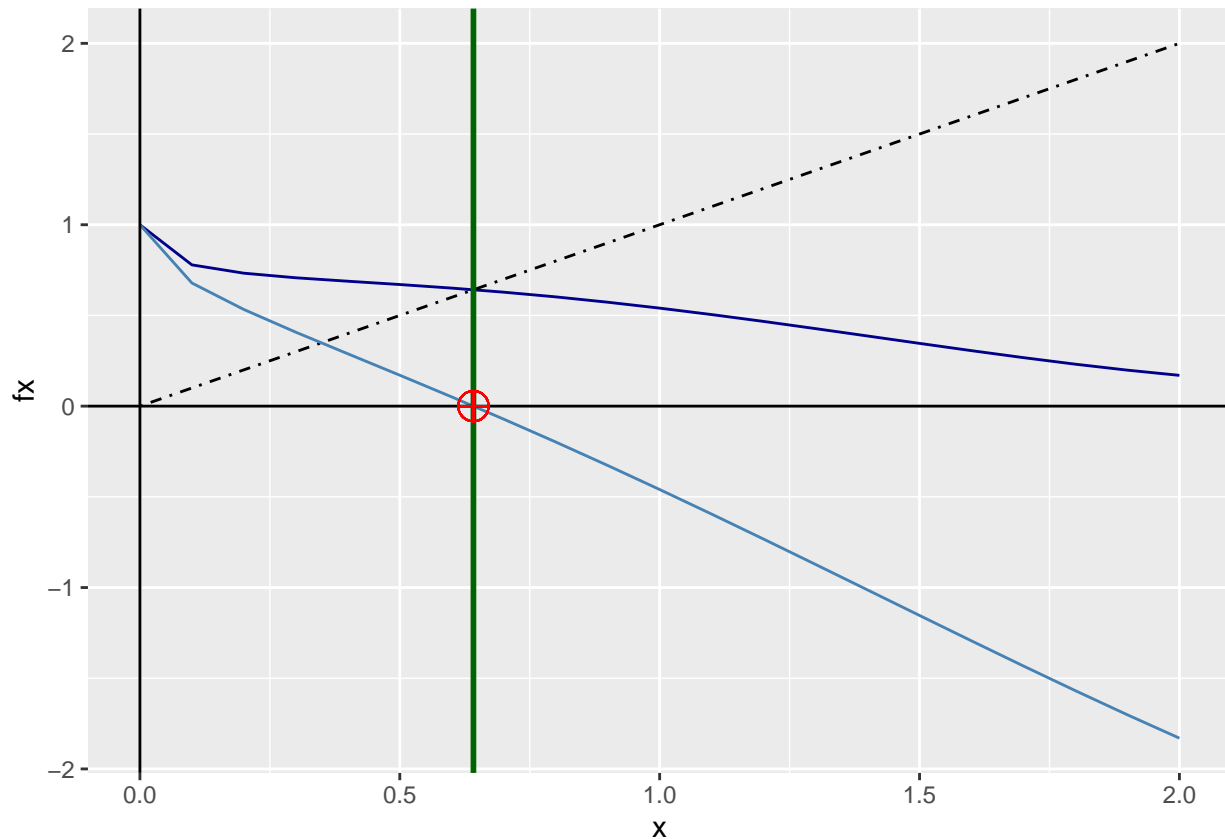
```
f=function(x){  
  -(x)^0.5+cos(x)+x  
}  
g=function(x){  
  -(x)^0.5+cos(x)  
}
```

Grafico

```
graphics.off()  
x <- seq(0, 2, by = 0.1) #Genero vector para graficar  
fx <- f(x)  
df <- data.frame(x, fx) #Creo dataframe  
gggx <- ggplot(data = df) #Cargo los datos  
gggx <- gggx + aes(x = x, y = fx) #Cargo las variables  
gggx <- gggx + geom_line(linetype = 1, colour = "darkblue")  
#Gráfico x = y  
gggx <- gggx + geom_line(aes(y = x),linetype=4, colour = "black")  
#Gráfico la función del ejercicio donde esta la raiz  
gggx <- gggx + geom_line(aes(x = x, y = g(x)), colour = "steelblue")  
#Agrego el eje X e Y  
gggx <- gggx + geom_vline(xintercept = 0, linetype = 1)+  
  geom_hline(yintercept = 0, linetype = 1)  
  
gggx
```

Calculo el punto fijo y verifico graficamente que coincida con la raiz

```
gggx=gggx+geom_vline(xintercept=iteracion_punto_fijo(f,1,0.0001,100),  
  linetype=1,size=1,colour="darkgreen")+  
  geom_point(aes(x=iteracion_punto_fijo(f,1,0.0001,100),  
    y=g(iteracion_punto_fijo(f,1,0.0001,100))),pch=10,size=5,col="red")  
  
gggx
```



```
paste("La raiz se encontre en ", iteracion_punto_fijo(f,1,0.0001,100))
```

```
## [1] "La raiz se encontre en 0.641702668200056"
```

```
##Ejercicio 7
```

```
f=function(x){
  log(2+cos(exp(x)-2))
}
g=function(x){
  -x+log(2+cos(exp(x)-2))
}
```

Grafico

```
graphics.off()
x <- seq(0, 2, by = 0.1) #Genero vector para graficar
fx <- f(x)
df <- data.frame(x, fx) #Creo dataframe
gggx <- ggplot(data = df) #Cargo los datos
gggx <- gggx + aes(x = x, y = fx) #Cargo las variables
gggx <- gggx + geom_line(linetype = 1, colour = "darkblue")
#Grafico x = y
gggx <- gggx + geom_line(aes(y = x), linetype=4, colour = "black")
```

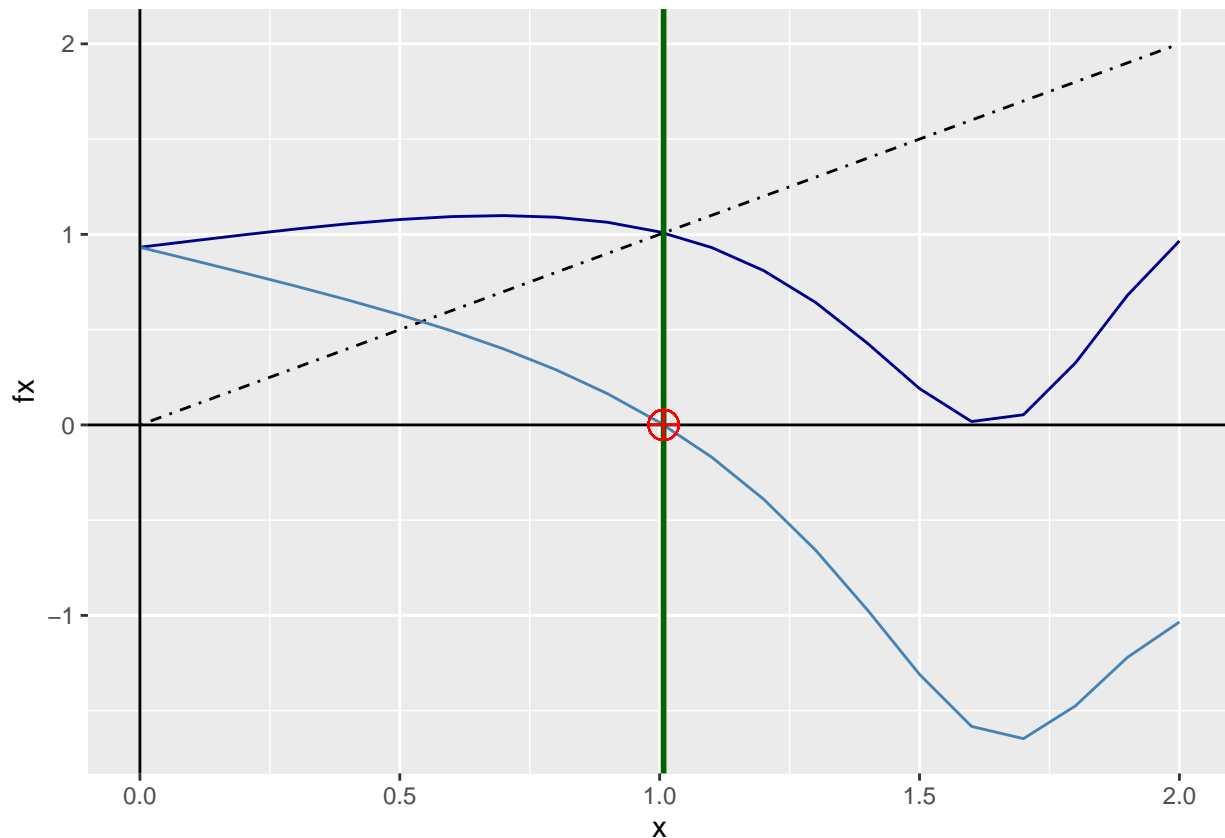
```
#Gráfico la función del ejercicio donde esta la raíz
gggx <- gggx + geom_line(aes(x = x, y = g(x)), colour = "steelblue")
#Agrego el eje X e Y
gggx <- gggx + geom_vline(xintercept = 0, linetype = 1)+
  geom_hline(yintercept = 0, linetype = 1)
```

```
gggx
```

Calculo el punto fijo y verifico graficamente que coincida con la raíz

```
gggx=gggx+geom_vline(xintercept=iteracion_punto_fijo(f,1,0.0001,100),
  linetype=1,size=1,colour="darkgreen")+
  geom_point(aes(x=iteracion_punto_fijo(f,1,0.0001,100),
    y=g(iteracion_punto_fijo(f,1,0.0001,100))),pch=10,size=5,col="red")
```

```
gggx
```



```
paste("La raíz se encuentra en ", iteracion_punto_fijo(f,1,0.0001,100))
```

```
## [1] "La raíz se encuentra en 1.00759410607042"
```

## ##Ejercicio 8

```
f=function(x){
  (x^3-7*x^2-6)/(-14)
}
g=function(x){
  (x^3-7*x^2-6)/(-14)-x
}
```

Grafico

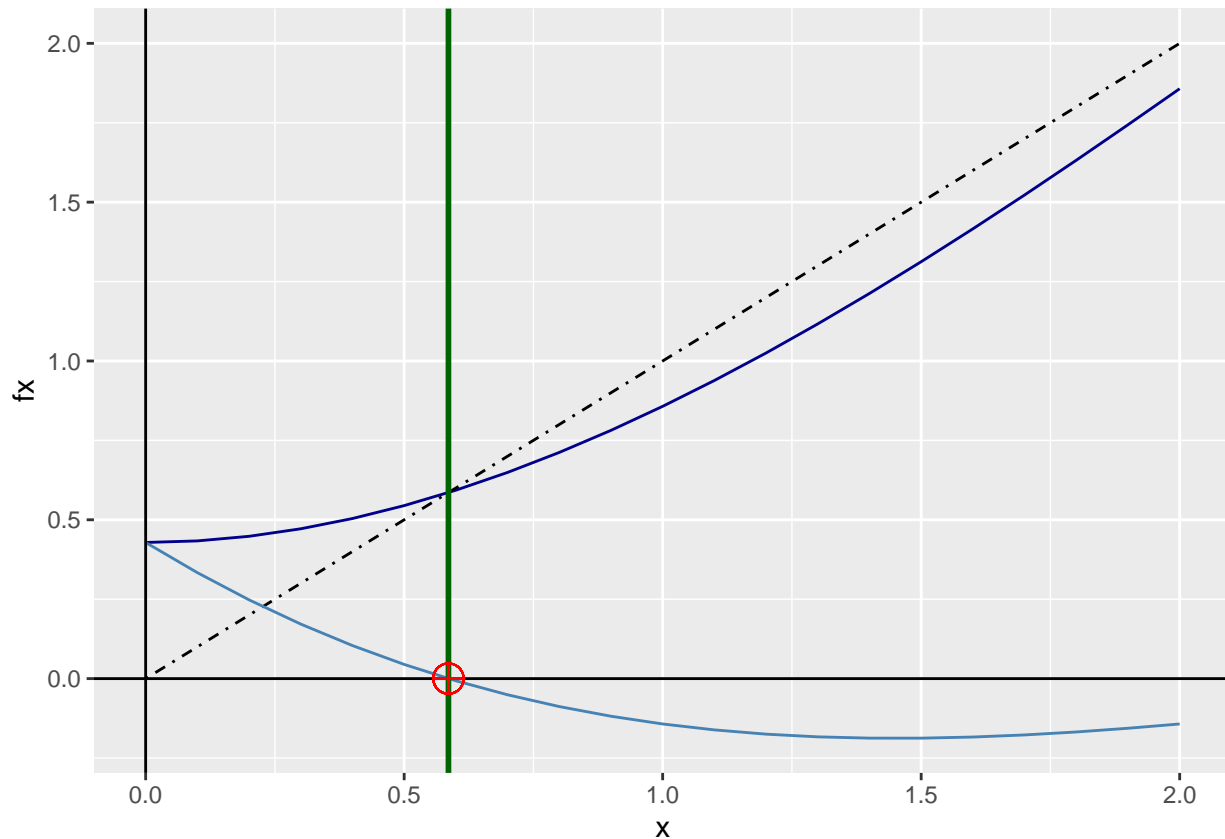
```
graphics.off()
x <- seq(0, 2, by = 0.1) #Genero vector para graficar
fx <- f(x)
df <- data.frame(x, fx) #Creo dataframe
gggx <- ggplot(data = df) #Cargo los datos
gggx <- gggx + aes(x = x, y = fx) #Cargo las variables
gggx <- gggx + geom_line(linetype = 1, colour = "darkblue")
#Gráfico x = y
gggx <- gggx + geom_line(aes(y = x), linetype=4, colour = "black")
#Gráfico la función del ejercicio donde esta la raiz
gggx <- gggx + geom_line(aes(x = x, y = g(x)), colour = "steelblue")
#Agrego el eje X e Y
gggx <- gggx + geom_vline(xintercept = 0, linetype = 1)+
  geom_hline(yintercept = 0, linetype = 1)

gggx
```

Calculo el punto fijo y verifico graficamente que coincida con la raiz

```
gggx=gggx+geom_vline(xintercept=iteracion_punto_fijo(f,0.5,0.0001,100),
  linetype=1,size=1,colour="darkgreen")+
  geom_point(aes(x=iteracion_punto_fijo(f,0.5,0.0001,100),
    y=g(iteracion_punto_fijo(f,0.5,0.0001,100))),pch=10,size=5,col="red")

gggx
```



```
paste("La raiz se encontre en ", iteracion_punto_fijo(f,0.5,0.0001,100))
```

```
## [1] "La raiz se encontre en 0.5856924422807"
```

#Metodo Newton-Raphson

##Ejercicio 1 Hallar la raiz para la funcion  $e^x + 2^{-x} + 2\cos(x) - 6 = 0$  en  $[1; 2]$  Primero busco la derivada de la funcion

```
f=function(x){
  exp(x)+2^(-x)+2*cos(x)-6
}
fe=expression(exp(x)+2^(-x)+2*cos(x)-6)
dfe=D(fe,"x")
fd=function(x){
  exp(x) - 2^(-x) * log(2) - 2 * sin(x)
}
```

Grafico

```
x<-seq(-1,3,0.1)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe
ggfx=ggplot(data=df) #carga los datos
```

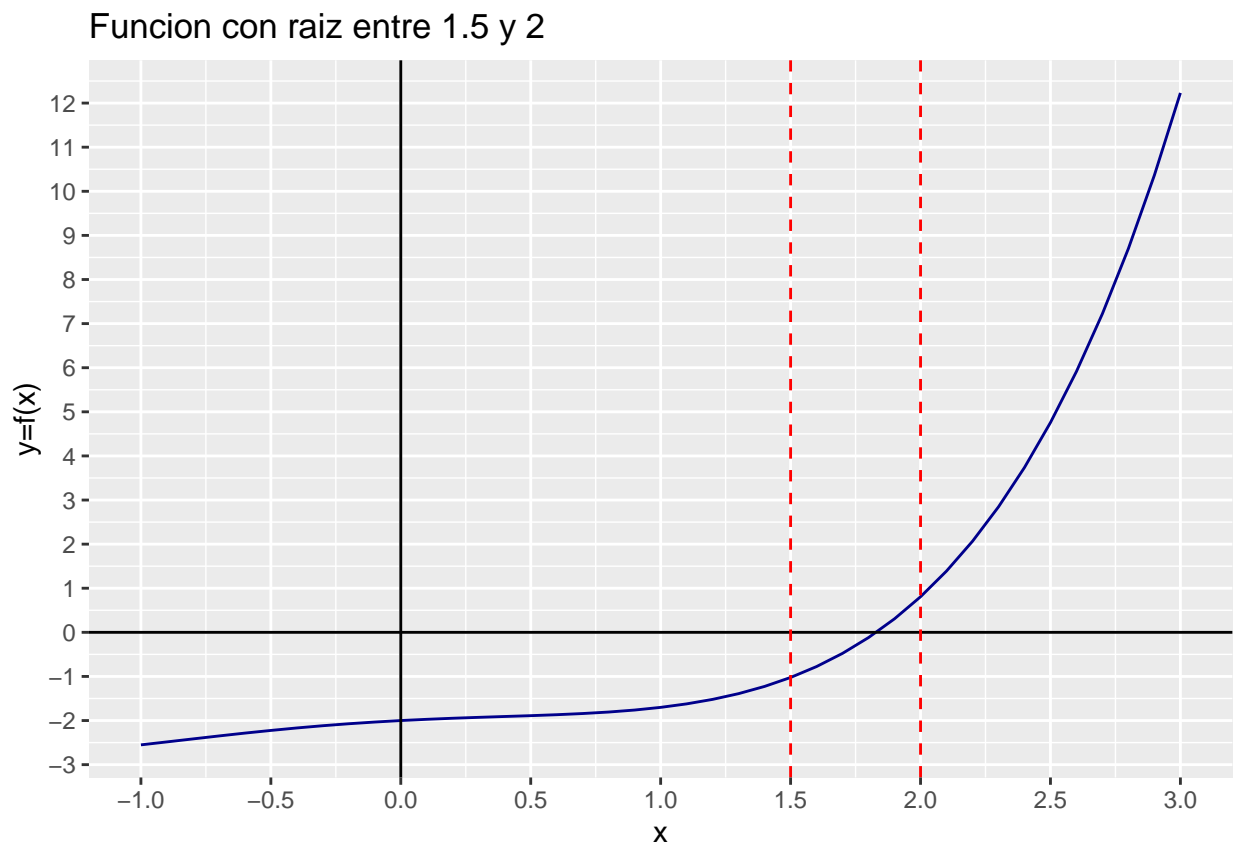
```

ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego línea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-1,4,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-5,15,1)) #Cambio escala eje Y

#Ahora se que la raiz esta entre 1.5 y 2. Grafico puntos en ese area
ggfx=ggfx+geom_vline(xintercept=c(1.5,2),linetype=2,colour="red")
ggfx=ggfx+ggtitle("Funcion con raiz entre 1.5 y 2") #Agrego titulo

ggfx

```



Aplico el metodo Newton-Raphson

```
paste("La raiz se encuentra en: ",metodo_newton(f,fd,1.5,0.0001,100))
```

```
## [1] "La raiz se encuentra en: 1.82938360193385"
```

##Ejercicio 2 Hallar la raiz para la funcion  $\log(x-1) + \cos(x-1) = 0$  en  $[1.3; 2]$  Primero busco la derivada de la funcion

```

f=function(x){
  log(x-1)+cos(x-1)
}

```

```
fe=expression(log(x-1)+cos(x-1))
dfe=D(fe,"x")

fd=function(x){
  1/(x - 1) - sin(x - 1)
}
```

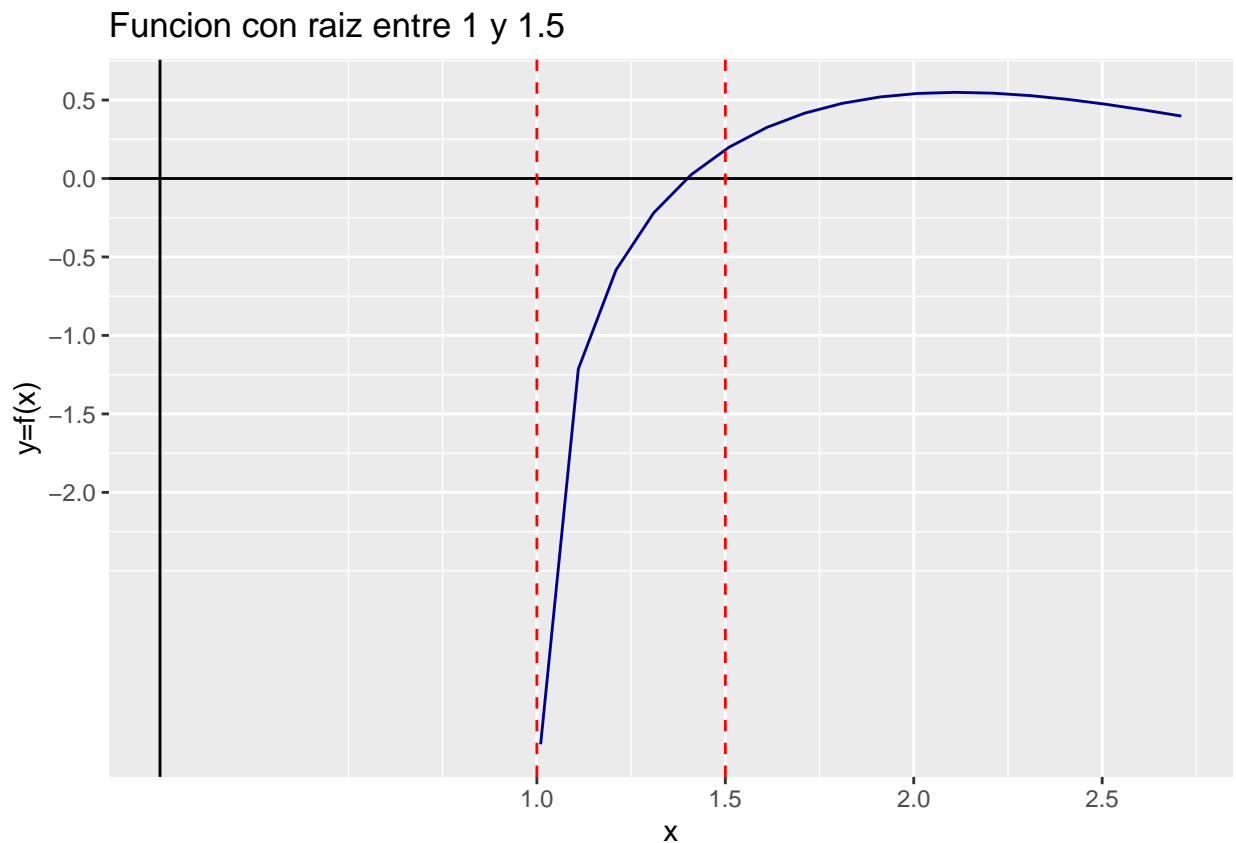
Grafico

```
x<-seq(1.01,2.8,0.1)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(1,3,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.5)) #Cambio escala eje Y

#Ahora se que la raiz esta entre 1 y 1.5. Grafico puntos en ese area
ggfx=ggfx+geom_vline(xintercept=c(1,1.5),linetype=2,colour="red")
ggfx=ggfx+ggtitle("Funcion con raiz entre 1 y 1.5") #Agrego titulo

ggfx
```



Aplico el metodo Newton-Raphson

```
paste("La raiz se encuentra en: ",metodo_newton(f,fd,1.1,0.0001,100))
```

```
## [1] "La raiz se encuentra en: 1.39774847515943"
```

##Ejercicio 3

```
f=function(x){
  2*x*cos(2*x)-(x-2)^2
}
fe=expression(2*x*cos(2*x)-(x-2)^2)
dfe=D(fe,"x")

fd=function(x){
  2 * cos(2 * x) - 2 * x * (sin(2 * x) * 2) - 2 * (x - 2)
}
```

Grafico en el intervalo [2;3] y [3;4]

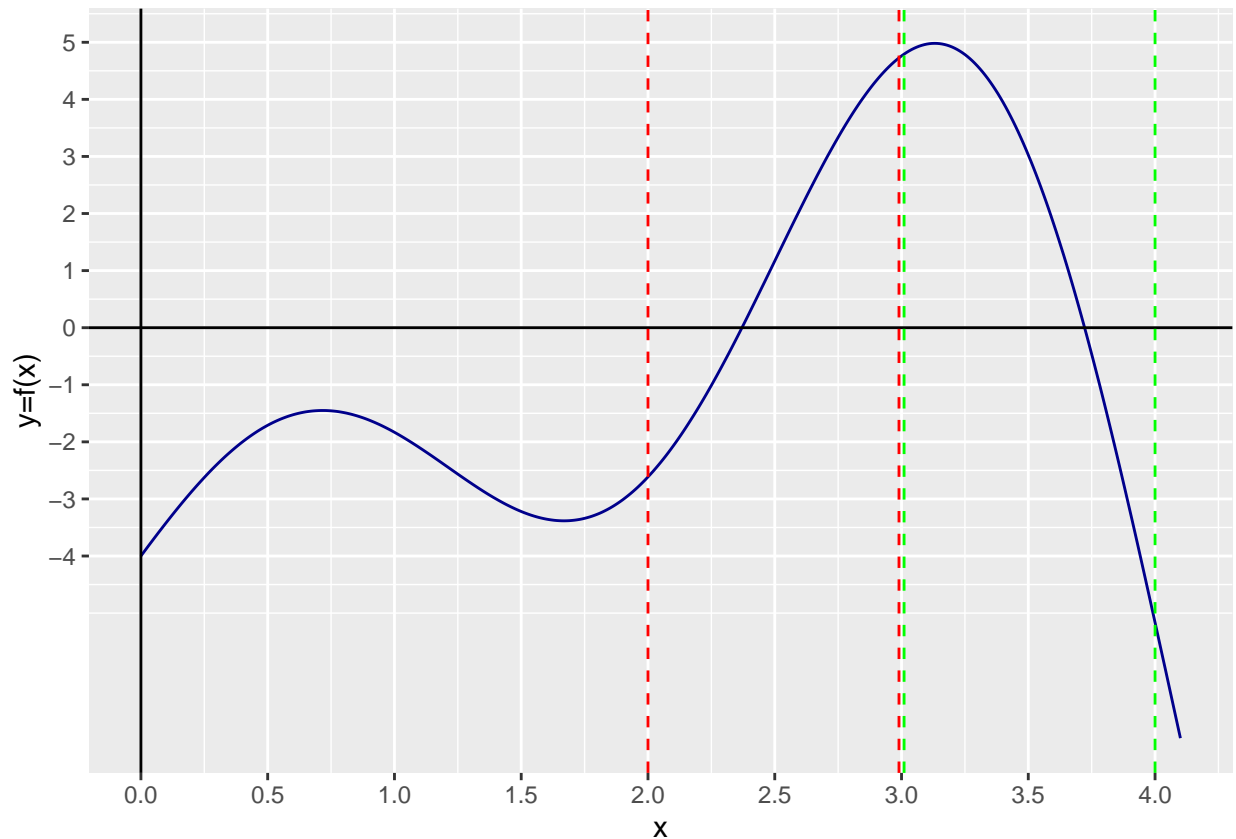
```
x<-seq(0,4.1,0.01)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,4,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-4,6,1)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(2,2.99),linetype=2,colour="red")+geom_vline(xintercept = c(3.01,4),linetype=2,colour="red")

ggfx
```





Aplico el metodo Newton-Raphson en ambos intervalos

```
paste("Las raices se encuentran en: ",metodo_newton(f,fd,2.5,0.0001,100)
      , " , y en :",metodo_newton(f,fd,3.5,0.0001,100))
```

```
## [1] "Las raices se encuentran en:  2.37068691766252 , y en : 3.72211277310661"
```

##Ejercicio 4

```
f=function(x){
  (x-2)^2-log(x)
}
fe=expression((x-2)^2-log(x))
dfe=D(fe,"x")
fd=function(x){
  2 * (x - 2) - 1/x
}
```

Grafico en el intervalo [1;2] y [e;4]

```
x<-seq(0,4,0.01)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
```

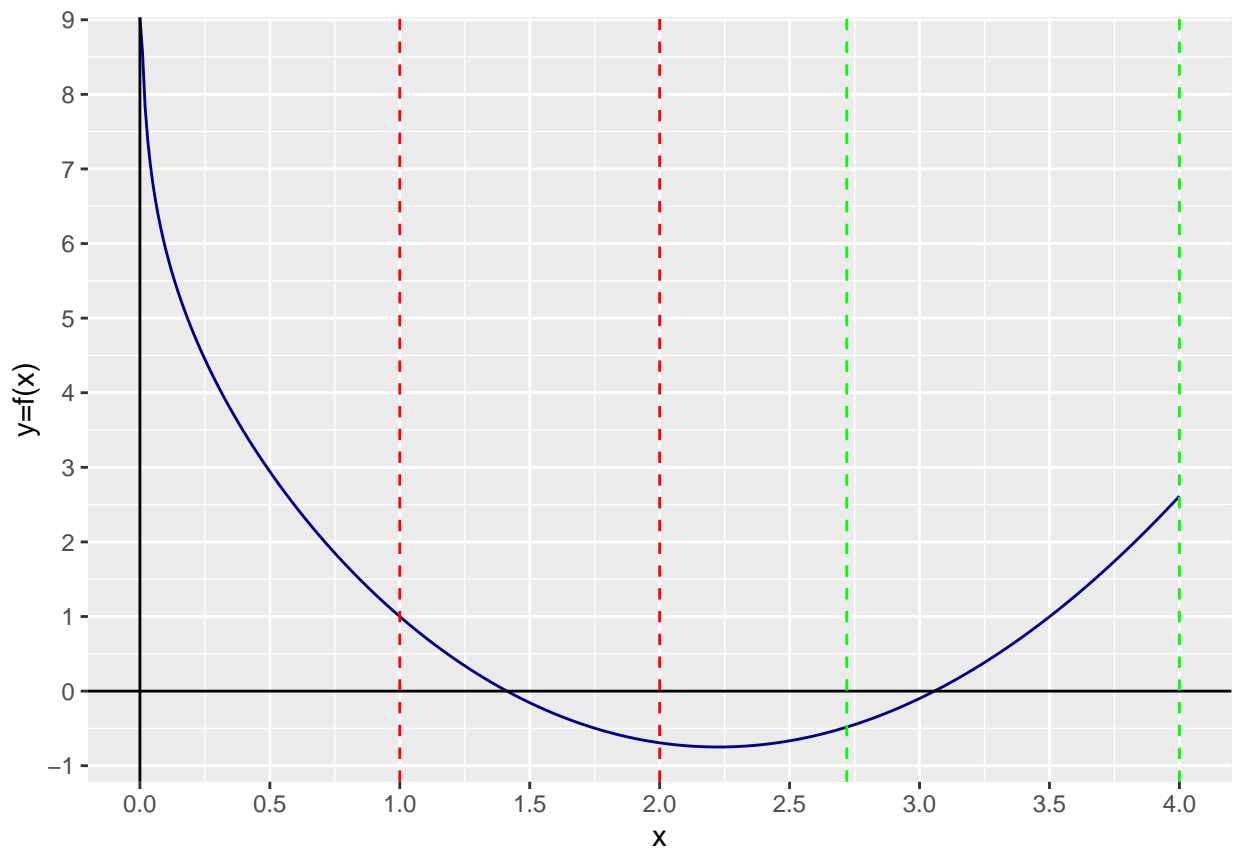
```

ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-1,10,1)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(1,2),linetype=2,colour="red")+geom_vline(xintercept = c(2.72,4),linetype=2,colour="green")

ggfx

```



Aplico el metodo Newton-Raphson en ambos intervalos

```

paste("Las raices se encuentran en: ",metodo_newton(f,fd,1.5,0.0001,100)
      , ", y en :",metodo_newton(f,fd,3,0.0001,100))

```

```
## [1] "Las raices se encuentran en: 1.41239117172501 , y en : 3.05710354999844"
```

##Ejercicio 5

```

f=function(x){
  exp(x)-3*x^2
}
fe=expression(exp(x)-3*x^2)

```

```
dfe=D(fe,"x")

fd=function(x){
  exp(x) - 3 * (2 * x)
}
```

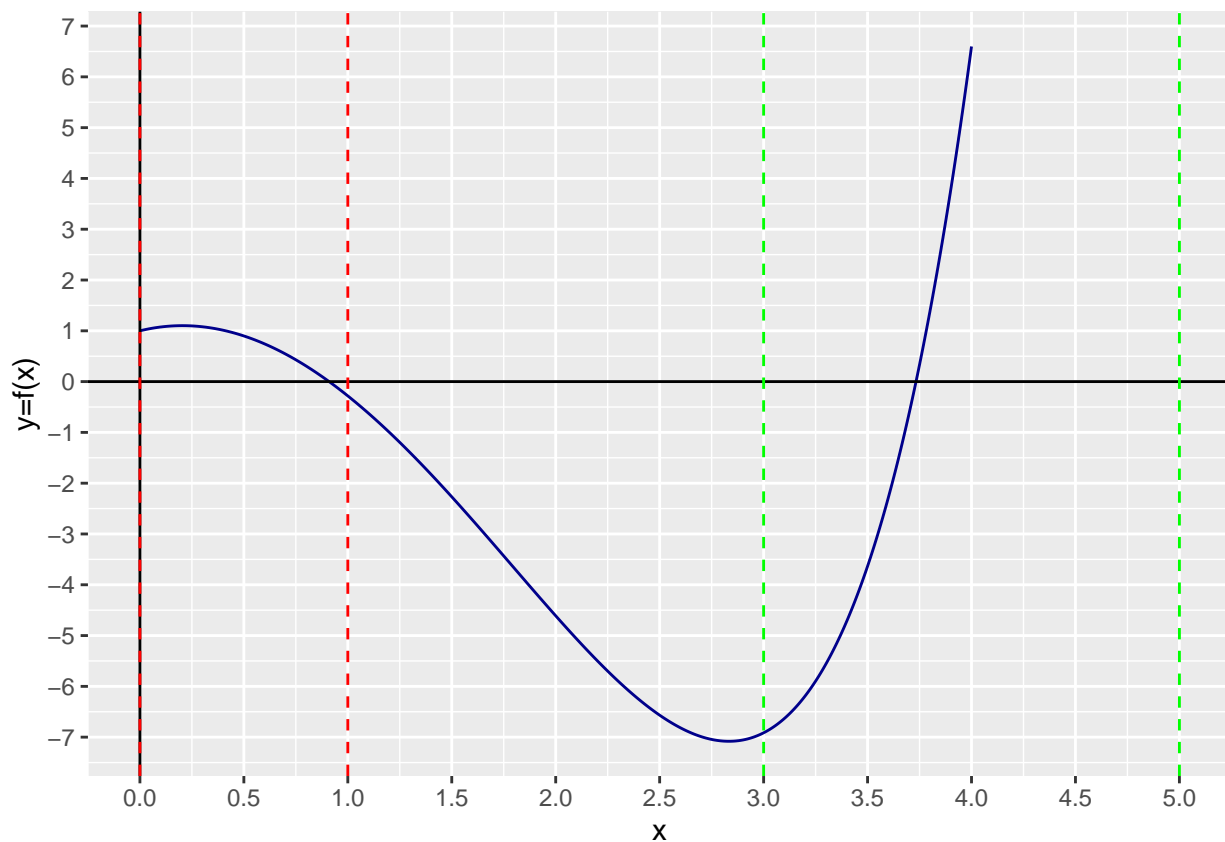
Grafico en el intervalo  $[0; 1]$  y  $[3; 5]$

```
x<-seq(0,4,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-8,8,1)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(0,1),linetype=2,colour="red")+geom_vline(xintercept = c(3,5),linetype=2,colour="green")

ggfx
```



Aplico el metodo Newton-Raphson en ambos intervalos

```
paste("Las raices se encuentran en: ",metodo_newton(f,fd,1.5,0.0001,100)
      , " , y en :",metodo_newton(f,fd,4,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 0.910007572539519 , y en : 3.73307902865468"
```

```
##Ejercicio 6
```

```
f=function(x){
  sin(x)-exp(-x)
}
fe=expression(sin(x)-exp(-x))
dfe=D(fe,"x")

fd=function(x){
  cos(x) + exp(-x)
}
```

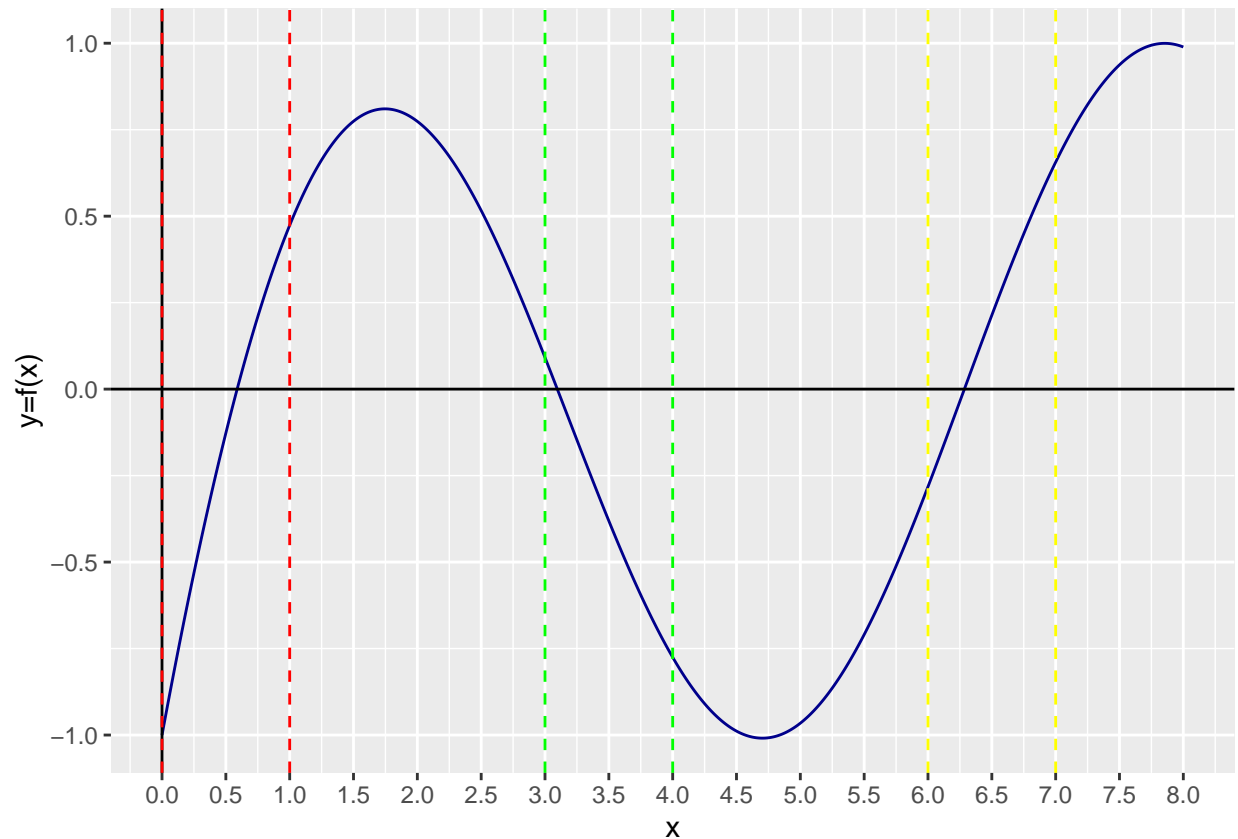
Grafico en el intervalo  $[0; 1]$  ,  $[3; 4]$  y  $[6; 7]$

```
x<-seq(0,8,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,10,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.5)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(0,1),linetype=2,colour="red")+geom_vline(xintercept = c(3,4),linetype=2)

ggfx
```



Aplico el metodo Newton-Raphson en todos intervalos

```
paste("Las raices se encuentran en: ",metodo_newton(f,fd,0.5,0.0001,100)," , en :", metodo_newton(f,fd,3.09636393241066,0.0001,100),
      , " , y en :",metodo_newton(f,fd,6.5,0.0001,100))
```

```
## [1] "Las raices se encuentran en:  0.588532743977419 , en : 3.09636393241066 , y en : 6.28504927338217"
```

##Ejercicio 7

```
f=function(x){
  cos(x)-sqrt(x)
}
fe=expression(cos(x)-sqrt(x))
dfe=D(fe,"x")

fd=function(x){
  -(sin(x) + 0.5 * x^-0.5)
}
```

Grafico en el intervalo [0;2]

```
x<-seq(0,2,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe
```

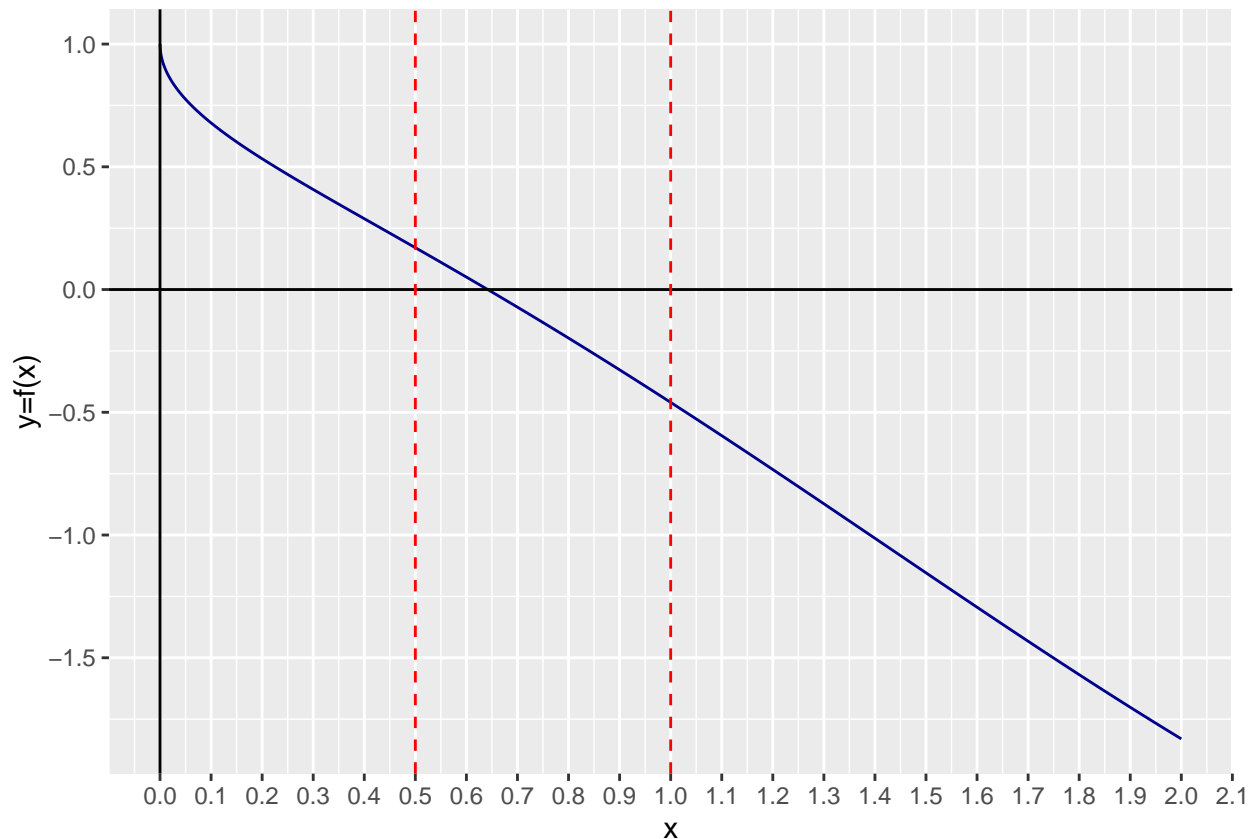
```

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego línea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,3,0.1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-3,2,0.5)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(0.5,1),linetype=2,colour="red")

ggfx

```



Aplico el metodo Newton-Raphson en el intervalo

```
paste("La raiz se encuentra en: ",metodo_newton(f,fd,0.6,0.0001,100))
```

```
## [1] "La raiz se encuentra en: 0.641714370872883"
```

##Ejercicio 8

```

f=function(x){
  2+cos(exp(x)-2)-exp(x)
}
fe=expression(2+cos(exp(x)-2)-exp(x))

```

```
dfe=D(fe,"x")
fd=function(x){
  -(sin(exp(x) - 2) * exp(x) + exp(x))
}
```

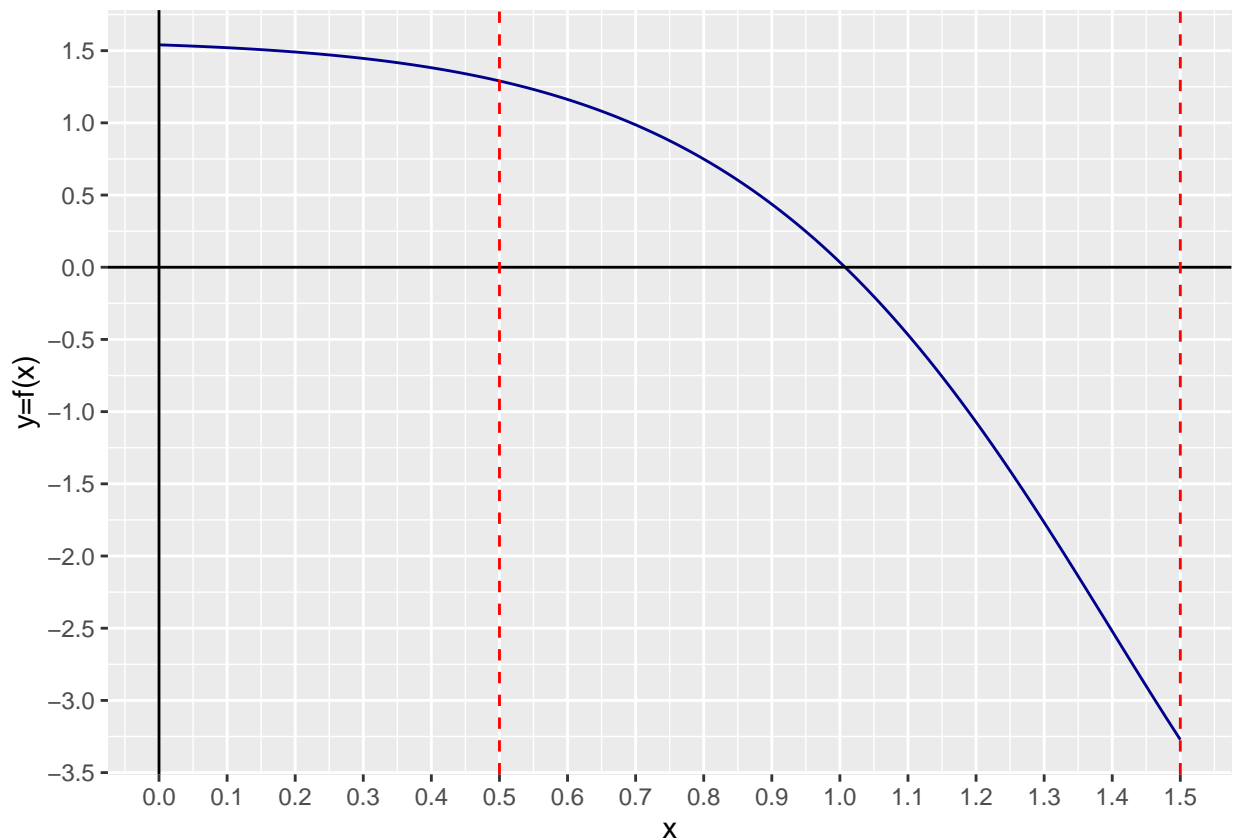
Grafico en el intervalo  $[0; 1.5]$

```
x<-seq(0,1.5,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,2,0.1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-4,2,0.5)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(0.5,1.5),linetype=2,colour="red")

ggfx
```



Aplico el metodo Newton-Raphson en el intervalo

```
paste("La raiz se encuentra en: ",metodo_newton(f,fd,0.8,0.0001,100))
```

```
## [1] "La raiz se encuentra en: 1.00762397218537"
```

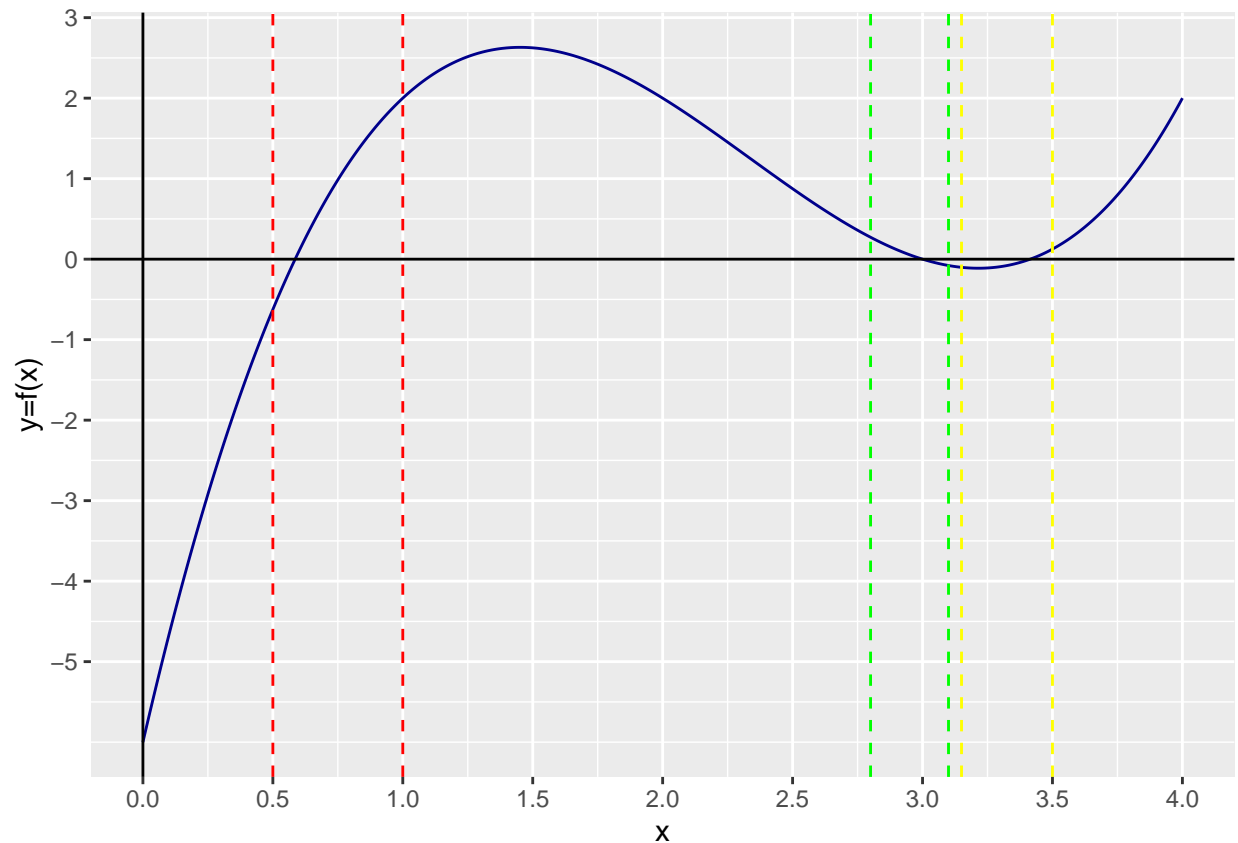
##Ejercicio 9

```
f=function(x){  
  x^3-7*x^2+14*x-6  
}  
fe=expression(x^3-7*x^2+14*x-6)  
dfe=D(fe,"x")  
fd=function(x){  
  3 * x^2 - 7 * (2 * x) + 14  
}
```

Grafico en el intervalo [0;4]

```
x<-seq(0,4,0.001)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos  
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables  
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea  
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0  
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X  
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-5,5,1)) #Cambio escala eje Y  
  
ggfx=ggfx+geom_vline(xintercept=c(0.5,1),linetype=2,colour="red")+  
  geom_vline(xintercept=c(2.8,3.1),linetype=2,colour="green")+  
  geom_vline(xintercept=c(3.15,3.5),linetype=2,colour="yellow")  
  
ggfx
```





Aplico el metodo Newton-Raphson en el intervalo

```
paste("Las raices se encuentran en: ",metodo_newton(f,fd,0.5,0.0001,100)," , en :", metodo_newton(f,fd,3.5,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 0.585786437314424 , en : 3 , y en : 3.41421356237479"
```

##Ejercicio 10

```
f=function(x){
  cos(x)-x
}
fe=expression(cos(x)-x)
dfe=D(fe,"x")
fd=function(x){
  -(sin(x) + 1)
}
```

Grafico en el intervalo  $[-2; 2]$

```
x<-seq(-2,2,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe
ggfx=ggplot(data=df) #carga los datos
```

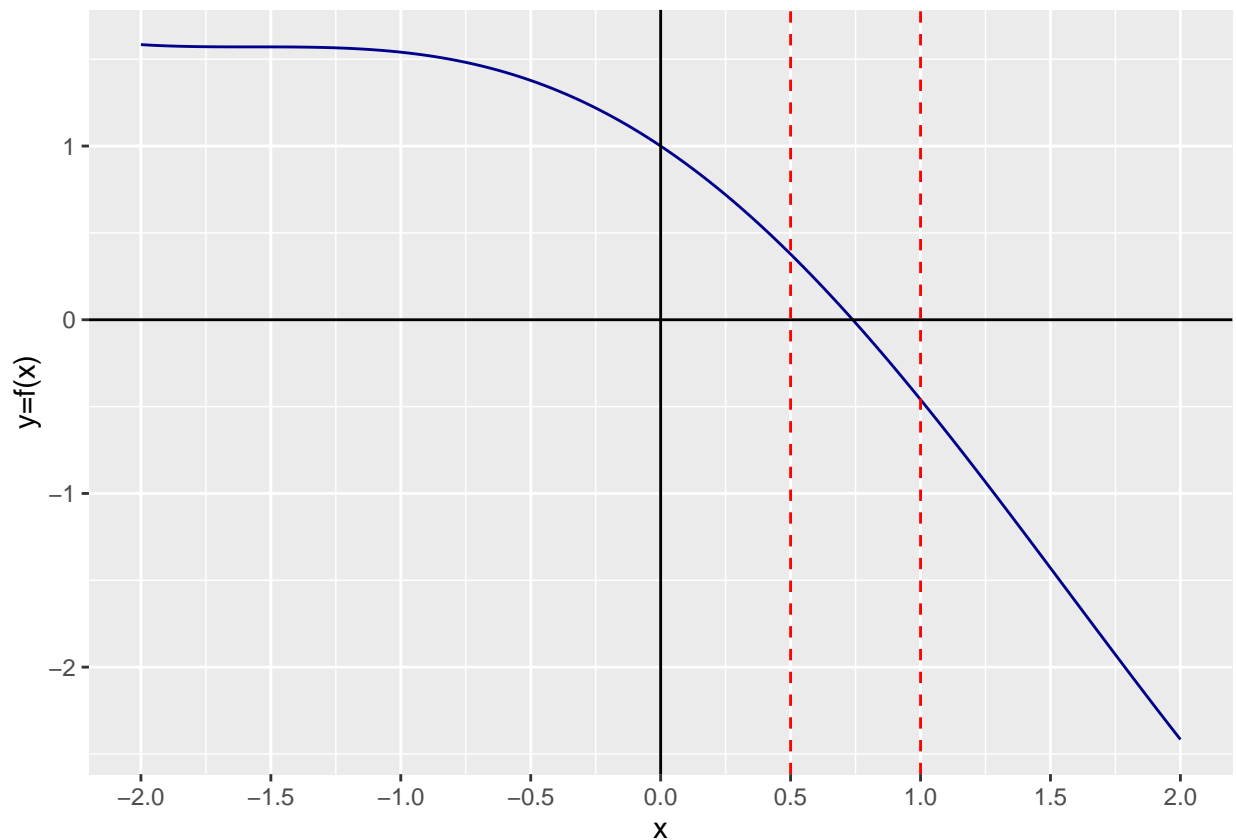
```

ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-3,3,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-3,2,1)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(0.5,1),linetype=2,colour="red")

ggfx

```



Aplico el metodo Newton-Raphson en el intervalo

```
paste("Las raices se encuentran en: ",metodo_newton(f,fd,0.5,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 0.739085133920807"
```

##Ejercicio 11

```

f=function(x){
  -x^3-cos(x)
}
fe=expression(-x^3-cos(x))
dfe=D(fe,"x")

```

```
fd=function(x){
  -(3 * x^2 - sin(x))
}
```

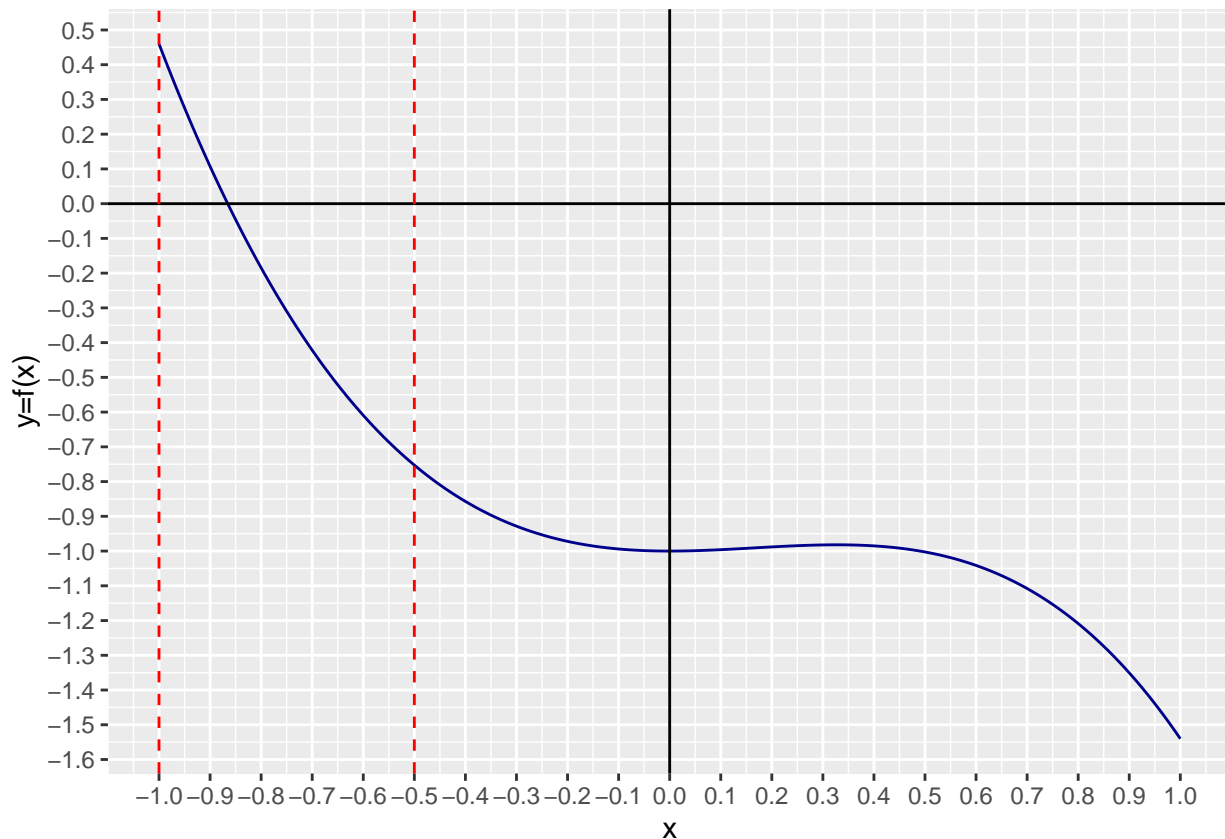
Grafico en el intervalo  $[-1;1]$

```
x<-seq(-1,1,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-1,1,0.1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.1)) #Cambio escala eje Y

ggfx=ggfx+geom_vline(xintercept=c(-0.5,-1),linetype=2,colour="red")

ggfx
```



Aplico el metodo Newton-Raphson en el intervalo

```
paste("Las raices se encuentran en: ",metodo_newton(f,fd,-0.5,0.0001,100))
```

```
## [1] "Las raices se encuentran en: -0.865474033110957"
```

#Metodo secante ##Ejercicio 1 Hallar la raiz para la funcion  $e^x + 2(-x) + 2\cos(x) - 6 = 0$  en  $[1; 2]$

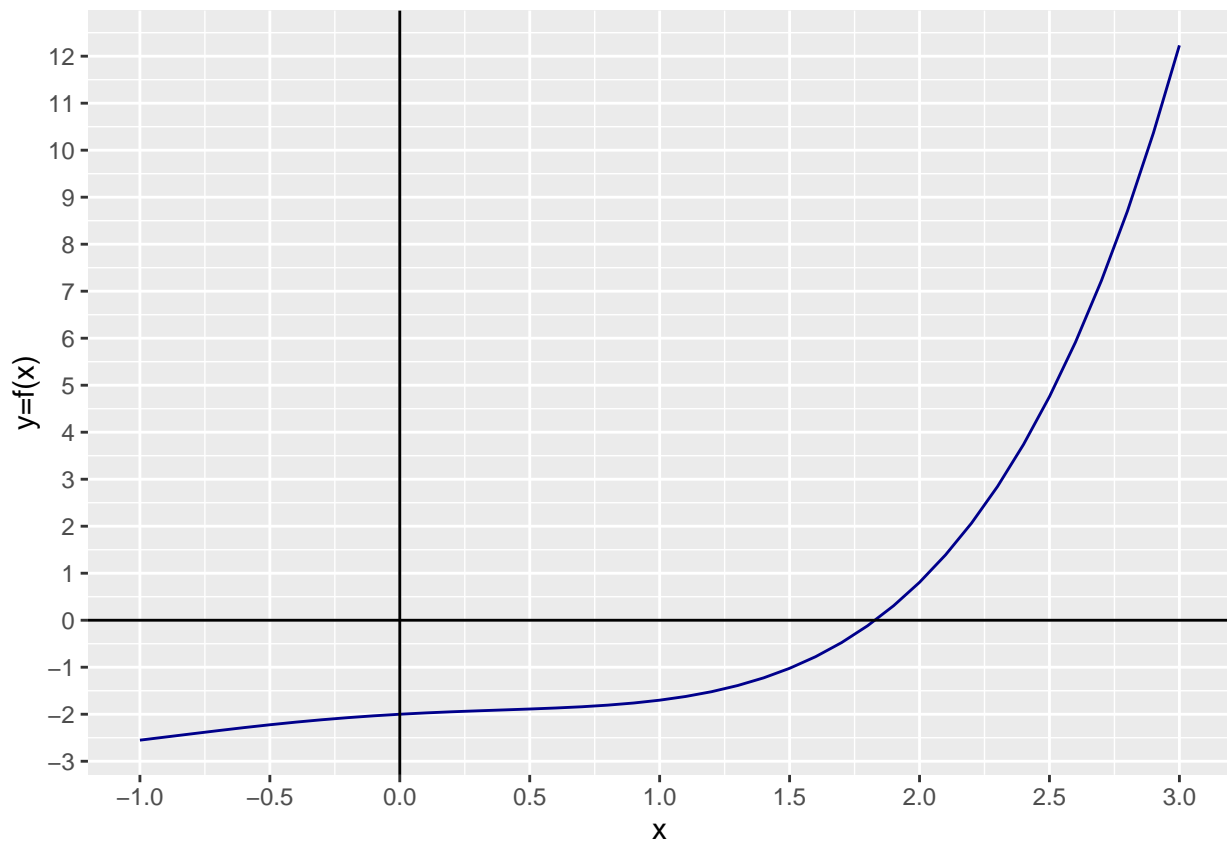
```
f=function(x){
  exp(x)+2*(-x)+2*cos(x)-6
}
```

Grafico

```
x<-seq(-1,3,0.1)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-1,4,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-5,15,1)) #Cambio escala eje Y

ggfx
```



Aplico el metodo de la secante

```
paste("La raiz se encuentra en: ",metodo_secante(f,1.5,3,0.0001,100))
```

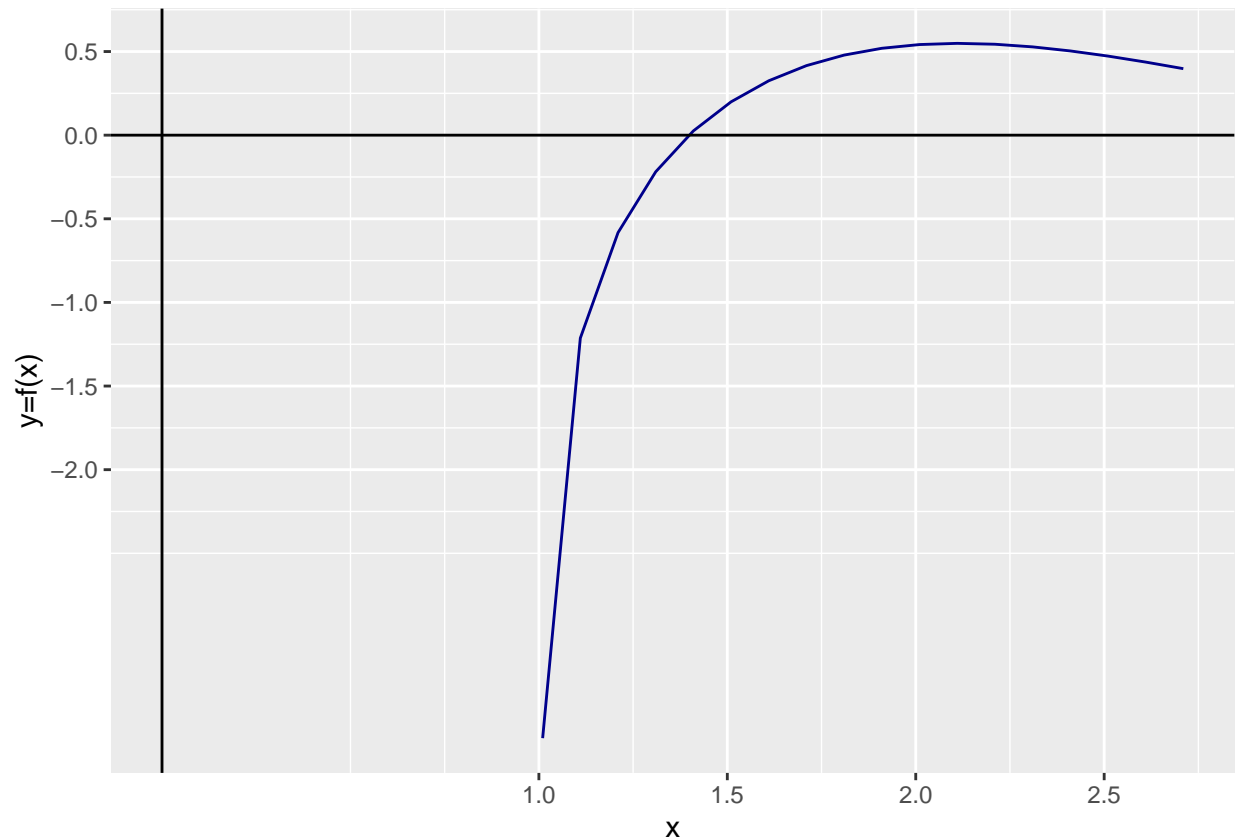
```
## [1] "La raiz se encuentra en: 1.82938383318023"
```

##Ejercicio 2 Hallar la raiz para la funcion  $\log(x-1) + \cos(x-1) = 0$  en  $[1.3; 2]$

```
f=function(x){  
  log(x-1)+cos(x-1)  
}
```

Grafico

```
x<-seq(1.01,2.8,0.1)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos  
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables  
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea  
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0  
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(1,3,0.5)) #cambio escala eje X  
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.5)) #Cambio escala eje Y  
  
ggfx
```



Aplico el metodo de la secante

```
paste("La raiz se encuentra en: ",metodo_secante(f,1.4,1.9,0.0001,100))
```

```
## [1] "La raiz se encuentra en: 1.3977484296268"
```

##Ejercicio 3

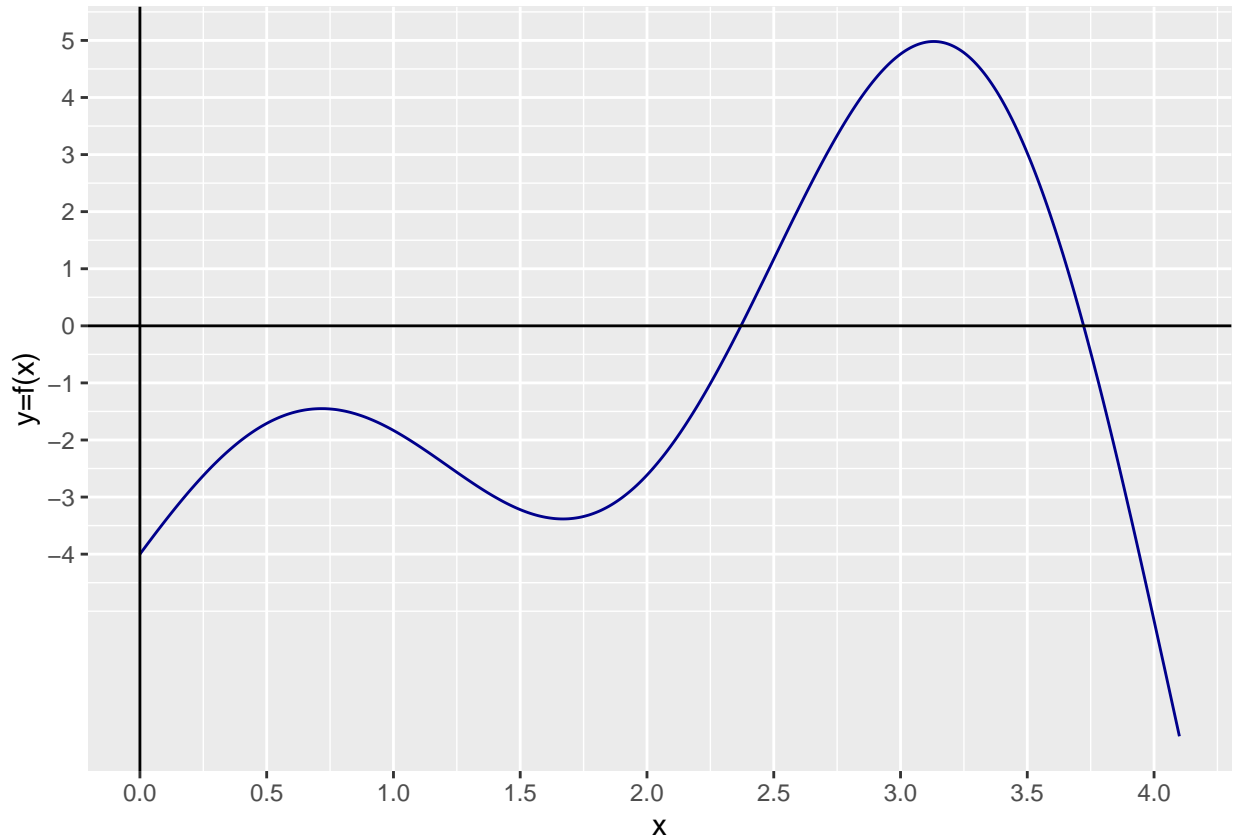
```
f=function(x){
  2*x*cos(2*x)-(x-2)^2
}
```

Grafico en el intervalo [2;3] y [3;4]

```
x<-seq(0,4.1,0.01)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,4,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-4,6,1)) #Cambio escala eje Y
```

ggfx



Aplico el metodo de la secante en ambos intervalos

```
paste("Las raices se encuentran en: ",metodo_secante(f,2.25,2.5,0.0001,100)  
      , ", y en :",metodo_secante(f,3.5,3.75,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 2.3706869176549 , y en : 3.72211309165018"
```

##Ejercicio 4

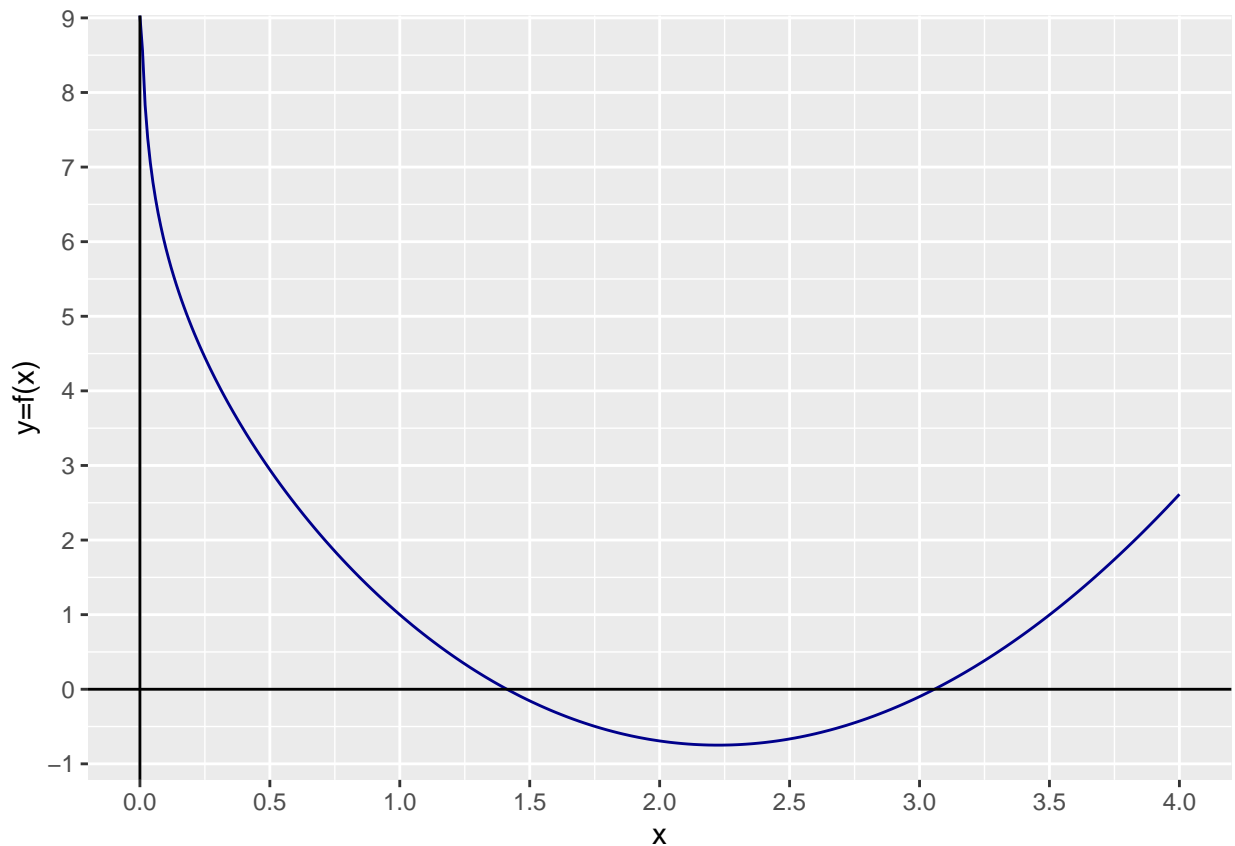
```
f=function(x){  
  (x-2)^2-log(x)  
}
```

Grafico en el intervalo  $[1;2]$  y  $[e;4]$

```
x<-seq(0,4,0.01)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos
```

```
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-1,10,1)) #Cambio escala eje Y
```

```
ggfx
```



Aplico el metodo de la secante en ambos intervalos

```
paste("Las raices se encuentran en: ",metodo_secante(f,1.25,1.5,0.0001,100)
, ", y en :",metodo_secante(f,3,3.25,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 1.41239117327502 , y en : 3.05710354920478"
```

##Ejercicio 5

```
f=function(x){
  exp(x)-3*x^2
}
```

Grafico en el intervalo [0; 1] y [3; 5]



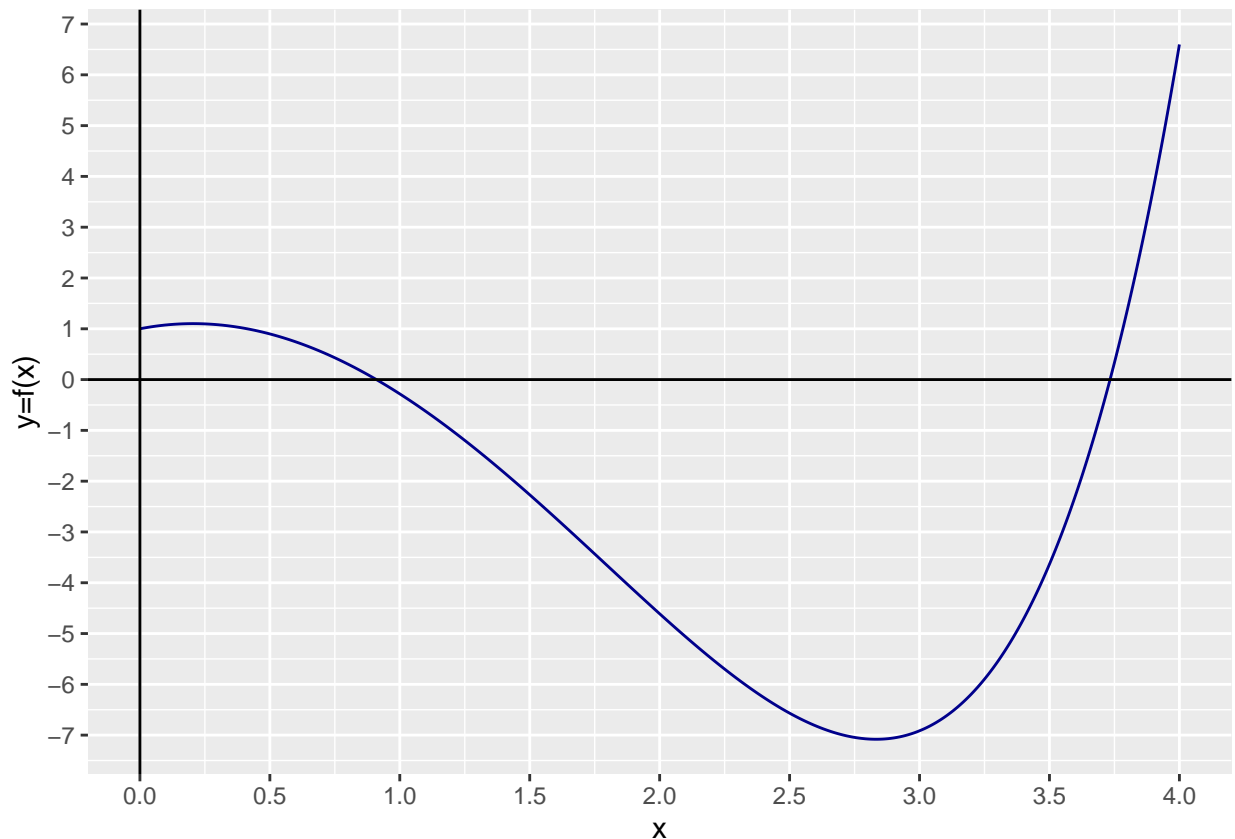
```

x<-seq(0,4,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-8,8,1)) #Cambio escala eje Y

```

```
ggfx
```



Aplico el metodo de la secante en ambos intervalos

```

paste("Las raices se encuentran en: ",metodo_secante(f,0.75,1,0.0001,100)
      , ", y en :",metodo_secante(f,3.5,4,0.0001,100))

```

```
## [1] "Las raices se encuentran en: 0.910007571845458 , y en : 3.73307902424239"
```

##Ejercicio 6

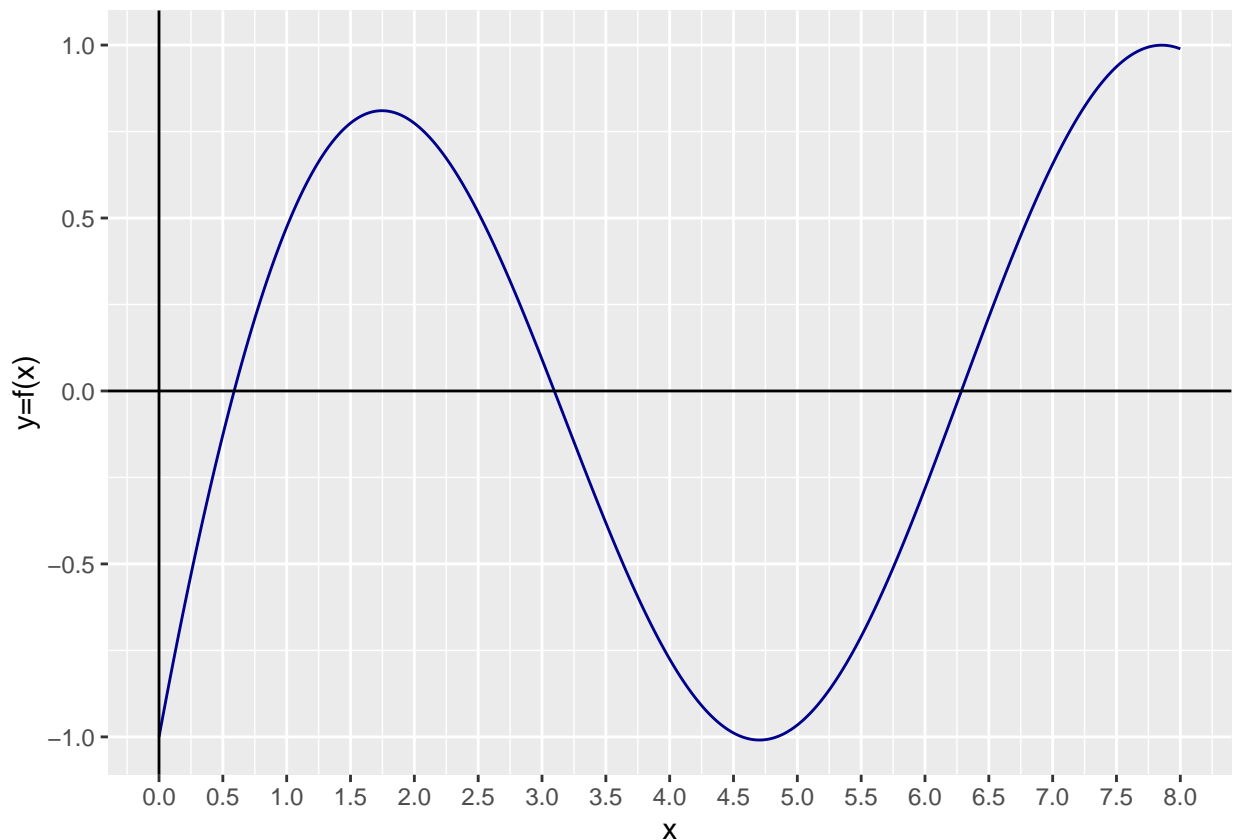
```
f=function(x){
  sin(x)-exp(-x)
}
```

Grafico en el intervalo  $[0; 1]$  ,  $[3; 4]$  y  $[6; 7]$

```
x<-seq(0,8,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,10,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.5)) #Cambio escala eje Y

ggfx
```



Aplico el metodo de la secante en todos intervalos

```
paste("Las raices se encuentran en: ",metodo_secante(f,0.5,0.75,0.0001,100),", en :", metodo_secante(f,
, " , y en :",metodo_secante(f,6.25,6.5,0.0001,100))
```

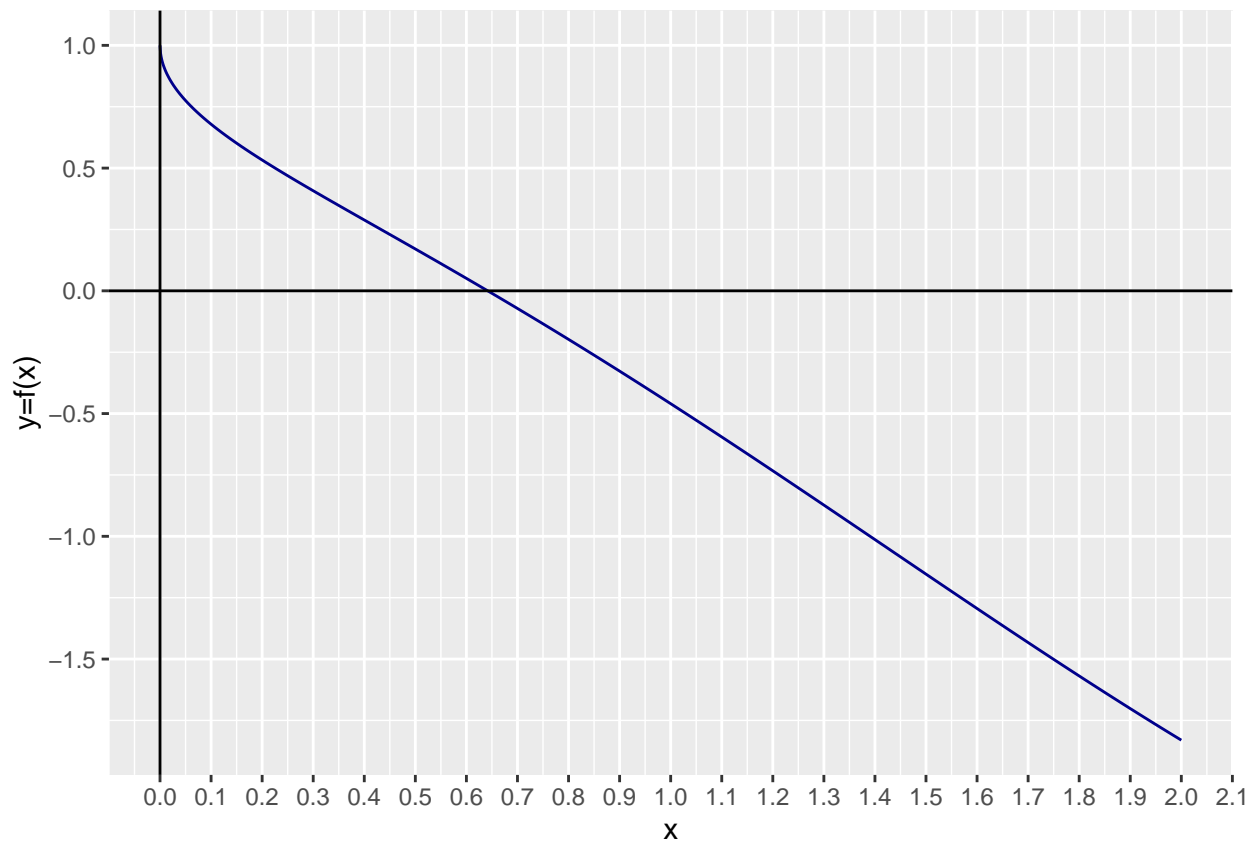
```
## [1] "Las raices se encuentran en: 0.588532744147662 , en : 3.09636393245321 , y en : 6.285049273383"
```

```
##Ejercicio 7
```

```
f=function(x){  
  cos(x)-sqrt(x)  
}
```

Grafico en el intervalo  $[0;2]$

```
x<-seq(0,2,0.001)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos  
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables  
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea  
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0  
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,3,0.1)) #cambio escala eje X  
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-3,2,0.5)) #Cambio escala eje Y  
  
ggfx
```



Aplico el metodo de la secante en el intervalo

```
paste("La raiz se encuentra en: ",metodo_secante(f,0.6,0.7,0.0001,100))
```

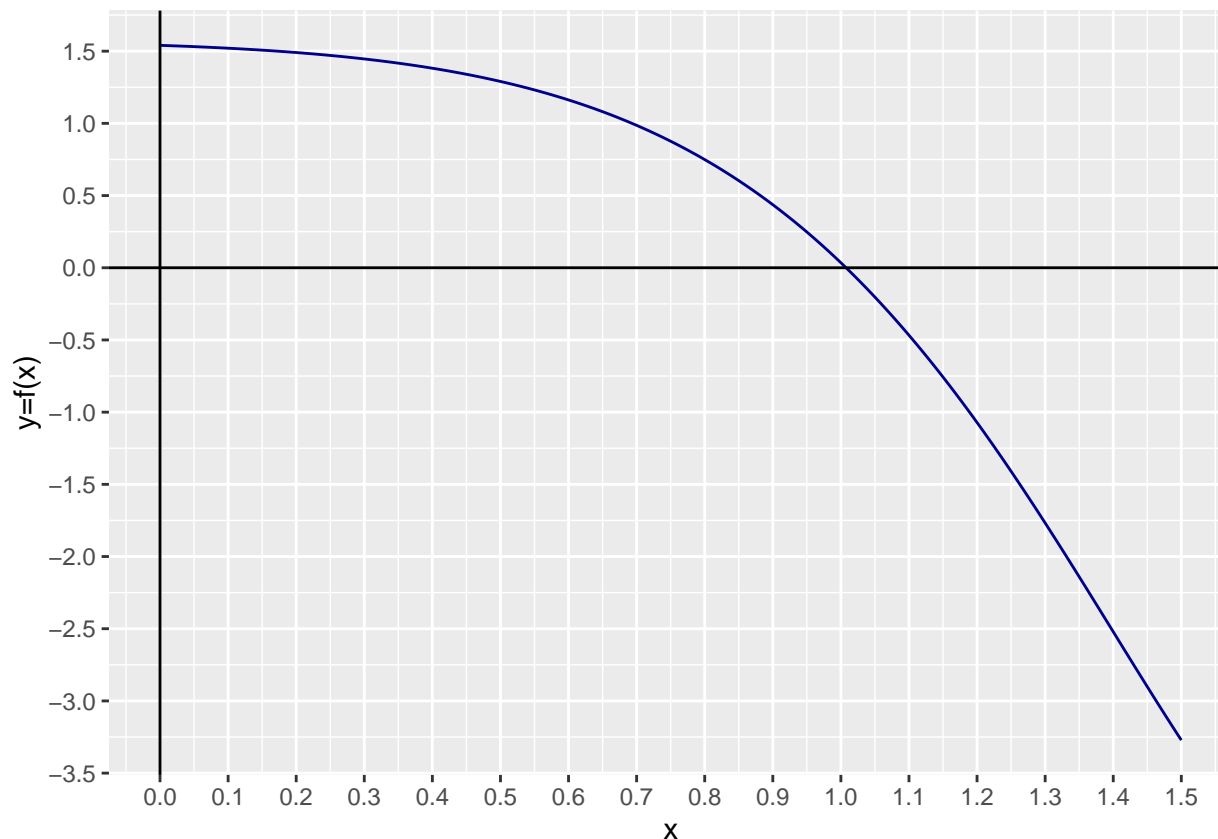
```
## [1] "La raiz se encuentra en: 0.64171437097005"
```

##Ejercicio 8

```
f=function(x){  
  2+cos(exp(x)-2)-exp(x)  
}
```

Grafico en el intervalo [0; 1.5]

```
x<-seq(0,1.5,0.001)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos  
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables  
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea  
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0  
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,2,0.1)) #cambio escala eje X  
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-4,2,0.5)) #Cambio escala eje Y  
  
ggfx
```



Aplico el metodo de la secante en el intervalo

```
paste("La raiz se encuentra en: ",metodo_secante(f,0.9,1.1,0.0001,100))
```

```
## [1] "La raiz se encuentra en: 1.00762395678363"
```

##Ejercicio 9

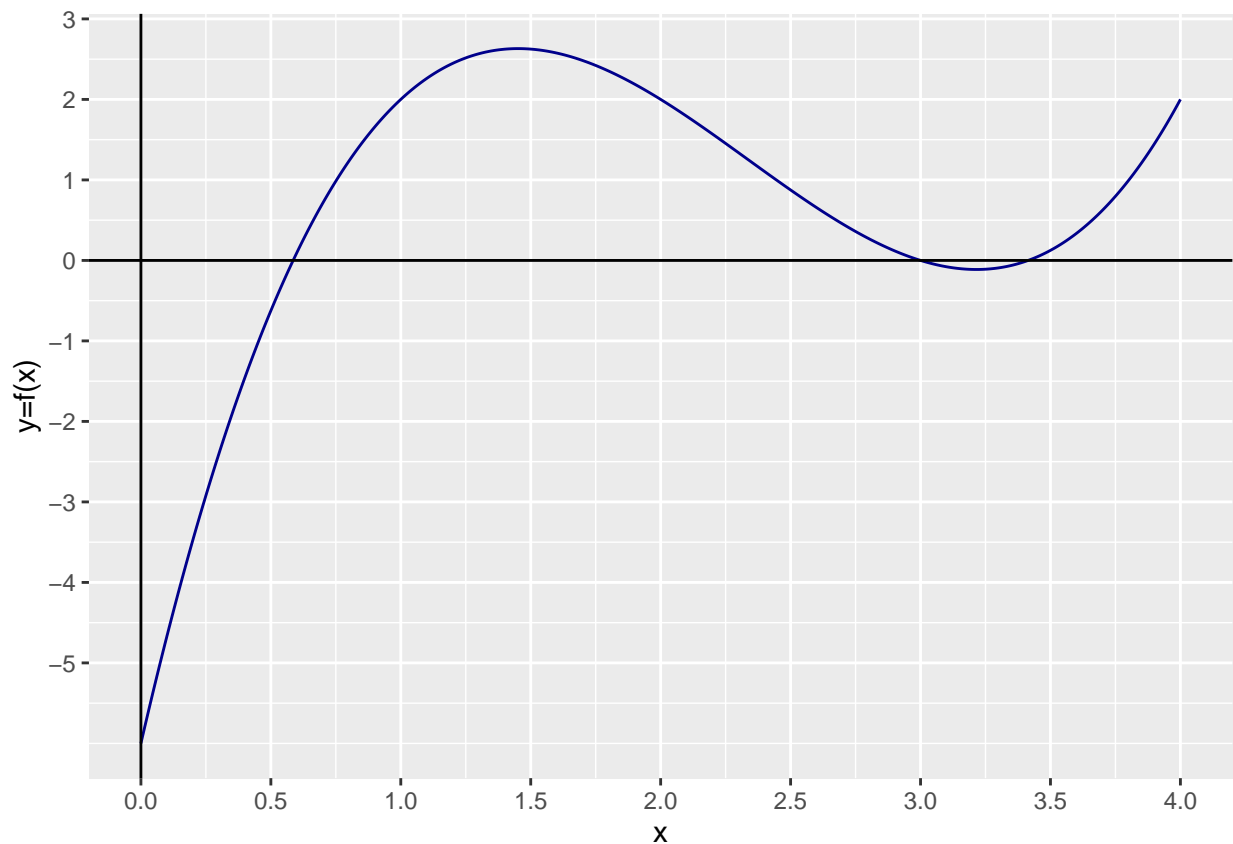
```
f=function(x){
  x^3-7*x^2+14*x-6
}
```

Grafico en el intervalo [0;4]

```
x<-seq(0,4,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-5,5,1)) #Cambio escala eje Y
```

ggfx



Aplico el metodo de la secante en el intervalo

```
paste("Las raices se encuentran en: ",metodo_secante(f,0.5,0.75,0.0001,100),", en :", metodo_secante(f,3.25,3.5,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 0.585786454415883 , en : 2.99999997949999 , y en : 3.414213362601"
```

##Ejercicio 10

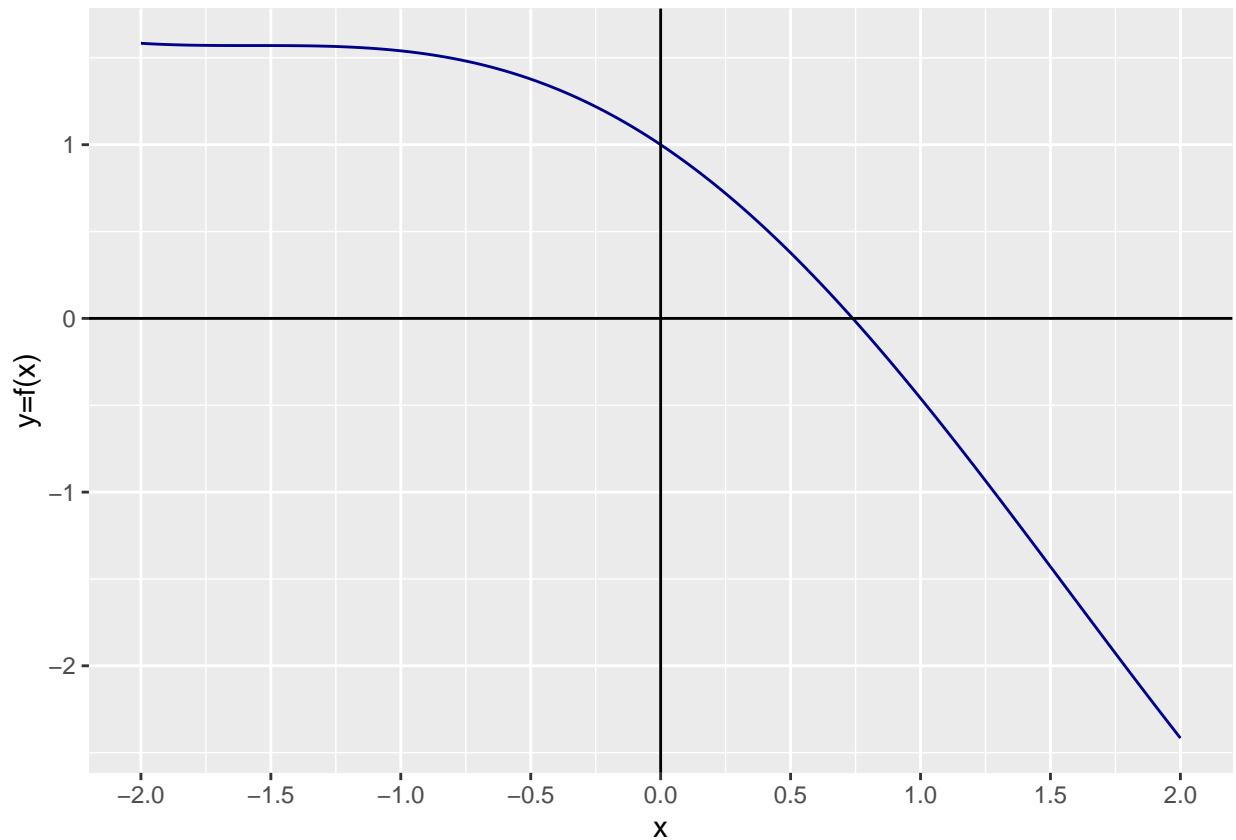
```
f=function(x){  
  cos(x)-x  
}
```

Grafico en el intervalo  $[-2; 2]$

```
x<-seq(-2,2,0.001)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos  
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
```

```
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego línea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-3,3,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-3,2,1)) #Cambio escala eje Y
```

```
ggfx
```



Aplico el metodo de la secante en el intervalo

```
paste("Las raices se encuentran en: ",metodo_secante(f,0.5,1,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 0.739085132900112"
```

##Ejercicio 11

```
f=function(x){
  -x^3-cos(x)
}
```

Grafico en el intervalo  $[-1; 1]$

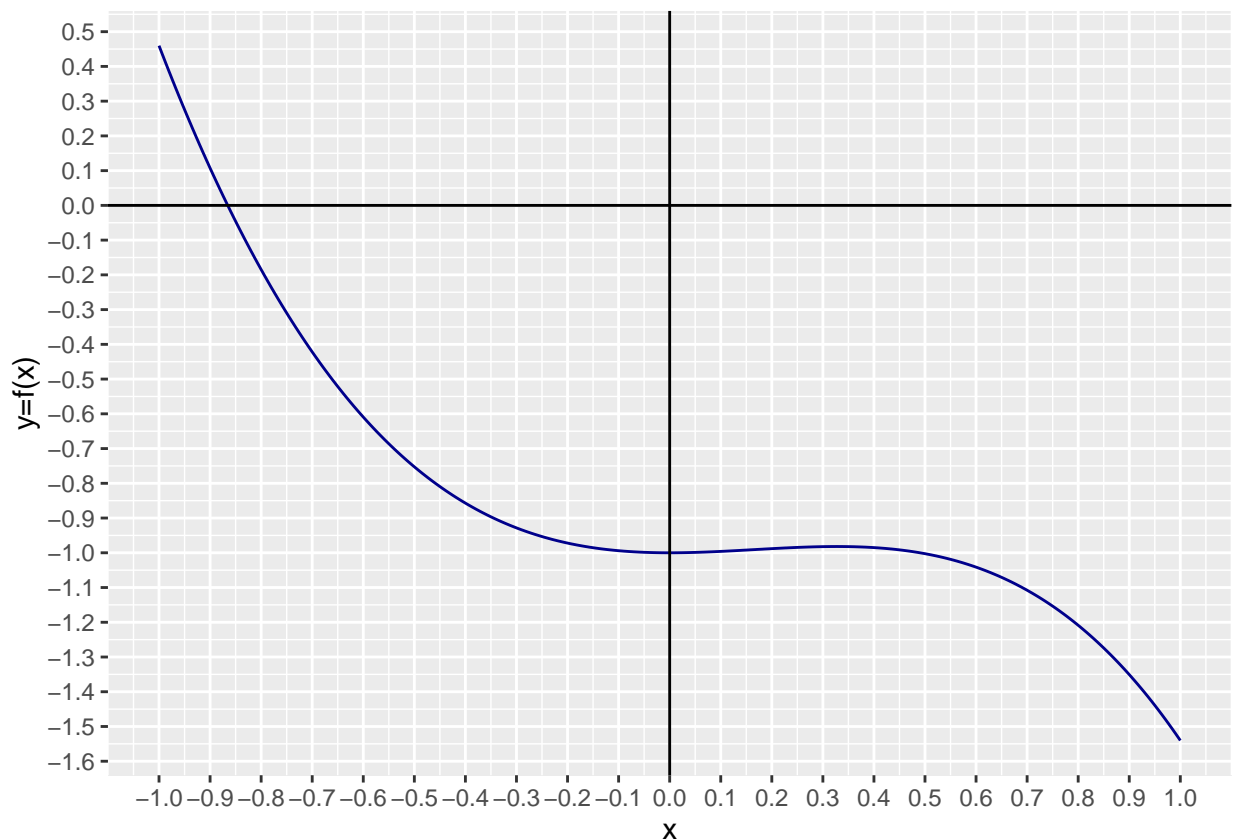
```

x<-seq(-1,1,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-1,1,0.1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.1)) #Cambio escala eje Y

ggfx

```



Aplico el metodo de la secante en el intervalo

```

paste("Las raices se encuentran en: ",metodo_secante(f,-0.9,-0.8,0.0001,100))

```

```
## [1] "Las raices se encuentran en: -0.865474033054677"
```

#Metodo posicion falsa ##Ejercicio 1 Hallar la raiz para la funcion  $e^x + 2^{(-x)} + 2\cos(x) - 6 = 0$  en  $[1; 2]$

```

f=function(x){
  exp(x)+2^(-x)+2*cos(x)-6
}

```

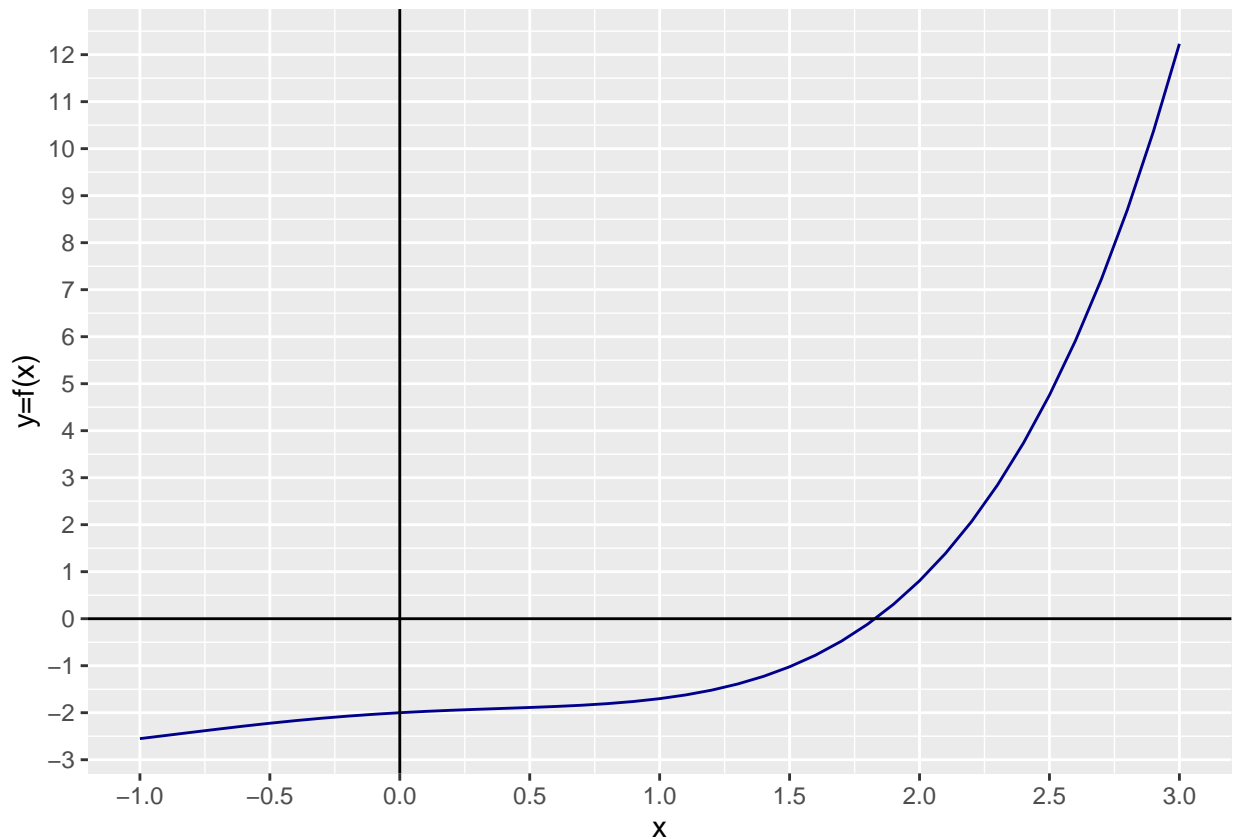


Grafico

```
x<-seq(-1,3,0.1)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-1,4,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-5,15,1)) #Cambio escala eje Y

ggfx
```



Aplico el metodo de posicion falsa

```
paste("La raiz se encuentra en: ",posicion_falsa(f,1.5,2,0.00001,100))
```

```
## [1] "La raiz se encuentra en: 1.77957859553387"
```

```
##Ejercicio 2 Hallar la raiz para la funcion  $\log(x-1) + \cos(x-1) = 0$  en  $[1.3; 2]$ 
```

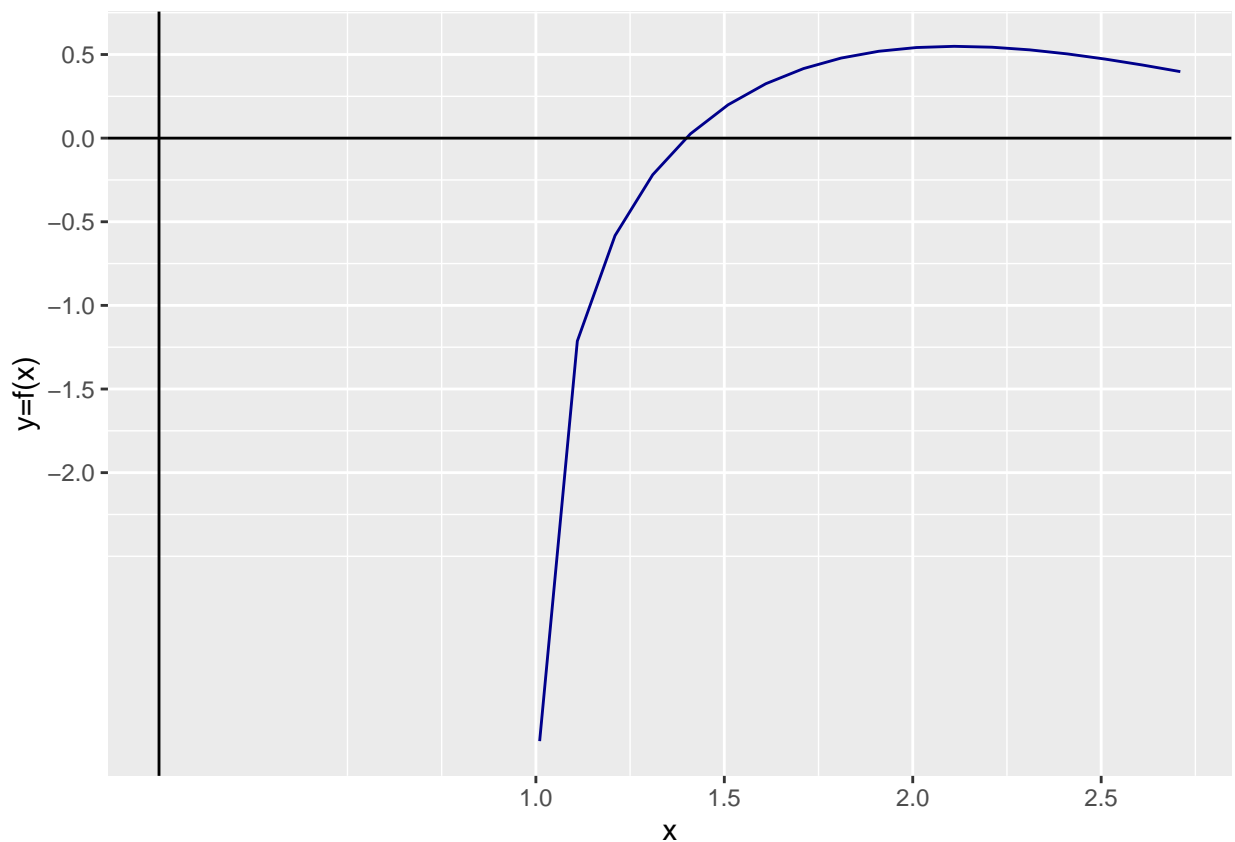
```
f=function(x){
  log(x-1)+cos(x-1)
}
```

Grafico

```
x<-seq(1.01,2.8,0.1)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(1,3,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.5)) #Cambio escala eje Y

ggfx
```



Aplico el metodo de posicion falsa

```
paste("La raiz se encuentra en: ",posicion_falsa(f,1.25,1.5,0.0001,100))
```

```
## [1] "La raiz se encuentra en: 1.39777219056584"
```

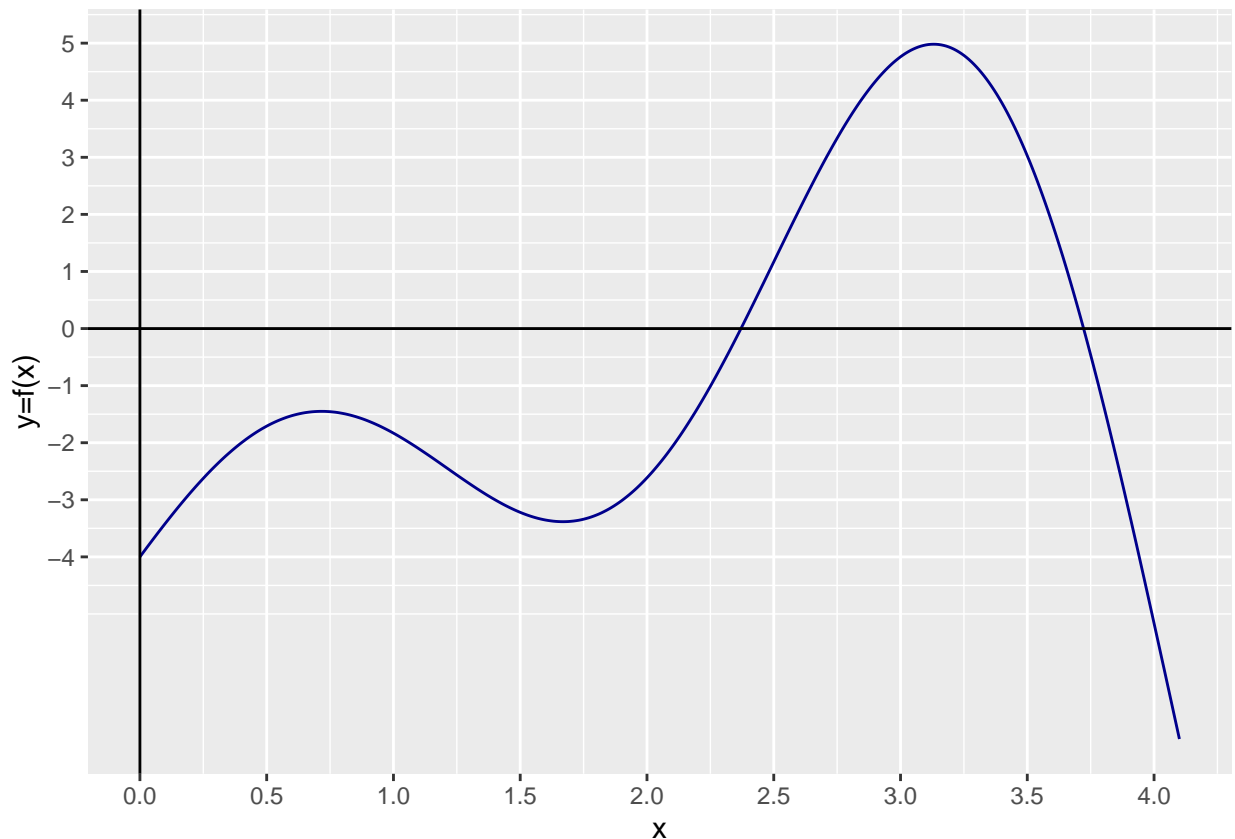
### ##Ejercicio 3

```
f=function(x){  
  2*x*cos(2*x)-(x-2)^2  
}
```

Grafico en el intervalo  $[2;3]$  y  $[3;4]$

```
x<-seq(0,4.1,0.01)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos  
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables  
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea  
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0  
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,4,0.5)) #cambio escala eje X  
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-4,6,1)) #Cambio escala eje Y
```

ggfx



Aplico el metodo de posicion falsa en ambos intervalos

```
paste("Las raices se encuentran en: ",posicion_falsa(f,2.25,2.5,0.0001,100)
      , ", y en :",posicion_falsa(f,3.5,3.75,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 2.36598201309748 , y en : 3.71685326876732"
```

```
##Ejercicio 4
```

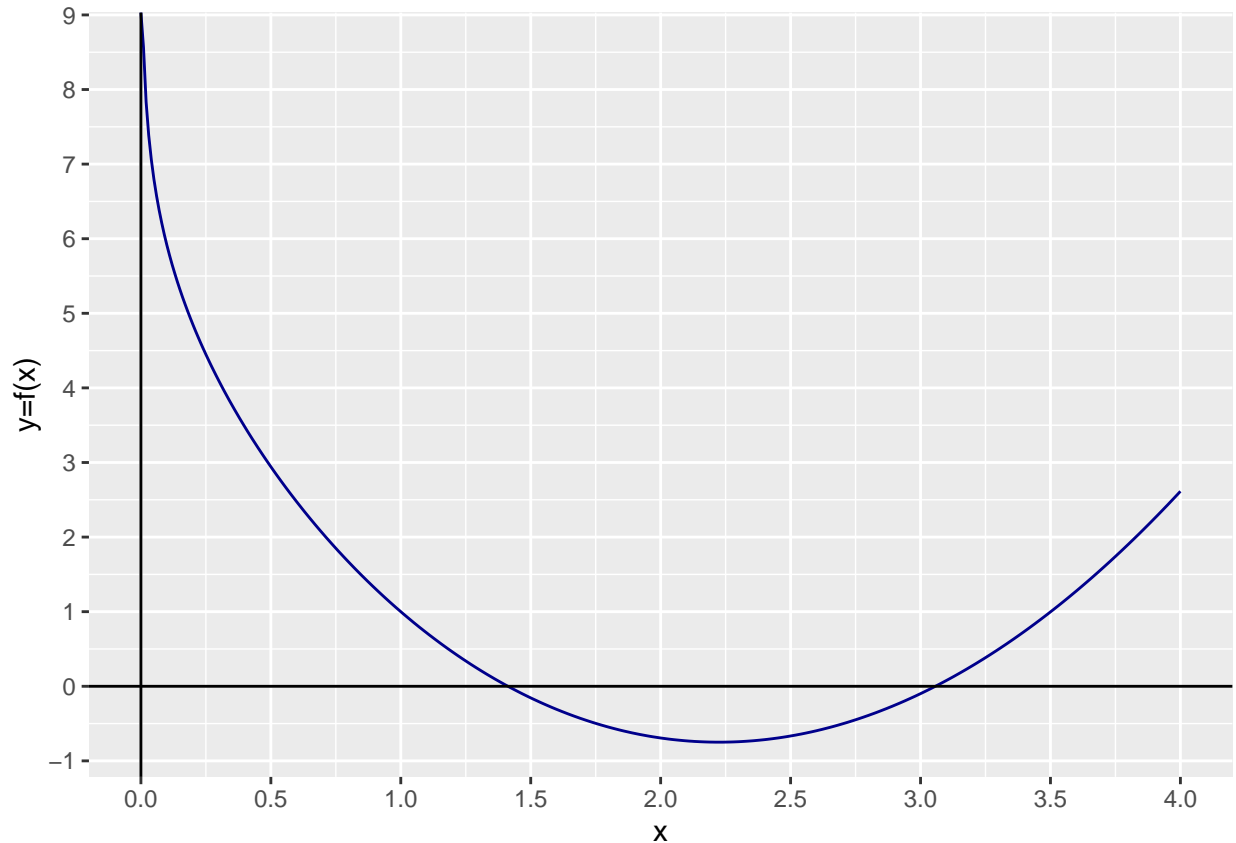
```
f=function(x){
  (x-2)^2-log(x)
}
```

Grafico en el intervalo  $[1; 2]$  y  $[e; 4]$

```
x<-seq(0,4,0.01)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-1,10,1)) #Cambio escala eje Y

ggfx
```



Aplico el metodo de posicion falsa en ambos intervalos

```
paste("Las raices se encuentran en: ",posicion_falsa(f,1.25,1.5,0.0001,100)
      , ", y en :",posicion_falsa(f,3,3.25,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 1.41239996269939 , y en : 3.0510989730265"
```

##Ejercicio 5

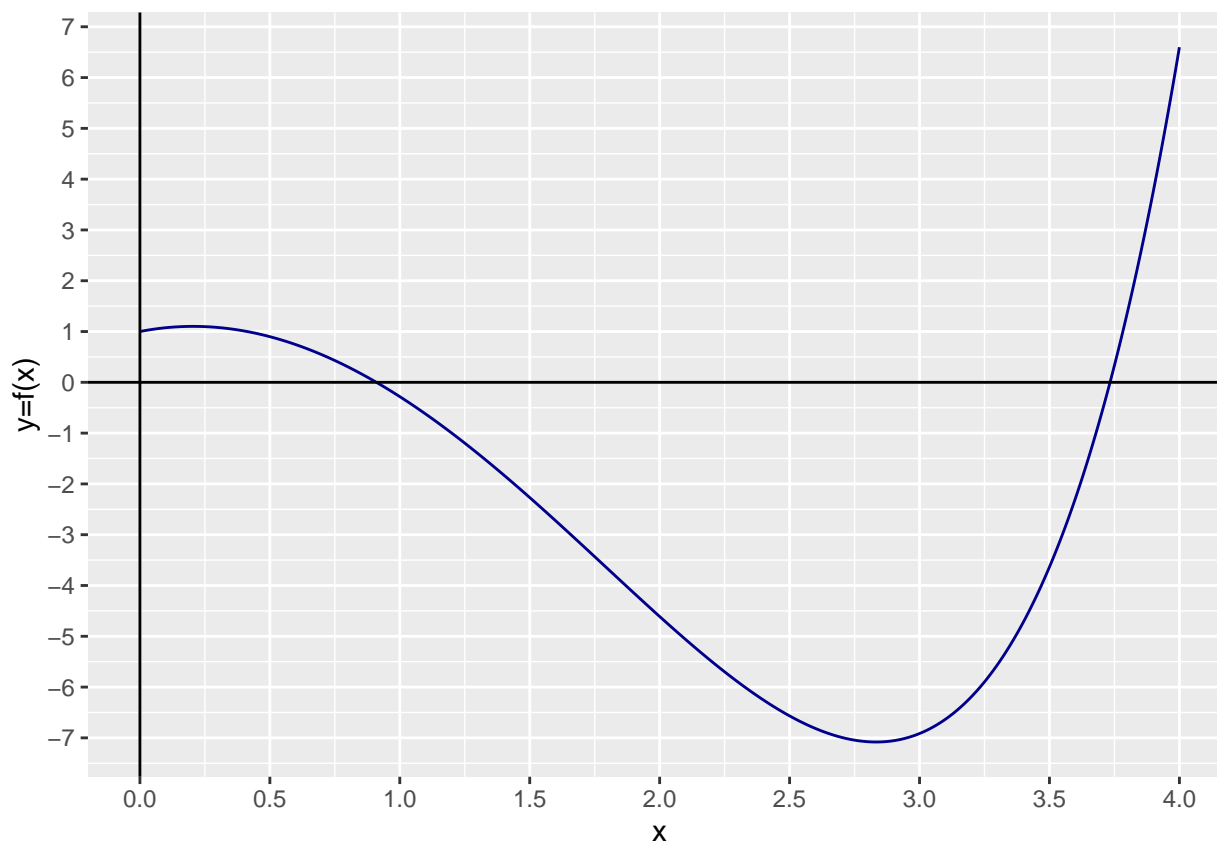
```
f=function(x){
  exp(x)-3*x^2
}
```

Grafico en el intervalo [0; 1] y [3; 5]

```
x<-seq(0,4,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-8,8,1)) #Cambio escala eje Y
```

ggfx



Aplico el metodo de posicion falsa en ambos intervalos

```
paste("Las raices se encuentran en: ",posicion_falsa(f,0.75,1,0.0001,100)  
      , " , y en :",posicion_falsa(f,3.5,4,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 0.900973366460047 , y en : 3.67759480514636"
```

##Ejercicio 6

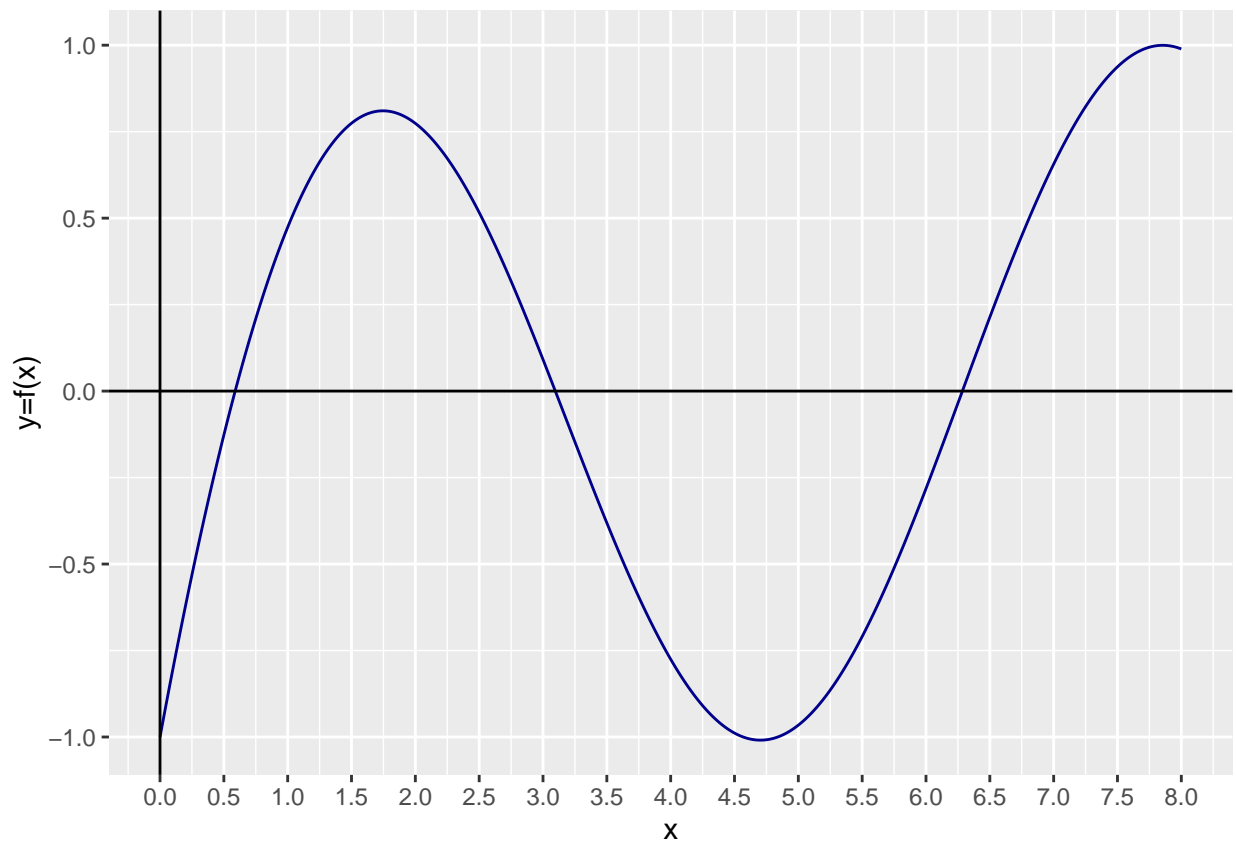
```
f=function(x){  
  sin(x)-exp(-x)  
}
```

Grafico en el intervalo [0; 1] , [3; 4] y [6; 7]

```
x<-seq(0,8,0.001)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe
```

```
ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego línea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,10,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.5)) #Cambio escala eje Y
```

ggfx



Aplico el metodo de posicion falsa en todos intervalos

```
paste("Las raices se encuentran en: ",posicion_falsa(f,0.5,0.75,0.0001,100),", en :", posicion_falsa(f,3.0,3.25,0.0001,100),", y en :",posicion_falsa(f,6.25,6.5,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 0.588532977490916 , en : 3.09581626632448 , y en : 6.285049240383"
```

##Ejercicio 7

```
f=function(x){
  cos(x)-sqrt(x)
}
```

Grafico en el intervalo [0; 2]

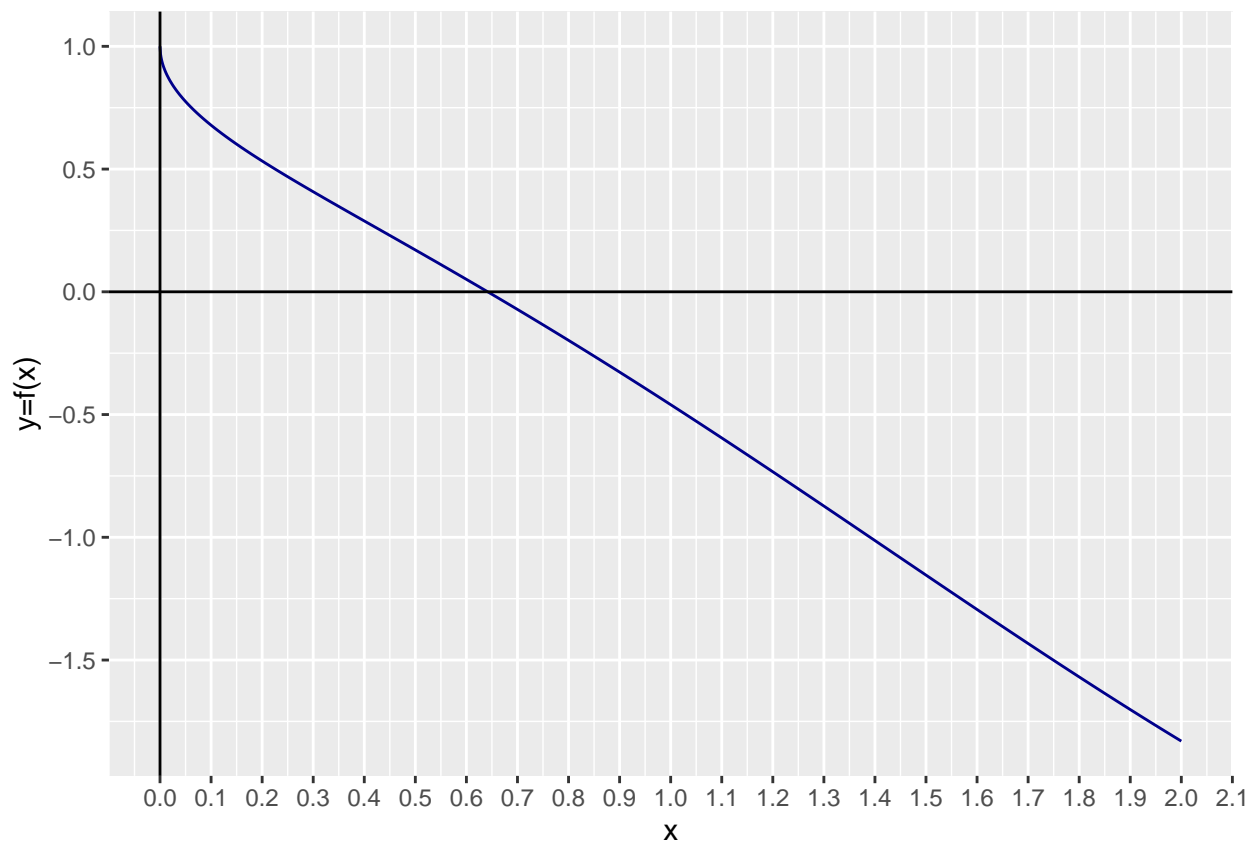
```

x<-seq(0,2,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,3,0.1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-3,2,0.5)) #Cambio escala eje Y

```

ggfx



Aplico el metodo de posicion falsa en el intervalo

```
paste("La raiz se encuentra en: ",posicion_falsa(f,0.6,0.7,0.0001,100))
```

```
## [1] "La raiz se encuentra en: 0.641400356334603"
```

##Ejercicio 8



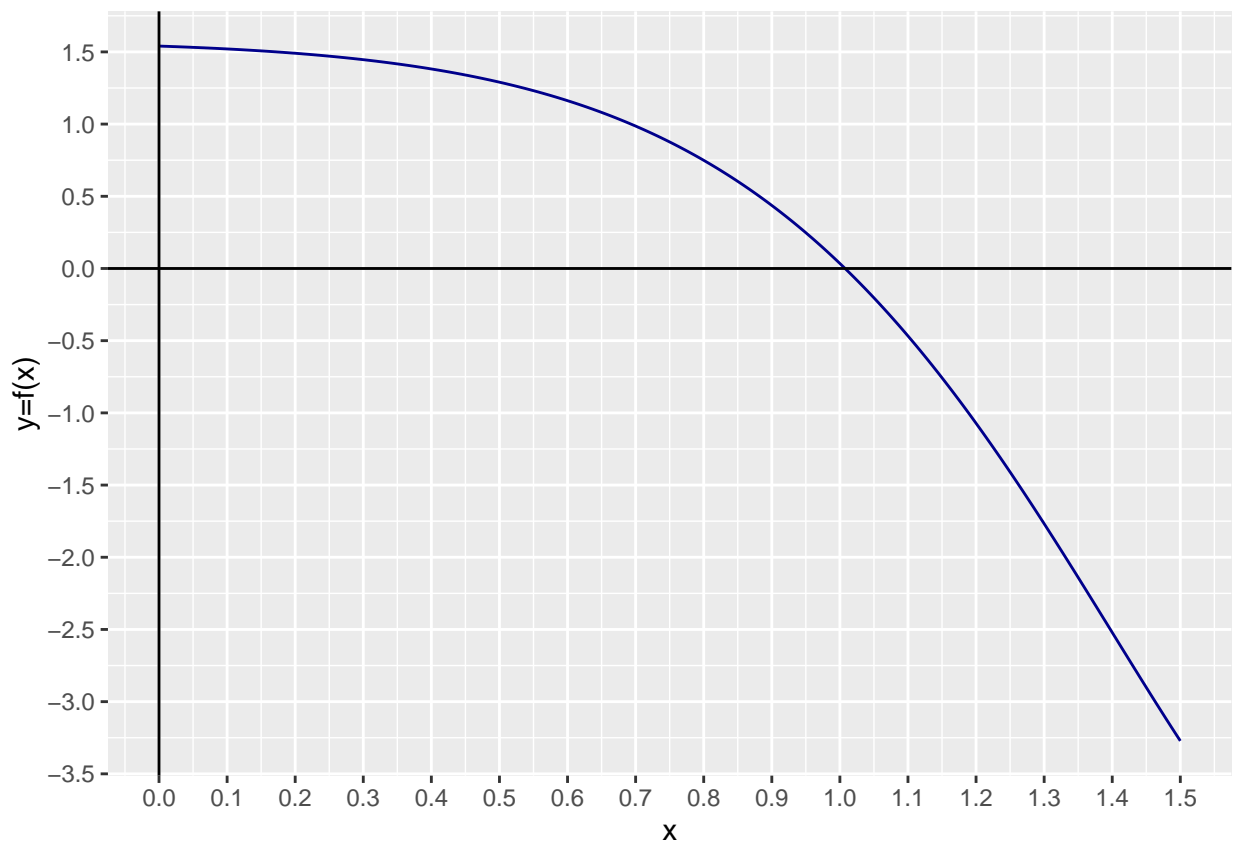
```
f=function(x){
  2+cos(exp(x)-2)-exp(x)
}
```

Grafico en el intervalo  $[0; 1.5]$

```
x<-seq(0,1.5,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,2,0.1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-4,2,0.5)) #Cambio escala eje Y
```

ggfx



Aplico el metodo de posicion falsa en el intervalo

```
paste("La raiz se encuentra en: ",posicion_falsa(f,0.9,1.1,0.0001,100))
```

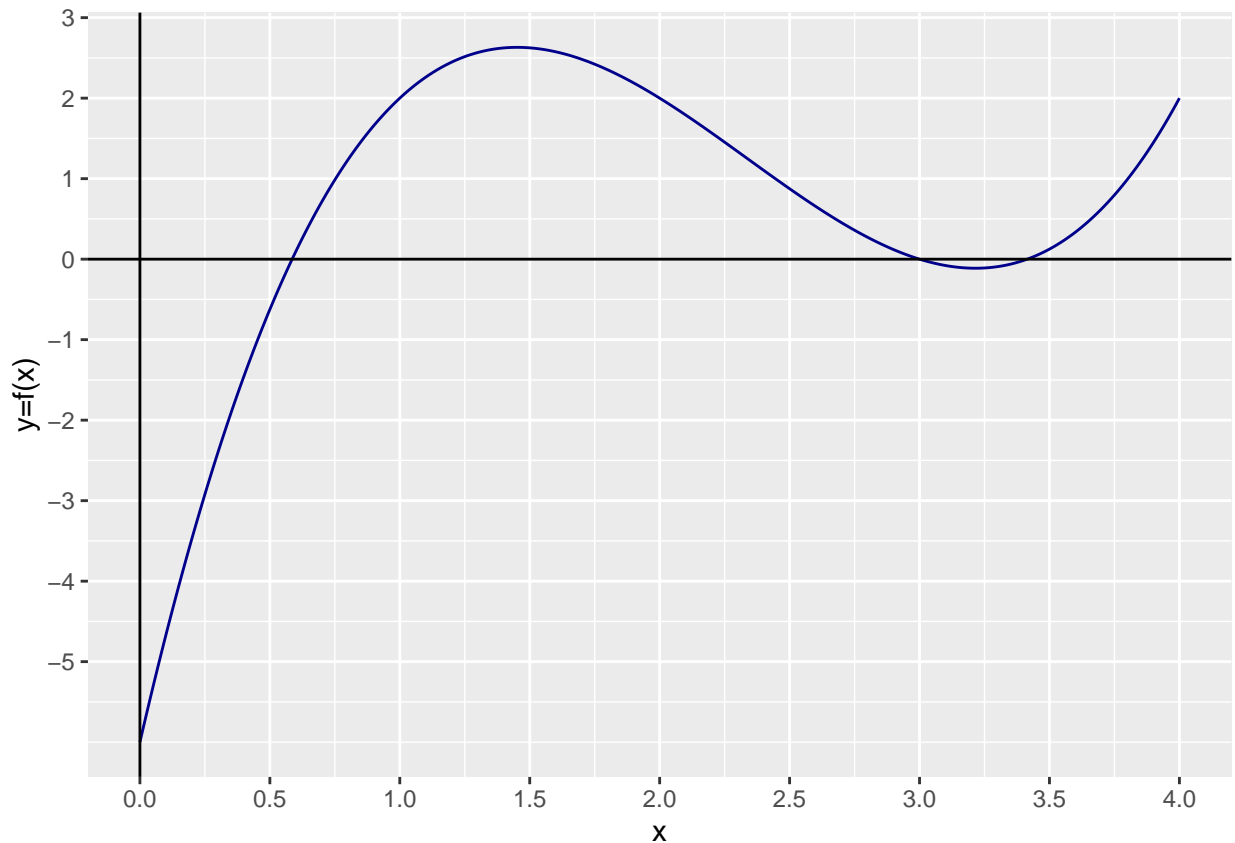
```
## [1] "La raiz se encuentra en: 0.996598618574031"
```

```
##Ejercicio 9
```

```
f=function(x){  
  x^3-7*x^2+14*x-6  
}
```

Grafico en el intervalo  $[0; 4]$

```
x<-seq(0,4,0.001)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos  
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables  
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea  
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0  
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(0,5,0.5)) #cambio escala eje X  
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-5,5,1)) #Cambio escala eje Y  
  
ggfx
```



Aplico el metodo de posicion falsa en el intervalo

```
paste("Las raices se encuentran en: ",posicion_falsa(f,0.5,0.75,0.0001,100)," , en :", posicion_falsa(f,1.5,1.75,0.0001,100), " , y en :",posicion_falsa(f,3.25,3.5,0.0001,100))
```

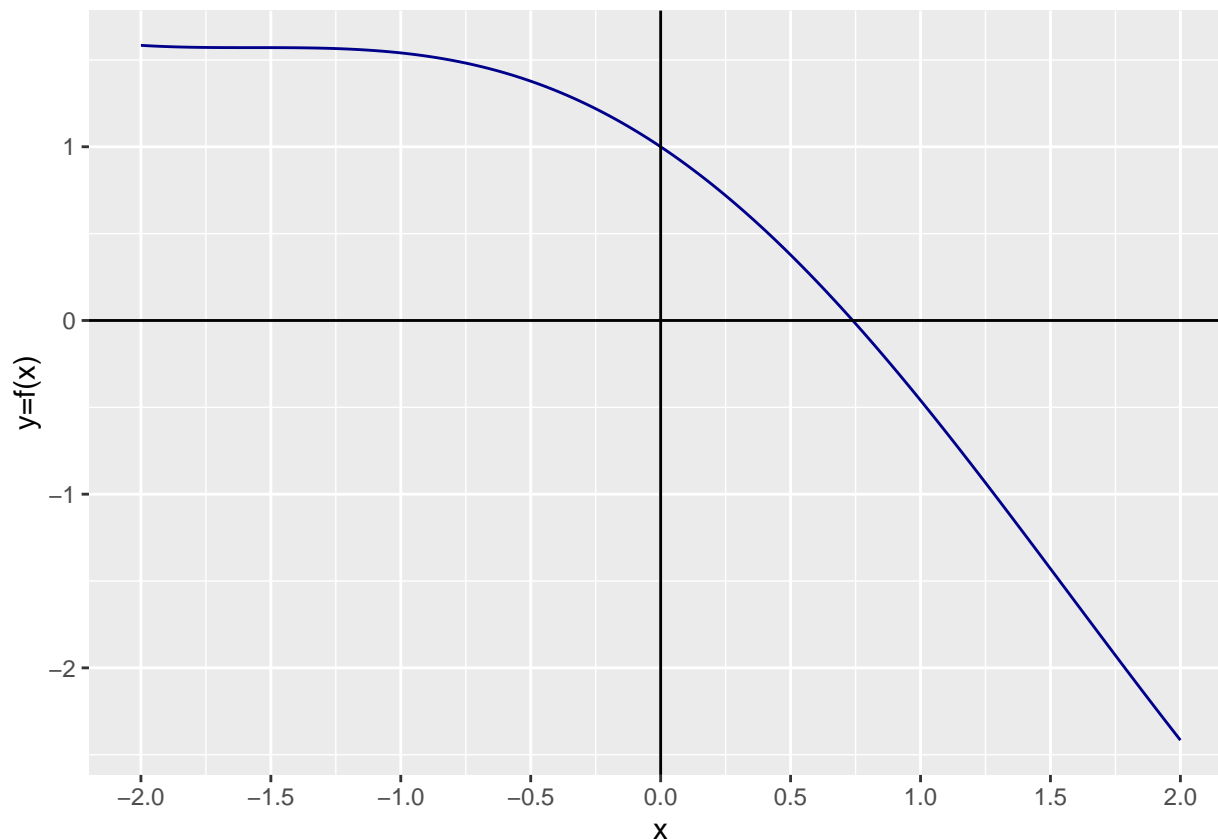
```
## [1] "Las raices se encuentran en: 0.585789247179731 , en : 3.00001479709664 , y en : 3.36666666666667"
```

##Ejercicio 10

```
f=function(x){  
  cos(x)-x  
}
```

Grafico en el intervalo  $[-2; 2]$

```
x<-seq(-2,2,0.001)#Genero vector para graficar f(x)  
fx<-f(x)  
df<-data.frame(x,fx) #Creo dataframe  
  
ggfx=ggplot(data=df) #carga los datos  
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables  
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea  
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0  
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-3,3,0.5)) #cambio escala eje X  
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-3,2,1)) #Cambio escala eje Y  
  
ggfx
```



Aplico el metodo de posicion falsa en el intervalo

```
paste("Las raices se encuentran en: ",posicion_falsa(f,0.5,1,0.0001,100))
```

```
## [1] "Las raices se encuentran en: 0.72548158706393"
```

##Ejercicio 11

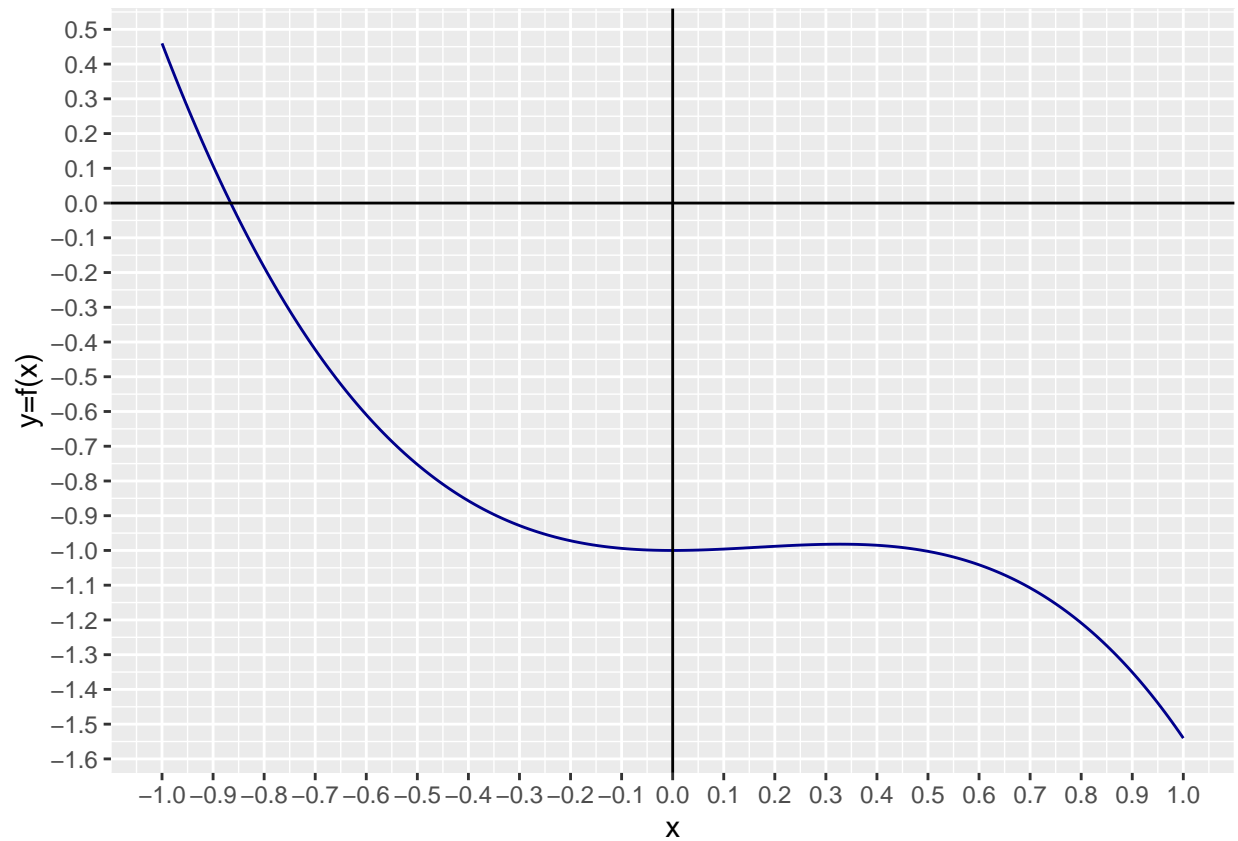
```
f=function(x){
  -x^3-cos(x)
}
```

Grafico en el intervalo  $[-1; 1]$

```
x<-seq(-1,1,0.001)#Genero vector para graficar f(x)
fx<-f(x)
df<-data.frame(x,fx) #Creo dataframe

ggfx=ggplot(data=df) #carga los datos
ggfx=ggfx+aes(x=x,y=fx)#Cargo variables
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-1,1,0.1)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(-2,2,0.1)) #Cambio escala eje Y

ggfx
```



Aplico el metodo de posicion falsa en el intervalo

```
paste("Las raices se encuentran en: ",posicion_falsa(f,-0.9,-0.8,0.0001,100))
```

```
## [1] "Las raices se encuentran en: -0.865471627305148"
```