

Lab4

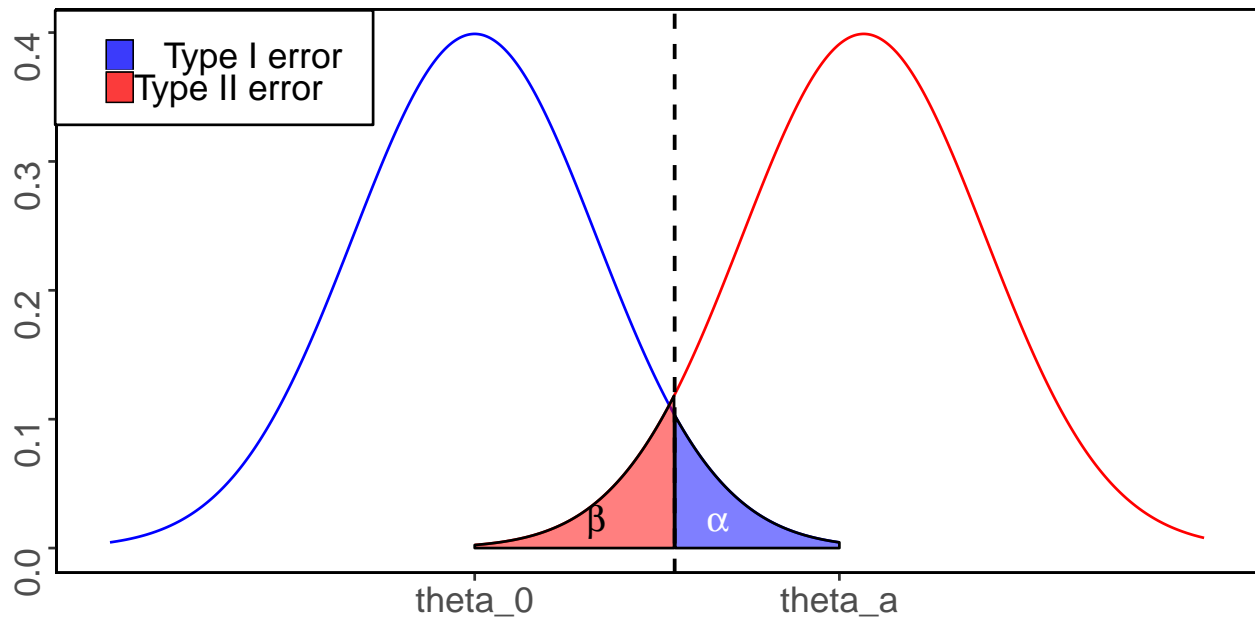
Lingchen Lou

2018/10/10

Q1 Write the R code using ggplot function to recreate the graph from the Lecture 8 notes

```
library(ggplot2)
x1=seq(-3,3, by=0.01)
y1=dnorm(x1)
x2 = seq(0,6, by=0.01)
y2 = dnorm(x2, 3.2, 1)
x = c(x1,x2)
y= c(y1,y2)
pdf_df1<-data.frame(x1,y1)
pdf_df2<-data.frame(x2,y2)
p1 <- ggplot()+
  geom_line(aes(x=x1, y= y1),col="blue")+stat_function(fun = dnorm,args = list(0,1))+
  geom_line(aes(x=x2, y= y2),col="red")+stat_function(fun = dnorm,args = list(3.2,1))+
  #colored area under curve
  geom_ribbon(data = pdf_df1[pdf_df1$x1 > qnorm(0.95),],
    aes(x = x1, ymin = 0, ymax = y1, fill = "Type I error"), alpha=0.5, colour="black")+
  geom_ribbon(data = pdf_df2[pdf_df2$x2 < qnorm(0.95),],
    aes(x = x2, ymin = 0, max = y2, fill = "Type II error"), alpha=0.5, colour="black")+
  #change the filling color
  scale_fill_manual(values=c("blue", "red"))+
  #add the verticle line
  geom_vline(xintercept = qnorm(0.95), linetype = 2, size =0.7)+
  #annotate alpha and beta
  annotate("text", x= c(1,2), y= 0.02,
    label = c(paste0(expression(beta)),paste0(expression(alpha))),
    color = c("black","white"),size =5, parse = TRUE)+
  #set up for background of coordinate axis
  theme(panel.background =element_blank(),
    panel.border = element_rect(color = 'black',fill=NA,size = 1),
    axis.text.y = element_text(size = 14, angle = 90),
    axis.text.x = element_text(size = 14, angle = 0),
    axis.title = element_blank(),
    legend.title = element_blank())+
  theme(legend.background = element_rect(fill="white",color = "black",size=0.5),
    legend.text = element_text(colour="black",size = 14),
    legend.key.width = unit(4, "mm"),
    legend.key.height = unit(1, "mm"),
    legend.text.align = 3,
    legend.spacing = unit(4, "mm"),
    legend.margin=margin(t = 4,b= 3,l=7,r=7, unit='mm'),
    legend.position = c(0.132, 0.895))+
  scale_x_continuous(breaks = c(0,3),
    label=c(paste0(expression(theta_0)),paste0(expression(theta_a))))
```

p1



Q2 Implement a function that will check if a given positive integer is a prime number.

```
is.prime<-function(x){
  if( x %% 1 ==0){
    if(x<2){
      print(paste(x,"is not a prime number"))
    }
    else{
      n = 1
      for(i in 2:(x-1)){
        if ((x %% i) == 0 ){
          n = 0
          break
        }
      }
    }
    if ( x == 2){
      n =1
    }
    if(n == 1){
      print(paste(x, "is a prime number"))
    }
    else{
      print(paste(x, "is not a prime number"))
    }
  }
}
else{
  print(paste(x,"is not valid"))
}
}
```

```
#check
is.prime(1)

## [1] "1 is not a prime number"
is.prime(2)

## [1] "2 is a prime number"
is.prime(1.5)

## [1] "1.5 is not valid"
is.prime(9)

## [1] "9 is not a prime number"
```