# causal_alcohol_dysphagia

August 10, 2018

```
In [1]: library(IRdisplay)
        display_html('<script>
        code_show=true;
        function code_toggle() {
          if (code_show){
            $(\'div.input\').hide();
          } else {
            $(\'div.input\').show();
          }
          code_show = !code_show
        }
        $( document ).ready(code_toggle);
        </script>
          <form action="javascript:code_toggle()">
            <input type="submit" value="Click here to toggle on/off the raw code.">
         </form>'
        )
```

### 0.1 This script computes the average causal effect of alcohol use on dysphagia (*ssq score*) using two methods.

Date: July 28, 2017
Author: Wei

```
In [2]: library(boot)
        library(randomForest)
        library(foreach)
        library(doParallel)

        rm(list=ls())
```

```
randomForest 4.6-12
Type rfNews() to see new features/changes/bug fixes.
Warning message:
"package 'foreach' was built under R version 3.4.2"Loading required package: iterators
Loading required package: parallel
```

```
In [3]: # Input:
        # df: a data frame to be resampled
        # k: number of resampled datasets to generate.
        # f: a function of df giving the bootstrap statistic
        # (e.g. function(df) { mean(df$x1) })
        # q: a real number 0 < q
        # Output: a three element vector giving the statistic of interest (first element),
        # and lower and upper confidence intervals corresponding to
        # q and 1-q quantiles (second and third elements) of the empirical
        # bootstrap distribution.

        bootstrap_ci <- function(df, k, f, q){
            df_resample <- function(df){
          ind <- sample(1:nrow(df),size = nrow(df), replace = TRUE)
          return(df[ind,])
            }
          cl <- makeCluster(7)
          registerDoParallel(cl)
        #   means <- vector(mode = "numeric", length = k)
          original_mean <- f(df)

          means <- foreach (i=1:k, .packages="randomForest") %dopar% {
            data <- df_resample(df)
            f(data)
          }
          stopCluster(cl)

          means <- unlist(means)
          diff_statistics <- quantile(means-original_mean, probs = c(q,1-q))
          diff_statistics <- unname(diff_statistics)

          return(c(original_mean, original_mean + diff_statistics[1], original_mean + diff_sta
        }
```

### 0.1.1 Get dataset

```
In [5]: df<- read.csv("post.csv", TRUE, ",")
        df$mdadi_total <-NULL
        head(df, n=2L)
```

| X | tstage | nstage | mstage | weight | pain | dysphagia | dysgeusia | kps | smoking | ... | hpv_ever_po |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 3 | 2 | 0 | 91.7 | 2 | 3 | 2 | 100 | 2 | ... | 1 |
| 2 | 1 | 2 | 0 | 208.2 | 7 | 2 | 2 | 100 | 2 | ... | 1 |

Number of patients:

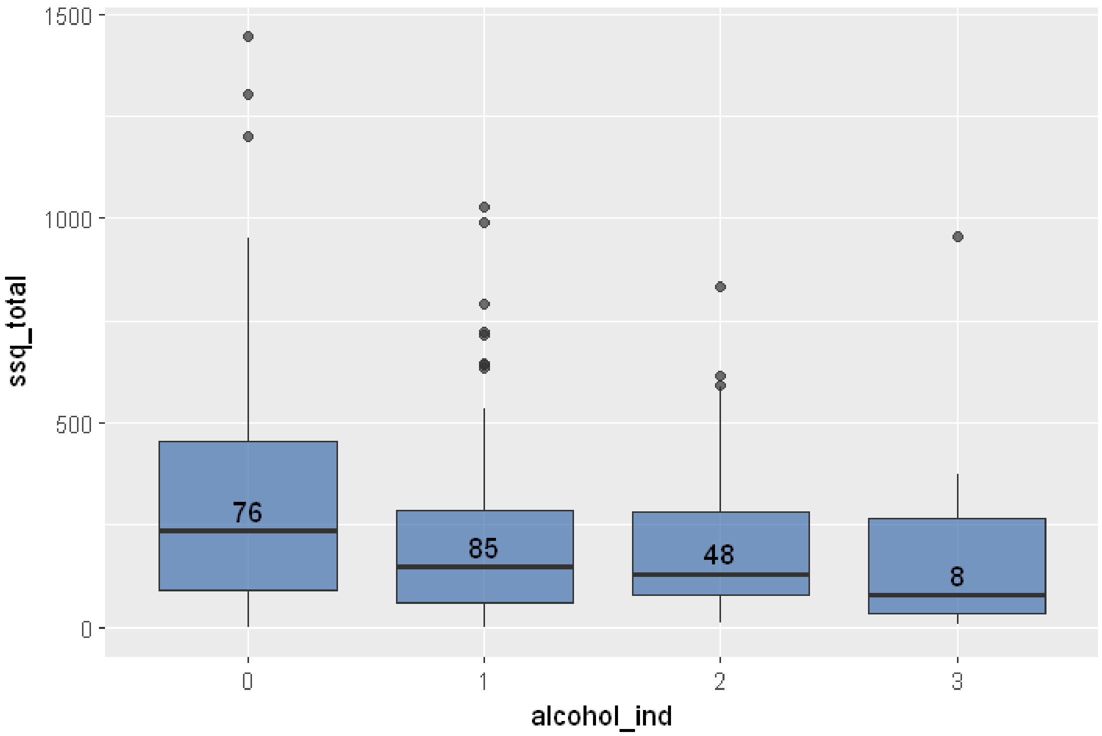
```
In [2]: nrow(df)
```

217

```
In [4]: library(repr)
        library(ggplot2)
        options(repr.plot.width=6, repr.plot.height=4)

        df$alcohol_ind <- as.factor(df$alcohol_ind)
        # function for number of observations
        give.n <- function(x){
          return(c(y = median(x)+50, label = length(x)))
        }
        p10 <- ggplot(df, aes(x = alcohol_ind, y = ssq_total)) +
                geom_boxplot(fill = "#4271AE", alpha = 0.7)+
                stat_summary(fun.data = give.n, geom = "text", fun.y = median)
        p10

        # p10<-ggplot(df, aes( x = ssq_total))+geom_hist()
```



### 0.1.2 Method 1: Adjust for all other baseline covariates. (Parametric g-formula)

Average causal effect:

$$\frac{1}{n}\sum_i (E[Y \mid A = 1, x_i; \hat{\alpha}] - E[Y \mid A = 0, x_i; \hat{\alpha}])$$

$Y$: outcome, dysphagia measured as ssq score, in our case
$A$: treatment, alcohol, in our case

$x_i$: all the other covariates or potential confounders
$\hat{\alpha}$: parameter for model $E[Y \mid A, X]$

```
In [6]: ## Compute average causal effect
        ACE_g <- function(post){
            bag.post = randomForest(ssq_total~., data = post, ntree = 500)

            post0 <- post
            post2 <- post
            post0$alcohol_ind <-0
            post2$alcohol_ind <-1

            yhat.bag0 = predict(bag.post, newdata= post0)

            yhat.bag2 = predict(bag.post, newdata= post2)

            delta <- mean(yhat.bag2 - yhat.bag0)
            return(delta)
        }

        main1 = function(){

          df<- read.csv("post.csv", TRUE, ",")
          df$mdadi_total <- NULL
          set.seed(1)
          k <- 500
          ci <- bootstrap_ci(df, k, ACE_g, q = 0.025)
          print("Average causal effect of alcohol (0 vs 1) on ssq score:")
          print(ci[1])
          print("Confidence interval (95%):")
          print(ci[2:3])

        }

        main1()
```

```
[1] "Average causal effect of alcohol (0 vs 1) on ssq score:"
[1] -26.83457
[1] "Confidence interval (95%):"
[1] -54.824986  -3.660523
```

### 0.1.3 Method 2: inverse probability weighting

1. Estimate propensity score **P**(**alcohol** | **all other covariates**) using multi-class random forest

   Average causal effect (from Ilya):

$$\frac{1}{N_1}\sum_{i}^{n}\frac{I(A_i = a_1)}{P(A_i = a_1 \mid x_i)}y_i - \frac{1}{N_0}\sum_{i}^{n}\frac{I(A_i = a_0)}{P(A_i = a_0 \mid x_i)}y_i$$

where:
$A$: treatment, in our case, alcohol use
$y$: dysphagia, ssq score
$x$: all the other covariates, or potential confounders

$$N_1 = \sum_i^n \frac{I(A_i = a_1)}{P(A_i = a_1 \mid x_i)}$$

$$N_0 = \sum_i^n \frac{I(A_i = a_0)}{P(A_i = a_0 \mid x_i)}$$

```
In [4]: ACE_IPW <- function(df, verbose=FALSE){
            df$alcohol_ind <- as.factor(df$alcohol_ind)

            data_train <- df[,names(df) != "ssq_total"]
            m <- randomForest(alcohol_ind~., data = data_train, ntree=500)

        #     m <- polr(alcohol_ind~., data=data_train, Hess=TRUE)
        #     summary(m)
        #     ctable <- coef(summary(m))
        #     p <- pnorm(abs(ctable[, "t value"]), lower.tail = FALSE) * 2
        #     ctable <- cbind(ctable, "p value" = p)
            # confidence intervals
        #     ci <- confint(m)
            probs <- predict(m, data_train, type="p")
            if (verbose){
                print(probs)
            }
            weight <- 1/probs
            weight0 <- weight[df$alcohol_ind==0, c("0")]
            weight2 <- weight[df$alcohol_ind==1, c("1")]

            y0 <- df[df$alcohol_ind=="0",c("ssq_total")]
            y2 <- df[df$alcohol_ind=="1",c("ssq_total")]

            return(sum(y2*weight2)/sum(weight2) - sum(y0*weight0)/sum(weight0))
        }
```

```
In [5]: main2 = function(){

          df<- read.csv("post.csv", TRUE, ",")
          df$mdadi_total <- NULL

          set.seed(1)
          k <- 500
          ci <- bootstrap_ci(df, k, ACE_IPW, q = 0.025)
          print("Average causal effect of alcohol (0 vs 1) on ssq score:")
          print(ci[1])
          print("Confidence interval (95%):")
```