RD Robustness Project

Exercise: stressout before replacing with zeros

Maor Milgrom

2020-03-21

Introduction

The goal of this project is to test the robustness of the Regression-Discontinuity analysis to different extreme cases, via simulations, using the 'rdrobust' package.

This Document: Excercise #2

Stressing out results, by adding noise inside Optimal Bandwitdh (OBW), before repeating exercise #1.

Select exercise type:

```
exercise="zero"      # Type of exercise - "zero" or "bwo" ('bandwidth only')
```

Set Parameters

Here we set the main parameters for the excercise:

```
jump=10            # Size of jump at cutoff
loop=1000
figs.iter.save=5
quadratic=T        # T - quadratic DGP, F - linear
symm_obw="mserd"   # mserd - symmetric OBW, msetwo - asymmetric OBW
normal.x=T         # T - normal draws of x around cutoff, F - uniform draws
dgp.sd=10          # sd of normal noise added to DGP
noisy.sd=30        # sd of noise added inside OBW
so.int=2           # interval inside OBW for adding noise
bc=F               # bias-corrected estimates or conventional
```

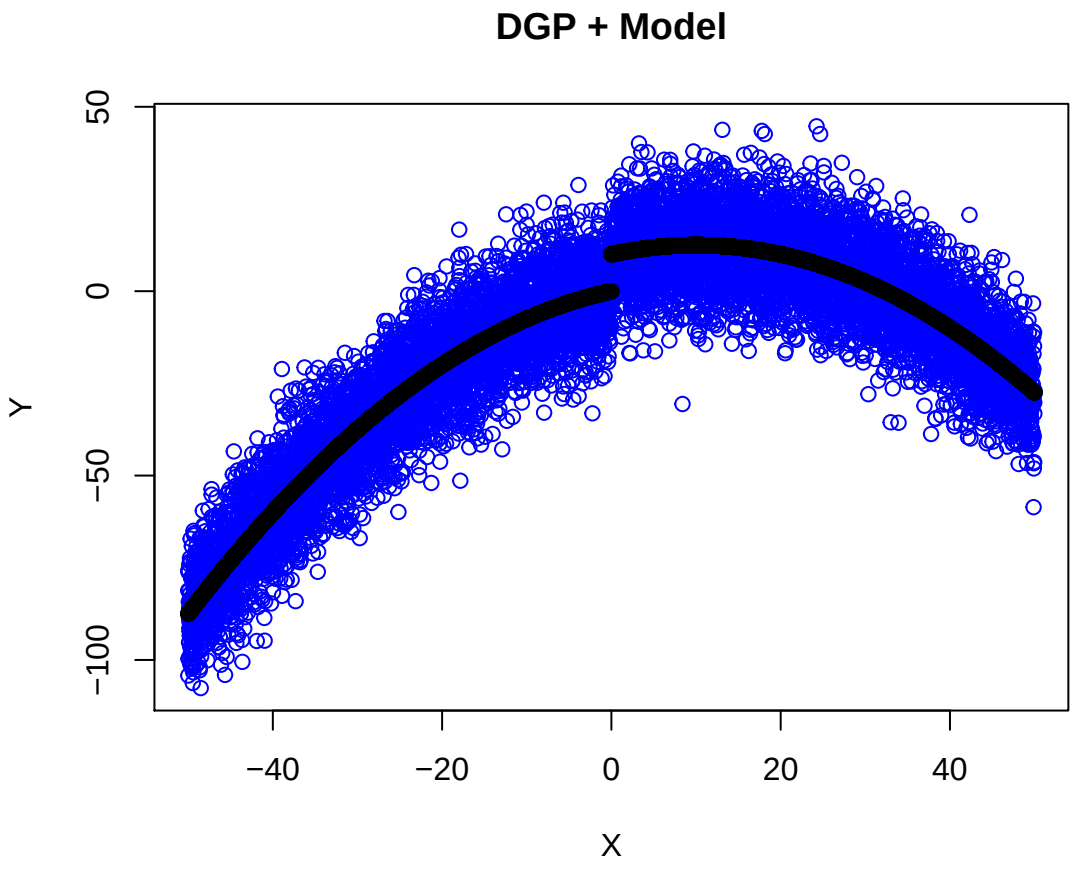
Constructing dataframes for simulation and for results

```
df <- as.data.frame(matrix(0, ncol = 0, nrow = length(seq(-100,100,0.01))))
df$x=round(seq(-100,100,0.01), digits=2)
df=subset(df,df$x!=0)
df$treated <- ifelse(df$x>0, 1, 0)
sample.x <- as.data.frame(matrix(0, ncol = 0, nrow = nrow(df)/10))

results<- as.data.frame(matrix(0, ncol = 10, nrow = loop))
colnames(results) <- c("coef","obw","obw_l","obw_r",
                       "coef_so","obw_l_so","obw_r_so","obw_so",
                       "coef_diff","obw_diff")
```

```
results.exercise<- as.data.frame(matrix(0, ncol = 8, nrow = loop))
colnames(results.exercise) <- c("coef","obw","obw_l","obw_r",
                                "coef_diff","obw_diff",
                                "coef_diff_so","obw_diff_so")
```

Simulate DGP and plot

```
df$y.model<- 0.5*df$x  - 0.025*df$x^2*quadratic + jump*df$treated
df$y=df$y.model+rnorm(length(df$x),0,dgp.sd)
df$y.noisy=df$y+rnorm(length(df$x),0,noisy.sd)

df %>%
  filter(x > -50 & x < 50) %T>%
  plot(y~x,., ylim = range(c(y,y.model)),
       col="blue", ylab = "Y", xlab = "X") %T>%
  par(new = T) %>%
  plot(y.model~x,., ylim = range(c(y,y.model)),
       axes = FALSE, xlab = "", ylab = "")
  title(main = "DGP + Model")
```



Iterations

Running 1000 iterations, and saving figures from 5 last iterations to file

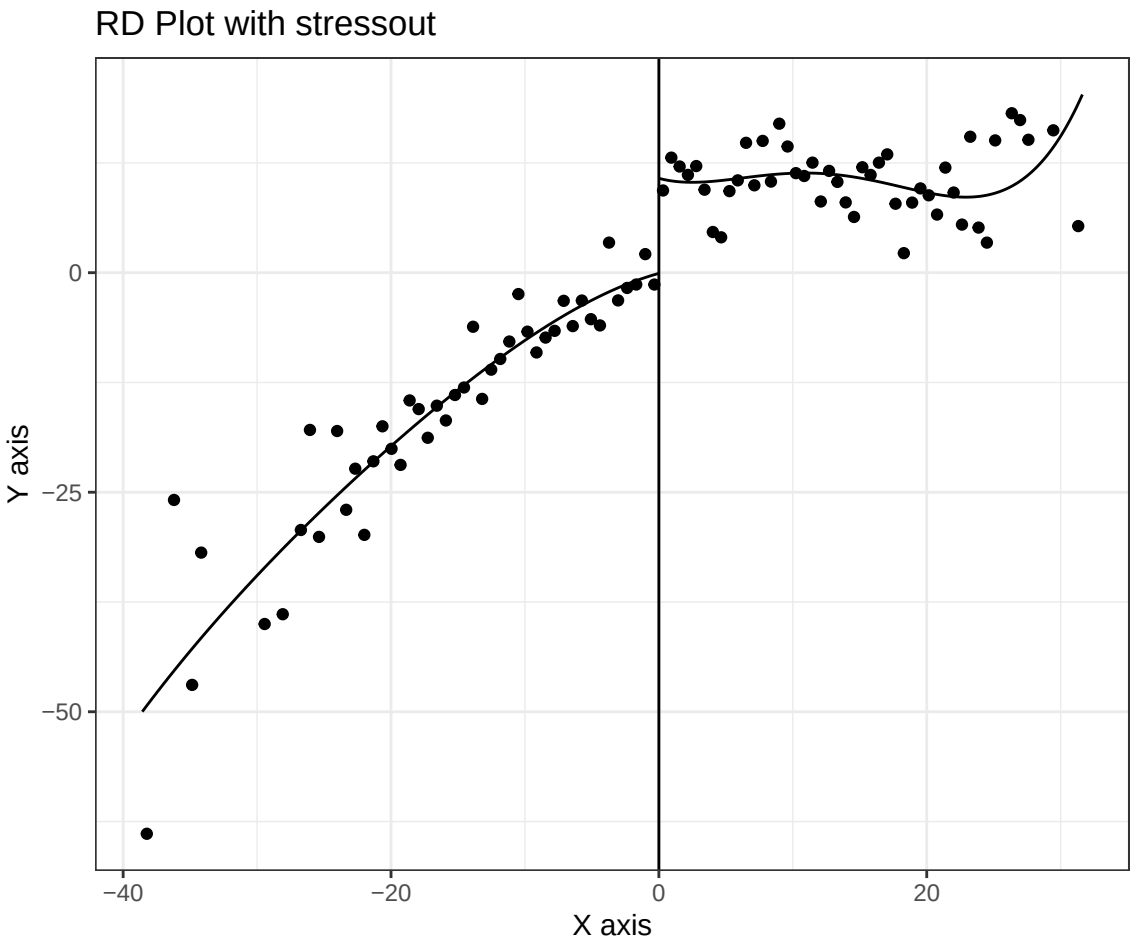
```
for(i in 1:loop) {
  if (normal.x==T) {
    sample.x$x <- round(rnorm(nrow(df)/10, 0, 10),digits = 2)
  } else {
    sample.x$x <- round(runif(nrow(df)/10, -20,20),digits = 2)
  }

  sample.x=subset(sample.x, x>-100 & x<100)
  sample=as.data.frame(inner_join(df, sample.x, by="x"))
  results.current=rdrobust(sample$y,sample$x,bwselect = symm_obw)
  results[i,1]=results.current$coef[bc+1]-jump # normalizing to zero
  results[i,3:4]=results.current$bws[bc+1,1:2]
  results[i,2]=results[i,3]+results[i,4]

  sample$y.so <- ifelse((sample$x > results[i,4]-so.int & sample$x < results[i,4]) |
                          (sample$x < -results[i,3]+so.int & sample$x > -results[i,3]),
                        sample$y.noisy,sample$y)
  results.current=rdrobust(sample$y.so,sample$x,bwselect = symm_obw)
  results[i,5]=results.current$coef[bc+1]-jump # normalizing to zero
  results[i,7:8]=results.current$bws[bc+1,1:2]
  results[i,6]=results[i,7]+results[i,8]
  results[i,9:10]=results[i,1:2]-results[i,5:6]

  if (exercise=="zero") {
    sample.exercise=sample
    sample.exercise$y.so <- ifelse(sample$x> -results[i,7] &
                                   sample$x< results[i,8], sample$y.so, 0)

  } else if (exercise=="bwo") {
    sample.exercise=subset(sample,x> -results[i,7] & x< results[i,8])
  }

    results.current=rdrobust(sample.exercise$y.so,sample.exercise$x,bwselect = symm_obw)
    results.exercise[i,1]=results.current$coef[bc+1]-jump # normalizing to zero
    results.exercise[i,3:4]=results.current$bws[bc+1,1:2]
    results.exercise[i,2]=results.exercise[i,3]+results.exercise[i,4]
    results.exercise[i,5:6]=results[i,1:2]-results.exercise[i,1:2]
    results.exercise[i,7:8]=results[i,5:6]-results.exercise[i,1:2]

        ### FIGURES INSIDE LOOP - SPECIFIC DRAWS ###
if (i >= begin.figures) {

  coef_base=paste("Base: Coef = ",round(results[i,1],digits = 2)+jump,sep = "")
  coef_so=paste("so: Coef = ",round(results[i,5],digits = 2)+jump,sep = "")
  coef_treat=paste(exercise,": Coef = ",round(results.exercise[i,1],digits = 2)+jump,sep = "")


  figure_name=paste(figs.dir,"sample_so",i,"_",exercise,"_",save.ext,".png",sep = "")
  png(figure_name)
  temp.exercise=sample.exercise %>%
    filter(x > -20 & x < 20)
  temp=sample %>%
    filter(x > -20 & x < 20)
```

```
plot(temp$y.so~temp$x, ylim = range(c(temp.exercise$y.so,temp$y.so)),
     xlim = range(c(temp.exercise$x,temp$x)),
     col="red", ylab = "Y", xlab = "X")
par(new = T)
plot(temp.exercise$y.so~temp.exercise$x, ylim = range(c(temp.exercise$y.so,temp$y.so)),
     xlim = range(c(temp.exercise$x,temp$x)),
     col="blue", ylab = "Y", xlab = "X")
abline(v = c(-results[i,3],-results[i,7], -results.exercise[i,3],
             results[i,4], results[i,8], results.exercise[i,4]),
       col=c("blue","red","green", "blue", "red","green"),
       lty=c(1,2,3,1,2,3), lwd=c(1,2,3,1,2,3))
legend("top", legend=c(coef_base, coef_so, coef_treat),
       col=c("blue", "red", "green"), lty=1:3, cex=0.8)
dev.off()

  figure_name=paste(figs.dir,"sample_so",i,"_",exercise,"_",save.ext,"_","model.png",sep = "")
  png(figure_name)
  temp.exercise=sample.exercise %>%
    filter(x > -20 & x < 20)
  temp=sample %>%
    filter(x > -20 & x < 20)
  plot(temp$y.so~temp$x, ylim = range(c(temp.exercise$y.so,temp$y.so)),
       xlim = range(c(temp.exercise$x,temp$x)),
       col="red", ylab = "Y", xlab = "X")
  par(new = T)
  plot(temp.exercise$y.so~temp.exercise$x, ylim = range(c(temp.exercise$y.so,temp$y.so)),
       xlim = range(c(temp.exercise$x,temp$x)),
       col="blue", ylab = "Y", xlab = "X")
  par(new = T)
  plot(temp$y.model~temp$x, ylim = range(c(temp.exercise$y.so,temp$y.so)),
       xlim = range(c(temp.exercise$x,temp$x)),
       col="black", ylab = "Y", xlab = "X")
  abline(v = c(-results[i,2],-results[i,6], -results.exercise[i,2],
               results[i,3], results[i,7], results.exercise[i,3]),
         col=c("blue","red","green", "blue", "red","green"),
         lty=c(1,2,3,1,2,3), lwd=c(1,2,3,1,2,3))
  legend("top", legend=c(coef_base, coef_so, coef_treat),
         col=c("blue", "red", "green"), lty=1:3, cex=0.8)
  dev.off()


  figure_name=paste(figs.dir,"rd_plot_so",i,"_",save.ext,".png",sep = "")
  png(figure_name)
  rdplot(sample$y.so,sample$x)
  dev.off()


  figure_name=paste(figs.dir,"sample",i,"_",save.ext,".png",sep = "")
  png(figure_name)
  sample %>%
    filter(x > -30 & x < 30) %T>%
    plot(y~x,., ylim = range(c(y,y.model)),
         col="blue", ylab = "Y", xlab = "X") %T>%
```

```
      par(new = T) %>%
      plot(y.model~x,., ylim = range(c(y,y.model)),
           axes = FALSE, xlab = "", ylab = "")
    dev.off()


    }
}
```

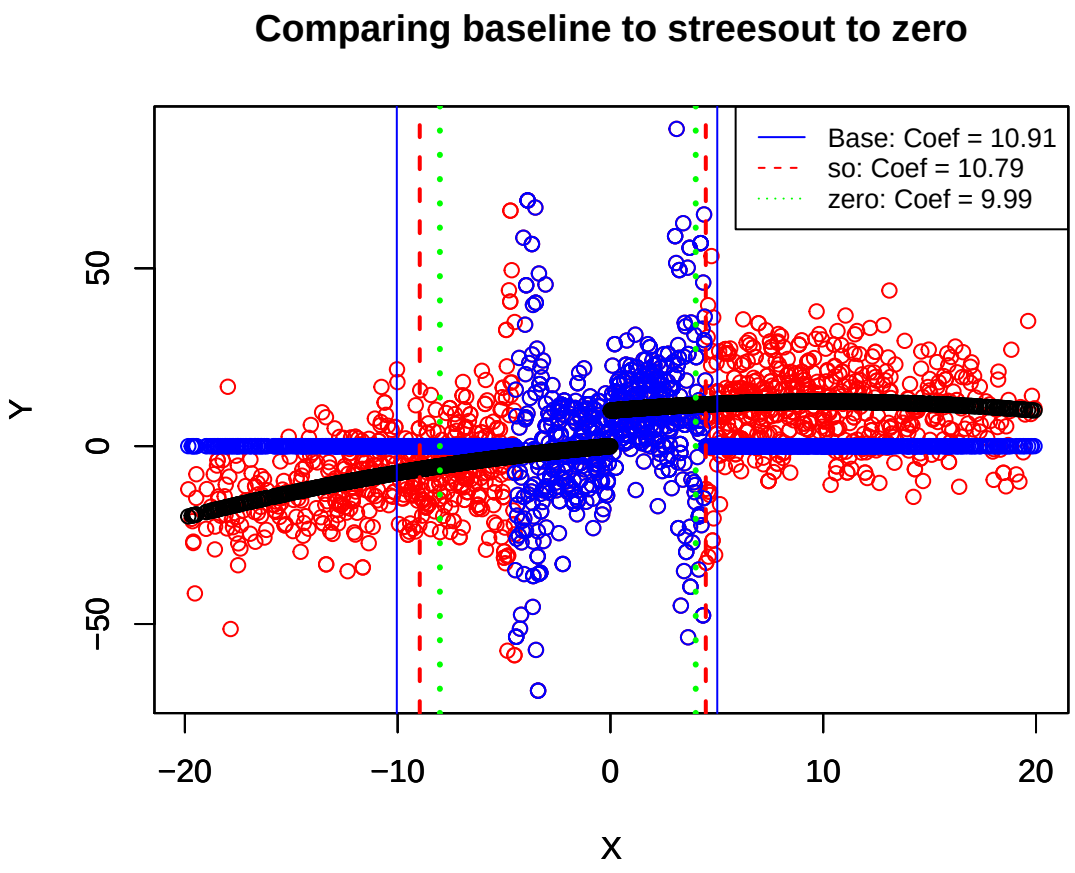
Presenting figures from last iteration for illustration

```
i=loop

  coef_base=paste("Base: Coef = ",round(results[i,1],digits = 2)+jump,sep = "")
  coef_so=paste("so: Coef = ",round(results[i,5],digits = 2)+jump,sep = "")
  coef_treat=paste(exercise,": Coef = ",round(results.exercise[i,1],digits = 2)+jump,sep = "")



    rdplot(sample$y.so,sample$x, title = "RD Plot with stressout")
```

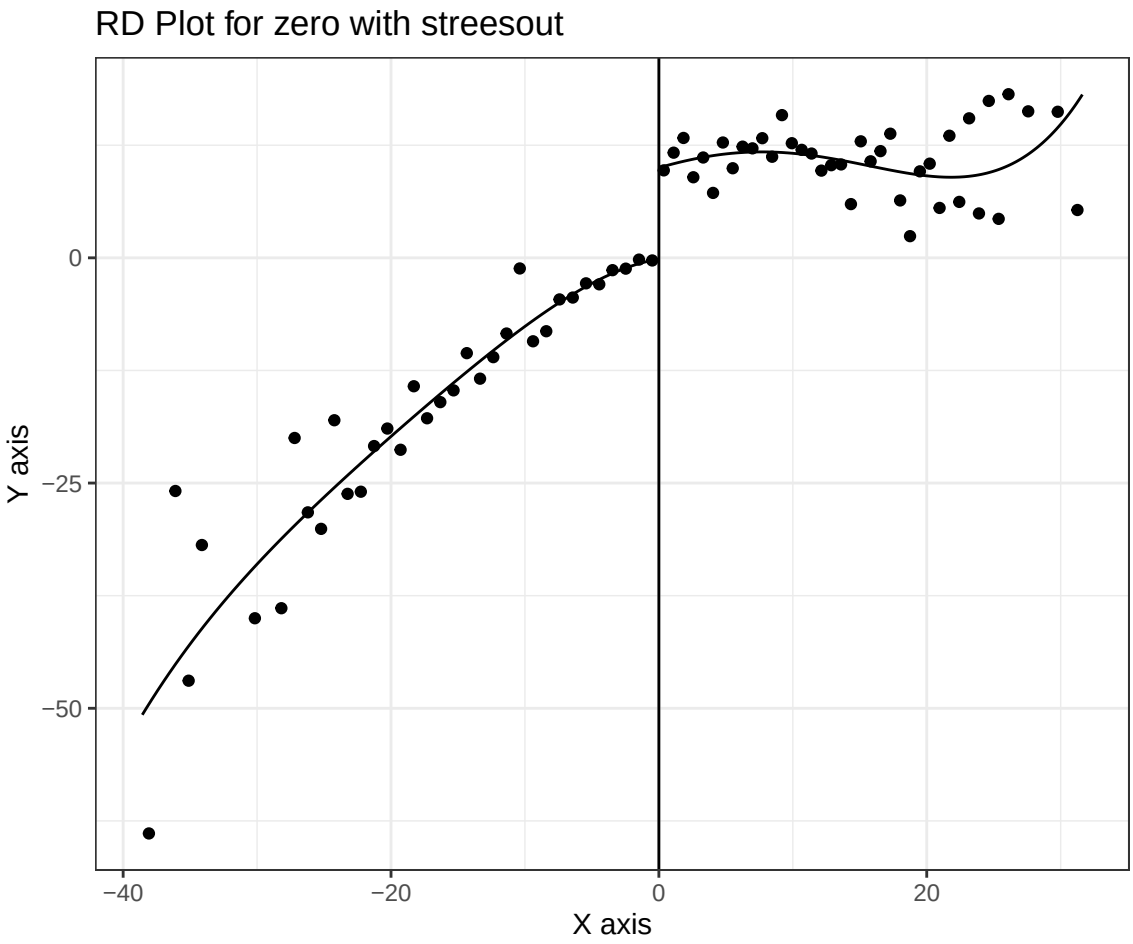



```
    temp.exercise=sample.exercise %>%
      filter(x > -20 & x < 20)
```

```
    temp=sample %>%
      filter(x > -20 & x < 20)
    plot(temp$y.so~temp$x, ylim = range(c(temp.exercise$y.so,temp$y.so)),
         xlim = range(c(temp.exercise$x,temp$x)),
         col="red", ylab = "Y", xlab = "X")
    par(new = T)
    plot(temp.exercise$y.so~temp.exercise$x, ylim = range(c(temp.exercise$y.so,temp$y.so)),
         xlim = range(c(temp.exercise$x,temp$x)),
         col="blue", ylab = "Y", xlab = "X")
    par(new = T)
    plot(temp$y.model~temp$x, ylim = range(c(temp.exercise$y.so,temp$y.so)),
         xlim = range(c(temp.exercise$x,temp$x)),
         col="black", ylab = "Y", xlab = "X")
    abline(v = c(-results[i,2],-results[i,6], -results.exercise[i,2],
                 results[i,3], results[i,7], results.exercise[i,3]),
           col=c("blue","red","green", "blue", "red","green"),
           lty=c(1,2,3,1,2,3), lwd=c(1,2,3,1,2,3))
    legend("topright", legend=c(coef_base, coef_so, coef_treat),
           col=c("blue", "red", "green"), lty=1:3, cex=0.8)
    title(main = paste0("Comparing baseline to streesout to ",exercise))
```



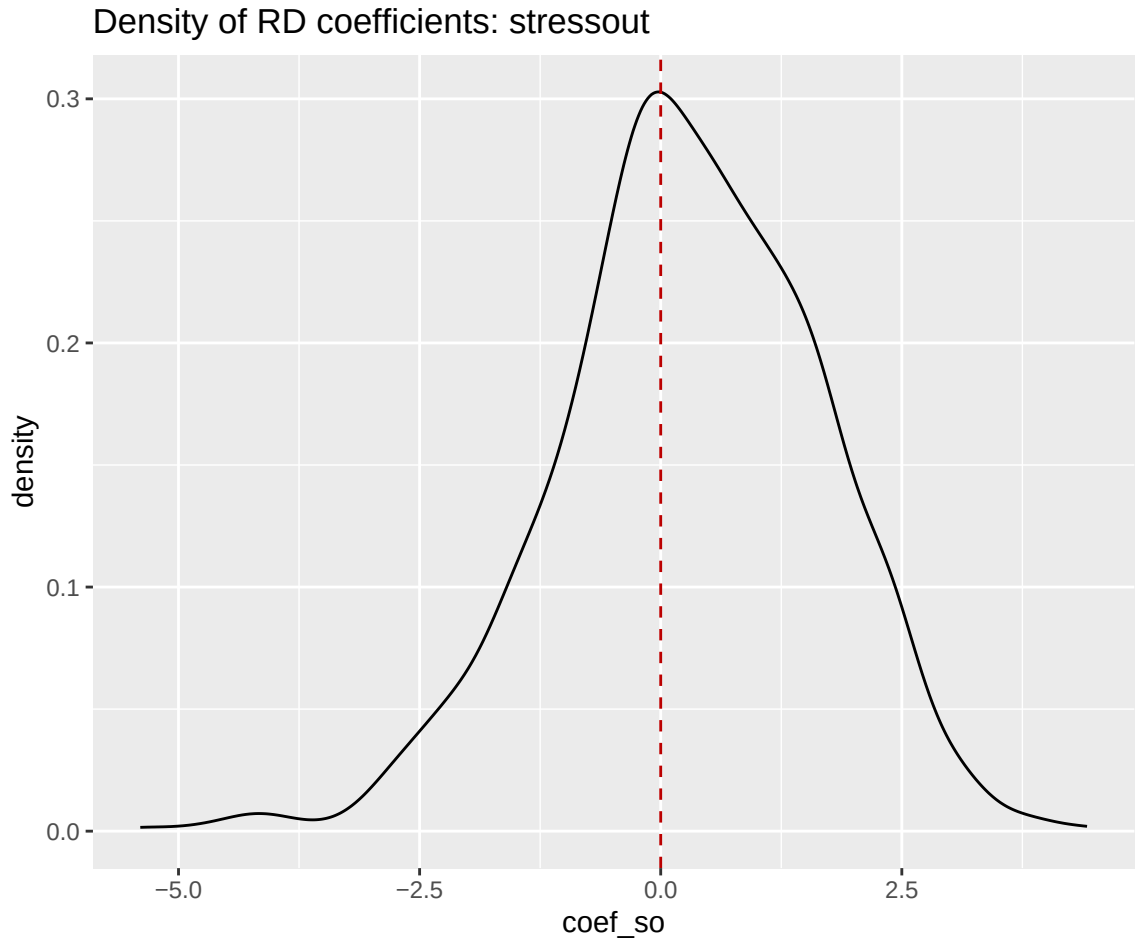
```
rdplot(sample.exercise$y,sample.exercise$x, title = paste("RD Plot for",exercise,"with streesout"))
```



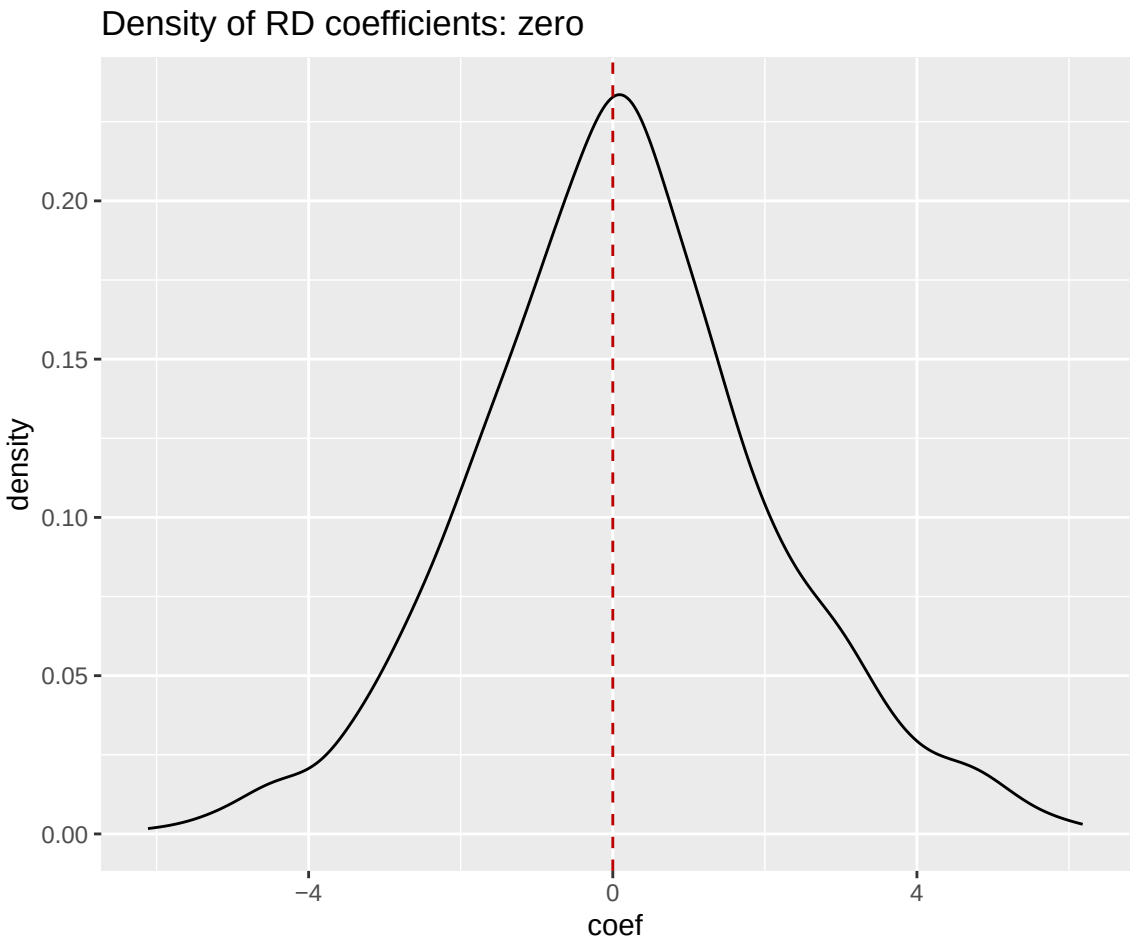
Results

Figures summarizing iterations

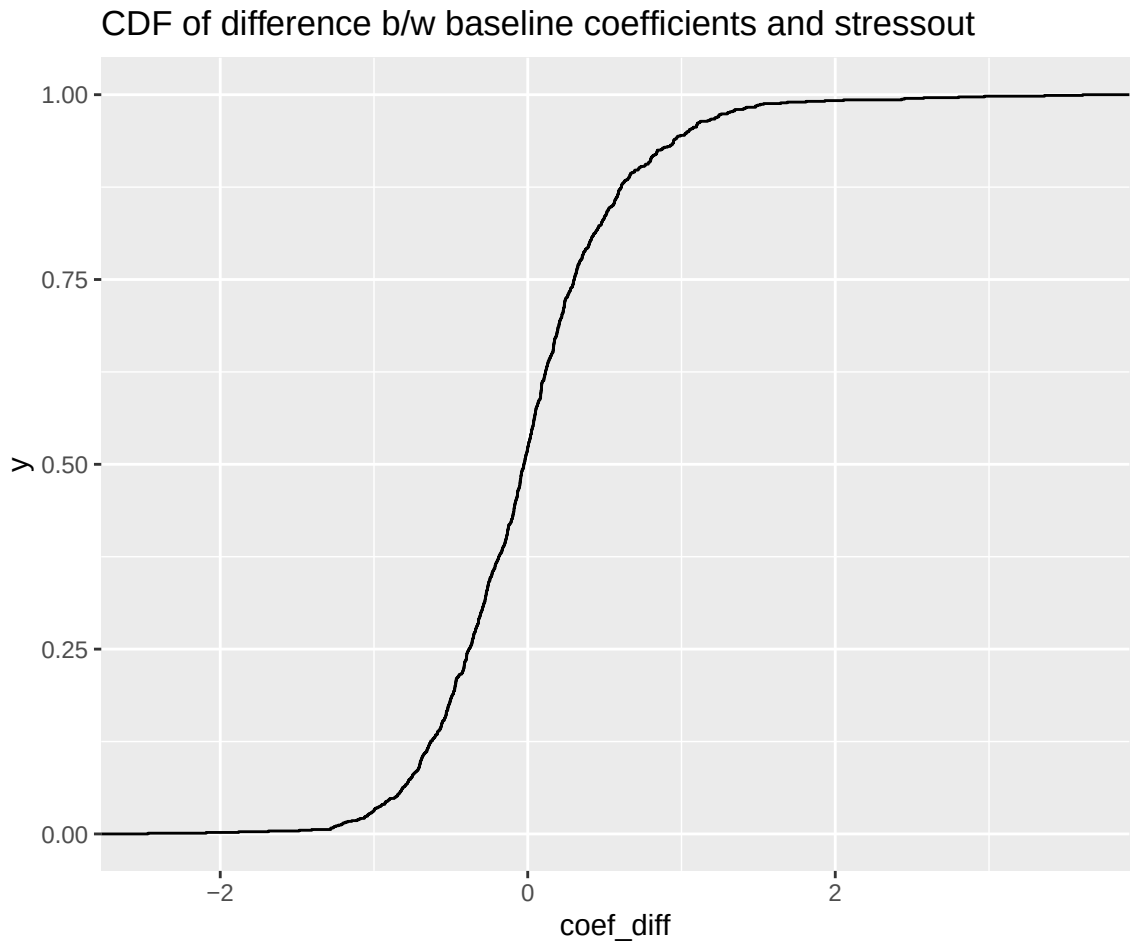
```
## PDF'S OF LEVELS
ggplot(results, aes(coef_so)) + stat_density(geom="line") +
   geom_vline(aes(xintercept=0), colour="#BB0000", linetype="dashed") +
  ggtitle("Density of RD coefficients: stressout")
```



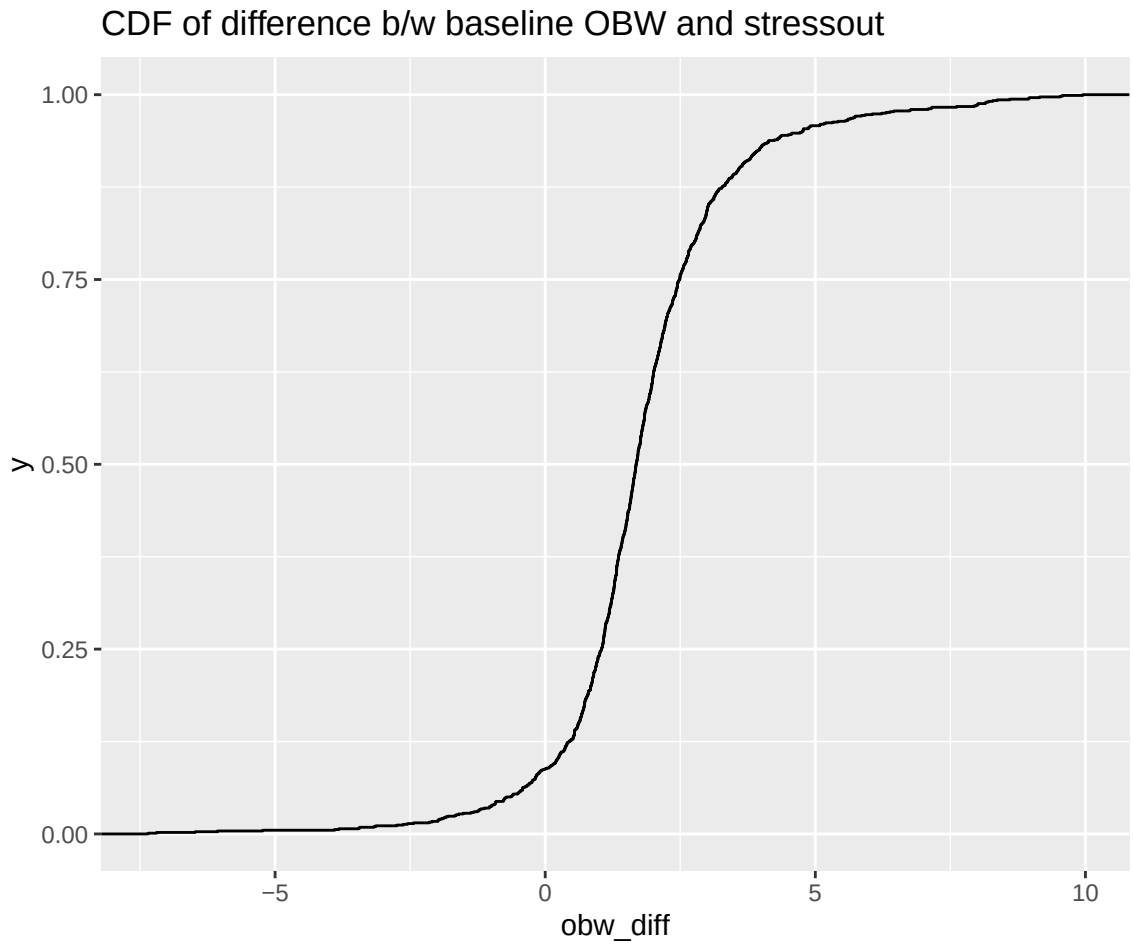

```
ggplot(results.exercise, aes(coef)) + stat_density(geom="line") +
  geom_vline(aes(xintercept=0), colour="#BB0000", linetype="dashed") +
    ggtitle(paste0("Density of RD coefficients: ",exercise))
```



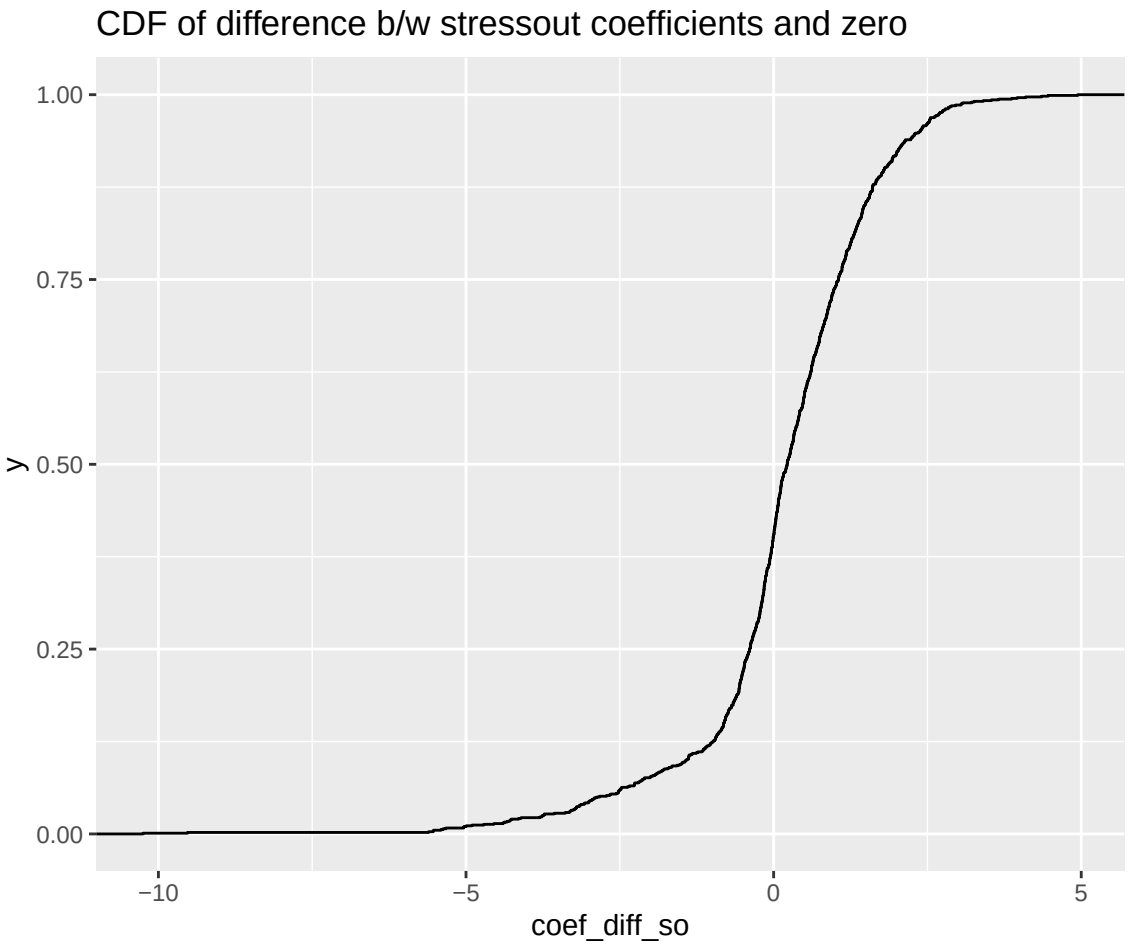

```
 ## so_base_diff_CDF
ggplot(results, aes(coef_diff)) + stat_ecdf(geom = "step") +
 ggtitle("CDF of difference b/w baseline coefficients and stressout")
```



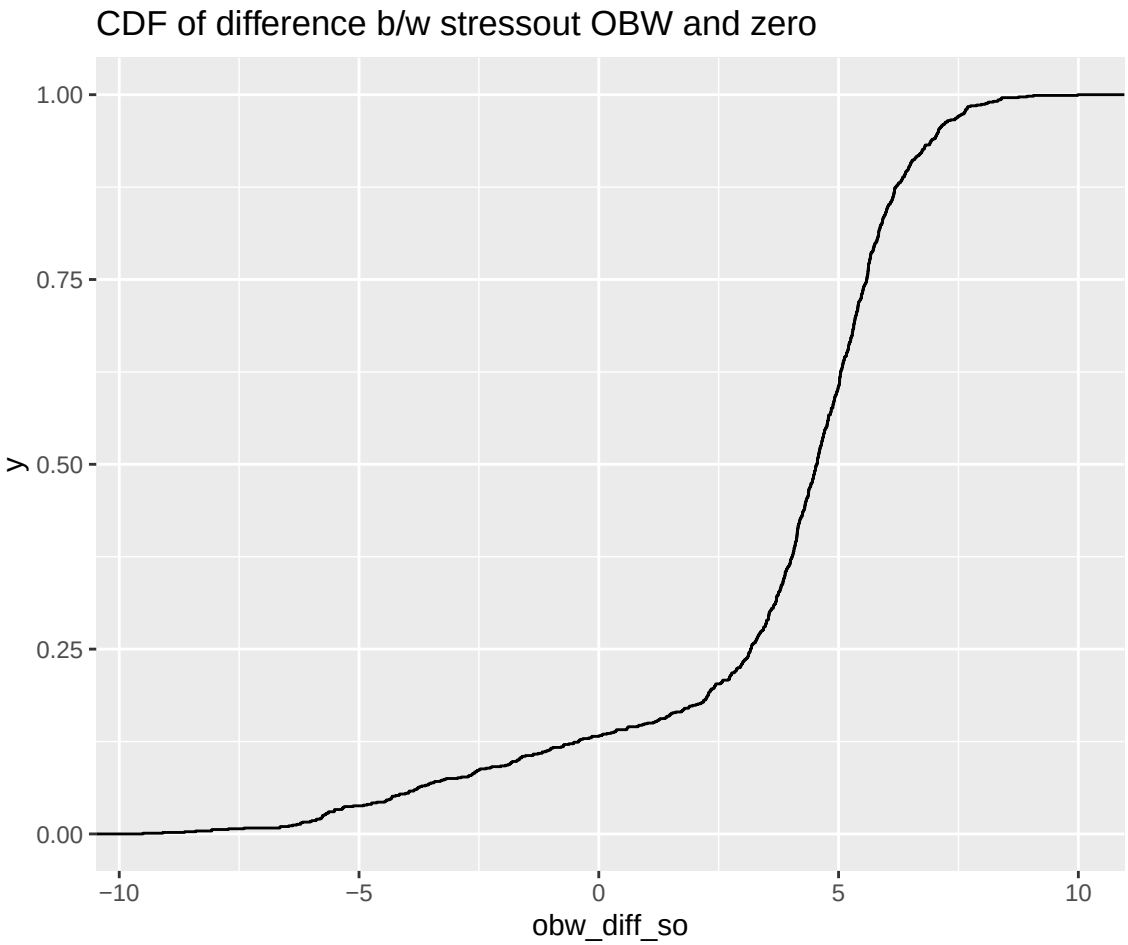

```
ggplot(results, aes(obw_diff)) + stat_ecdf(geom = "step") +
 ggtitle("CDF of difference b/w baseline OBW and stressout")
```




```
### treat_so_diff
  ggplot(results.exercise, aes(coef_diff_so)) + stat_ecdf(geom = "step") +
  ggtitle(paste0("CDF of difference b/w stressout coefficients and ",exercise))
```




```
ggplot(results.exercise, aes(obw_diff_so)) + stat_ecdf(geom = "step") +
ggtitle(paste0("CDF of difference b/w stressout OBW and ",exercise))
```



/newpage ### Summary results - TABLE

```
results_table=as.data.frame(matrix(0, ncol = 5, nrow = 0))
results_table[1,1]=round(mean(results$coef),digits = 4)
results_table[1,2]=round(mean(results$coef_so),digits = 4)
results_table[1,3]=round(mean(results.exercise$coef),digits = 4)
results_table[1,4]=round(mean(results.exercise$coef_diff),digits = 4)
results_table[1,5]=round(mean(results.exercise$coef_diff_so),digits = 4)
results_table[2,1]=round(mean(results$obw),digits = 4)
results_table[2,2]=round(mean(results$obw_so),digits = 4)
results_table[2,3]=round(mean(results.exercise$obw),digits = 4)
results_table[2,4]=round(mean(results.exercise$obw_diff),digits = 4)
results_table[2,5]=round(mean(results.exercise$obw_diff_so),digits = 4)
colnames(results_table) = c("base","so",exercise,"diff_base","diff_so")
rownames(results_table) = c("coef","obw")

kable(results_table, caption = "Summary Table")
```

Table 1: Summary Table

	base	so	zero	diff_base	diff_so
coef	0.2874	0.2853	0.1149	0.1724	0.1704
obw	13.1162	5.6522	7.6646	5.4516	3.6397