

Ejercicios Punto fijo

Agustin Huczok

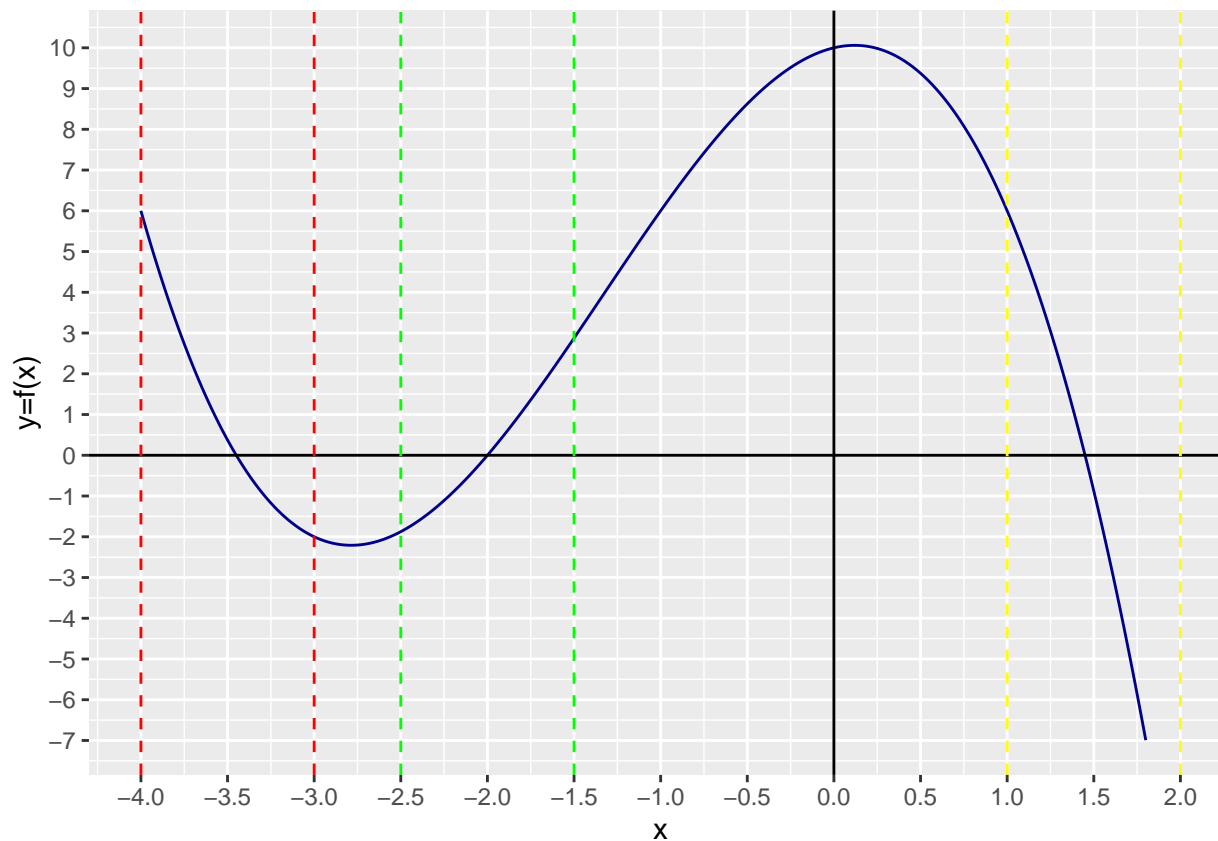
9/9/2021

#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
```

```

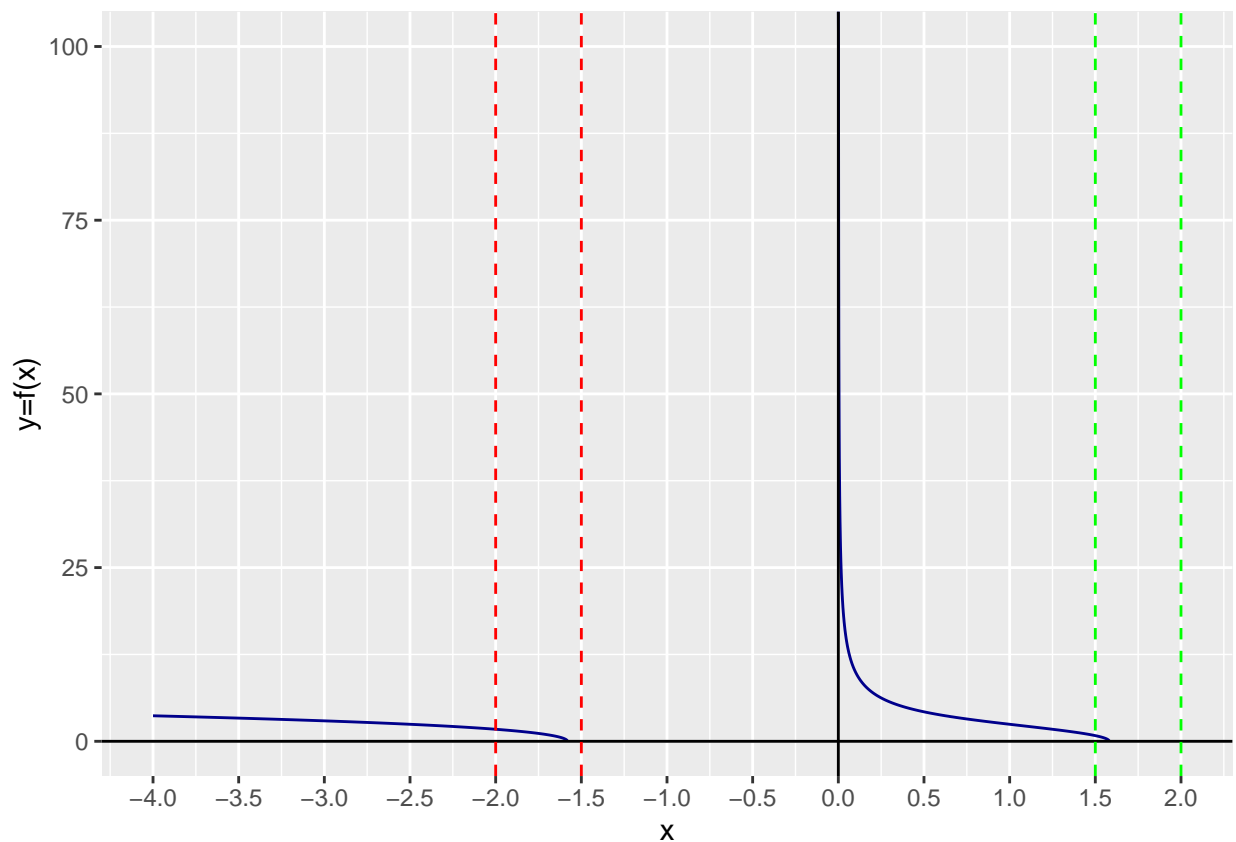
ggfx=ggfx+geom_line(linetype=1,colour="darkblue") #Agrego linea
ggfx=ggfx+geom_hline(yintercept=0,linetype=1)+geom_vline(xintercept = 0,linetype=1)#Creo x=0 e y=0
ggfx=ggfx+scale_x_continuous(name="x",breaks=seq(-4,2,0.5)) #cambio escala eje X
ggfx=ggfx+scale_y_continuous(name="y=f(x)",breaks=seq(0,100,25))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

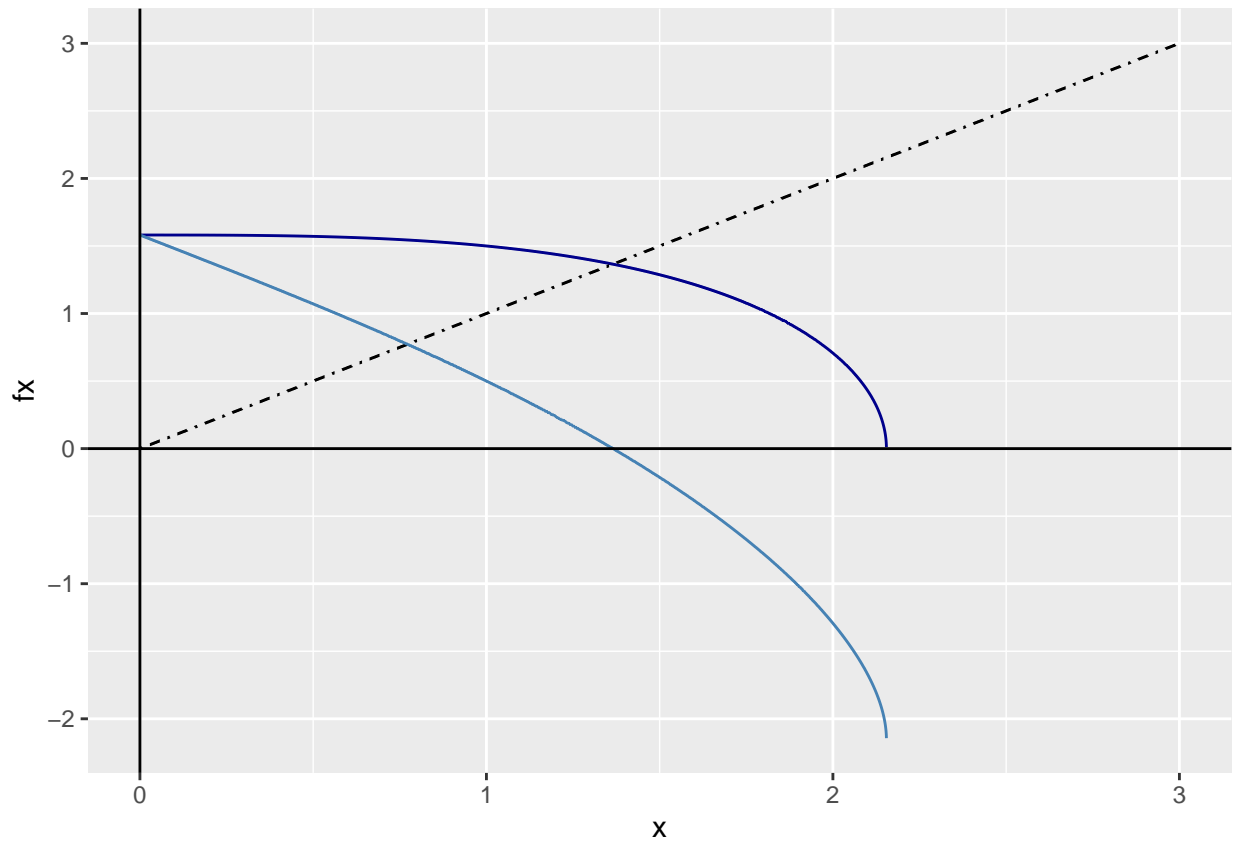
```
## Scale for 'x' is already present. Adding another scale for 'x', which will
## replace the existing scale.
```

```
## 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)
```

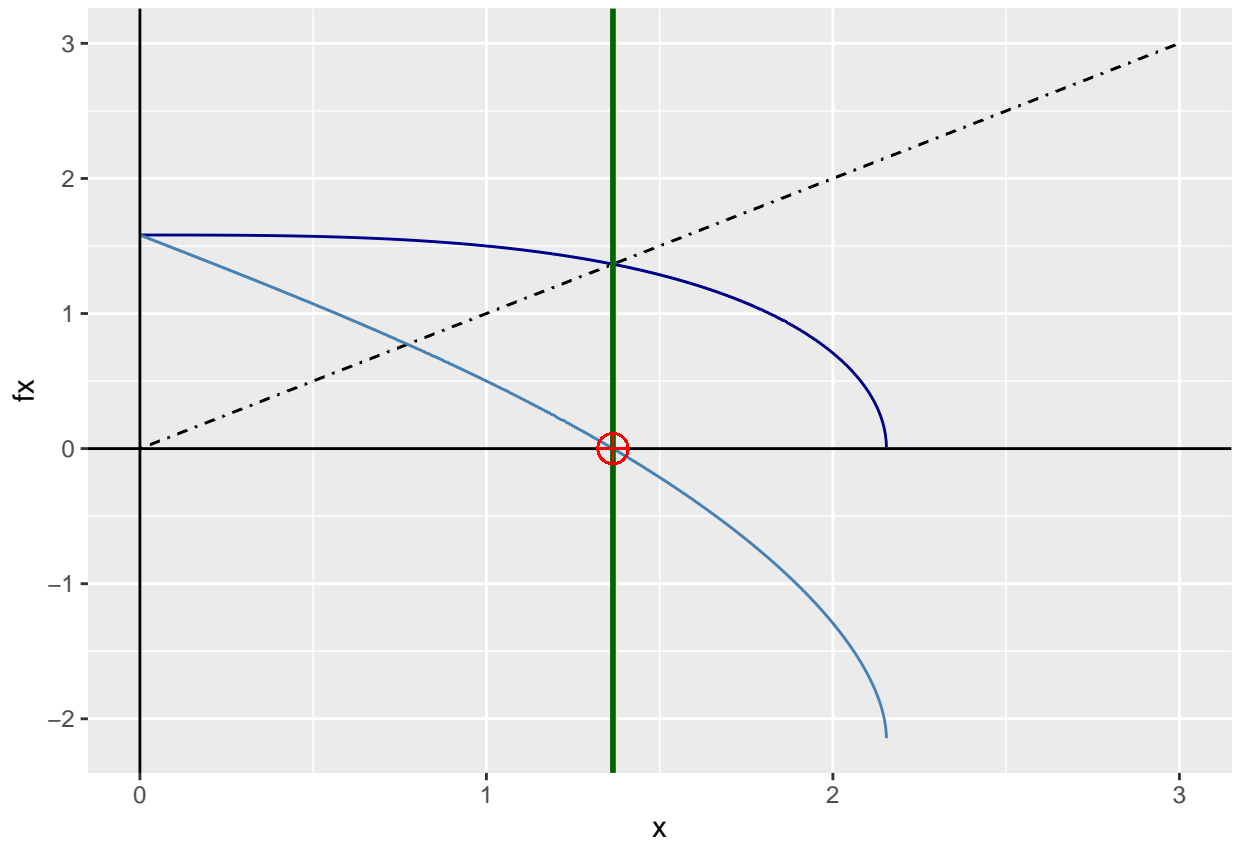
```
## [1] 1.365206
```

```
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 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)
#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.

```

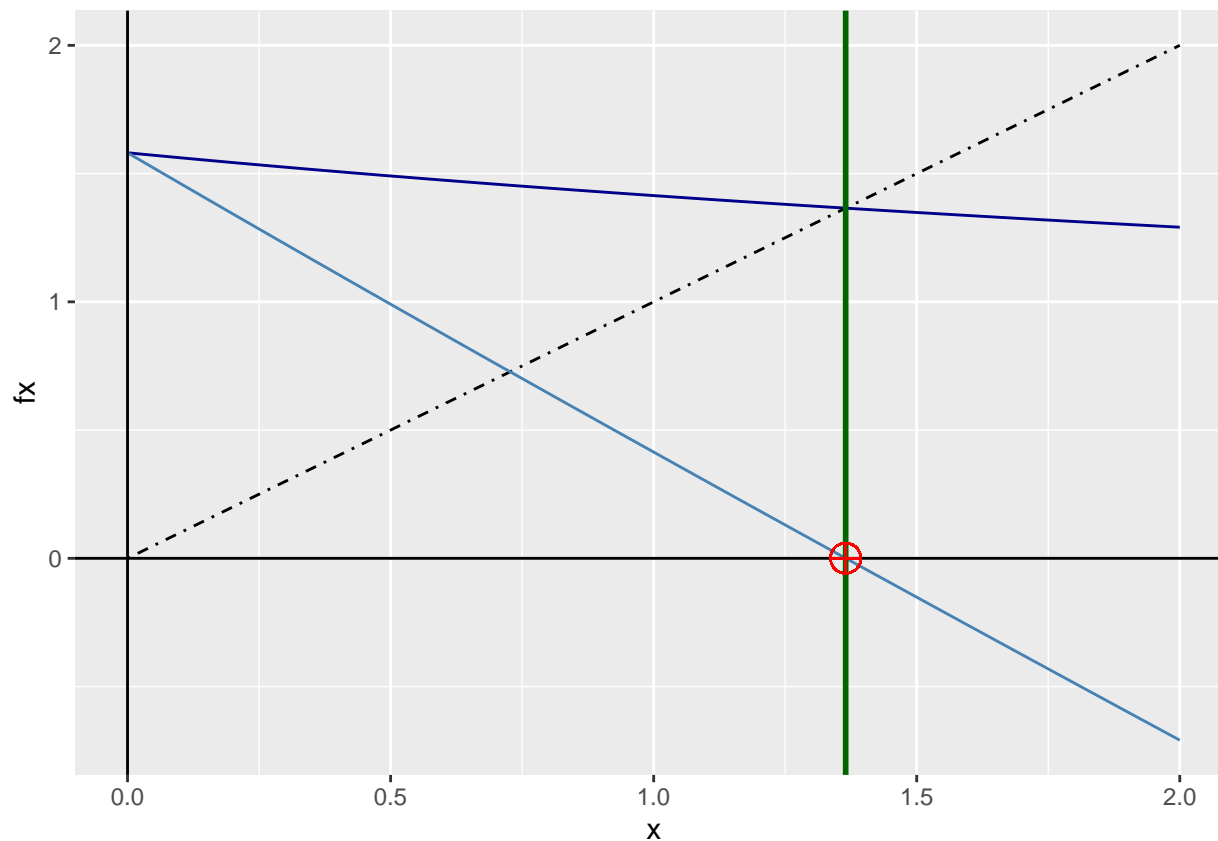
```
gggx
```

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

```

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)
```

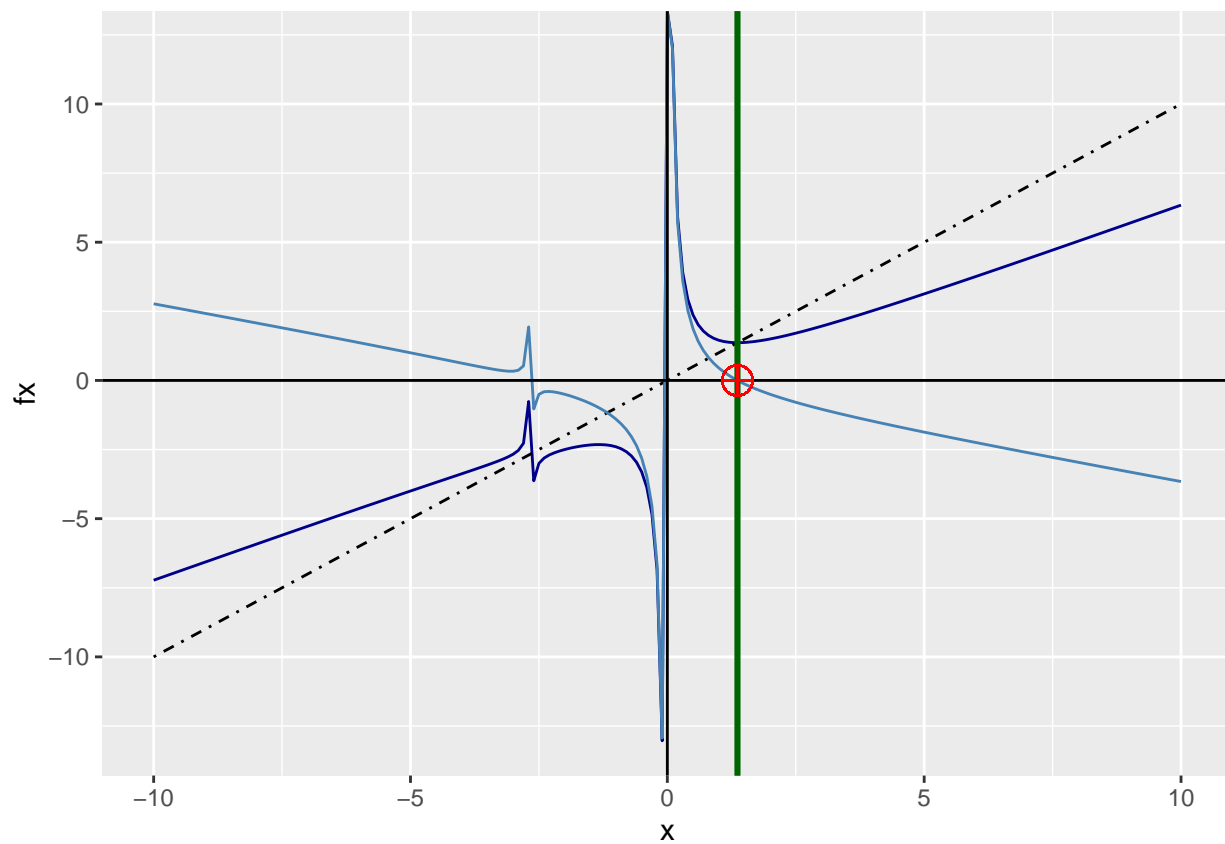
```
gggx
```

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

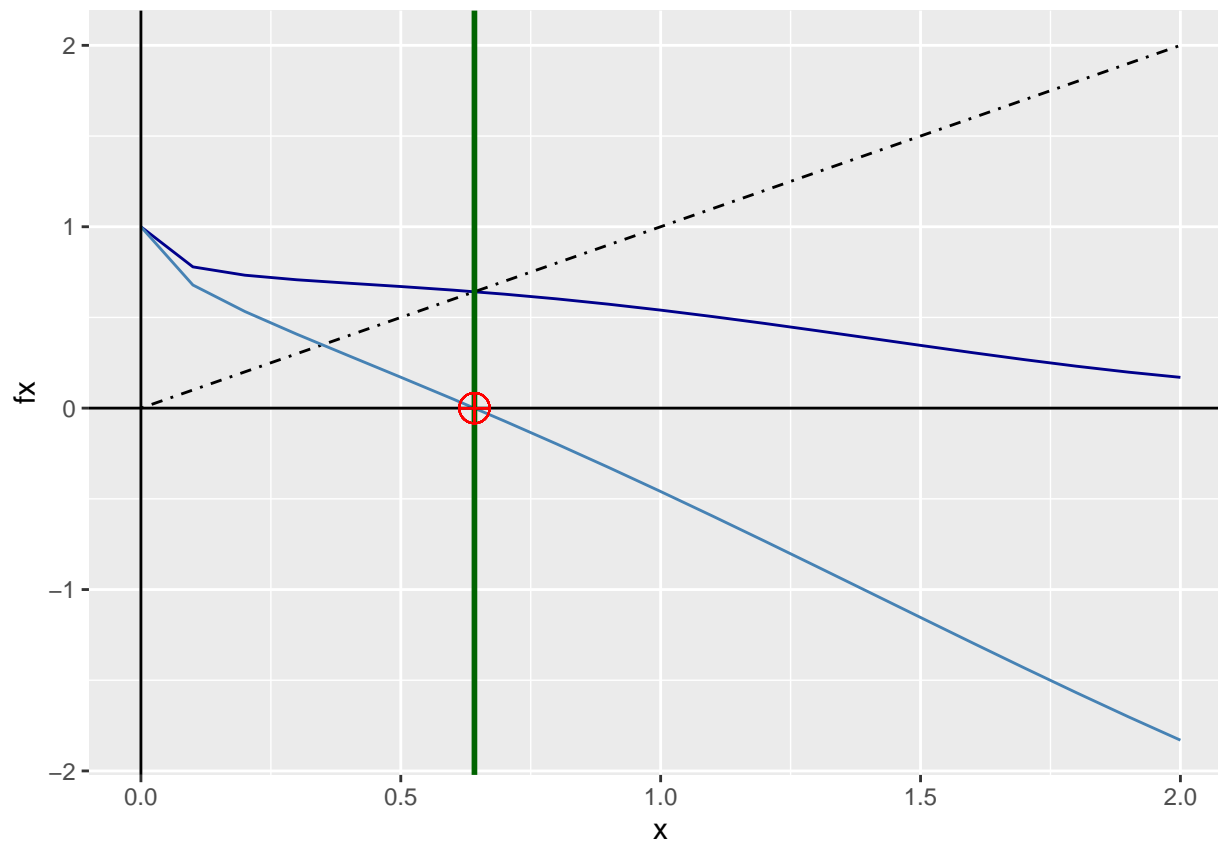
```
f=function(x){  
  -(x)^0.5+cos(x)+x  
}  
g=function(x){  
  -(x)^0.5+cos(x)  
}
```

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")
```

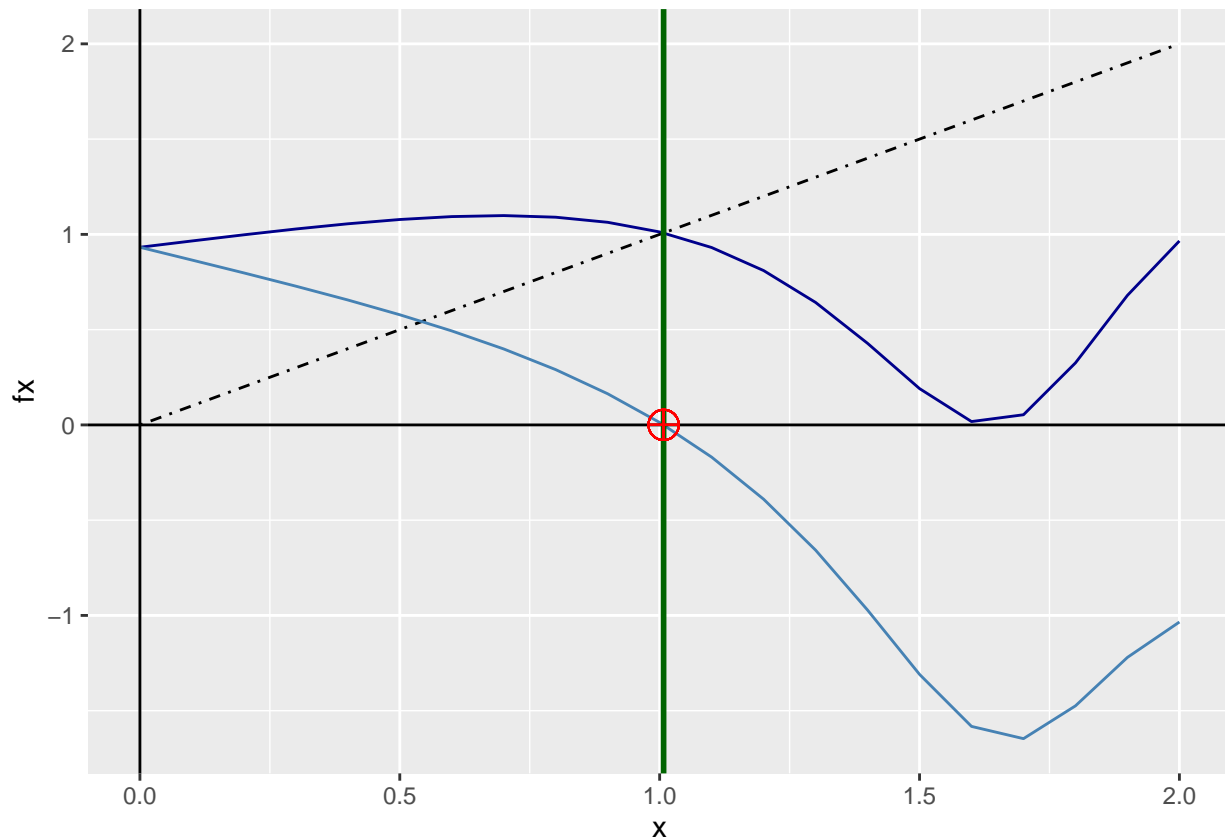
```
#Gráfico la función del ejercicio donde esta la raíz
gggx <- gggx + geom_line(aes(x = x, y = g(x)), colour = "steelblue")
#Agrego el eje X e Y
gggx <- gggx + geom_vline(xintercept = 0, linetype = 1)+
  geom_hline(yintercept = 0, linetype = 1)
```

```
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

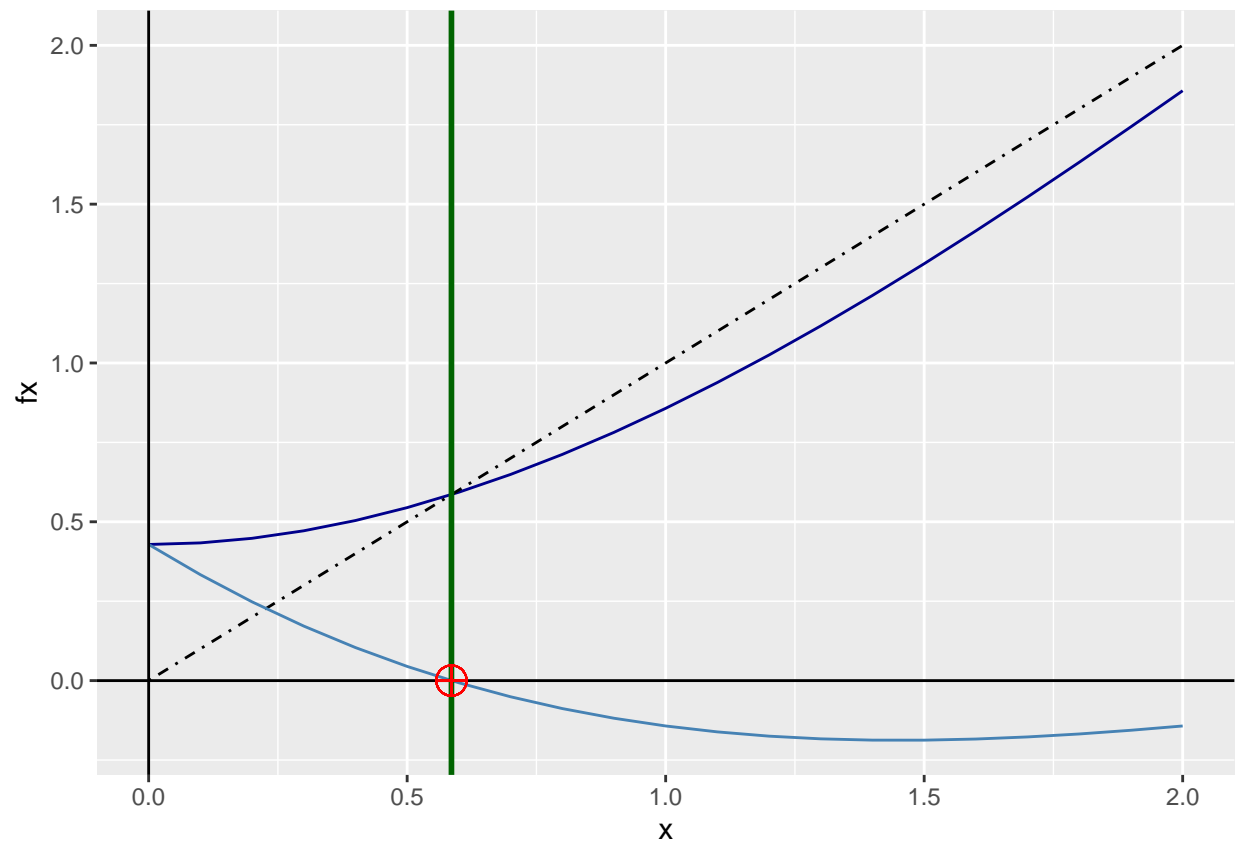
```
graphics.off()
x <- seq(0, 2, by = 0.1) #Genero vector para graficar
fx <- f(x)
df <- data.frame(x, fx) #Creo dataframe
gggx <- ggplot(data = df) #Cargo los datos
gggx <- gggx + aes(x = x, y = fx) #Cargo las variables
gggx <- gggx + geom_line(linetype = 1, colour = "darkblue")
#Gráfico x = y
gggx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