

Web Material

The implications of using lagged and baseline exposure terms in the longitudinal-causal and regression models

Mohammad Ali Mansournia, Ashley I Naimi, Sander Greenland

Web Appendix 1: software codes

i) Stata codes for marginal structural Cox model

```
logistic cursmok agebase hearta gout highbp diabet fib75 chol75 hbpsyst hbpdias
obese thin B* L* phase3 phase4
predict p if e(sample)
replace p=p* cursmok +(1-p)*(1- cursmok)
sort id visit
by id: replace p=p*p[_n-1] if _n!=2
rename p denom
```

```
logistic cursmok agebase B* Lcursmok phase3 phase4
predict p if e(sample)
replace p=p* cursmok +(1-p)*(1- cursmok)
sort id visit
by id: replace p=p*p[_n-1] if _n!=2
rename p num
```

```
gen sweight=num/denom
```

Model 1

```
logistic mi cursmok agebase Lcursmok B* phase3 phase4 [pw=sweight], cluster(id)
lincom cursmok+Lcursmok, or
```

Model 2

```
logistic mi cursmok agebase Lcursmok Bfib75 Bhearta Bgout Bhighbp Bchol75
Bdiabet Bhbpsyst Bhbpdias Bobese Bthin phase3 phase4 [pw=sweight], cluster(id)
lincom cursmok+Lcursmok, or
```

Model 3

```
logistic mi cursmok agebase B* phase3 phase4 [pw=sweight], cluster(id)
```

Model 4

```
logistic mi cursmok agebase Bfib75 Bhearta Bgout Bhighbp Bchol75 Bdiabet
Bhbpsyst Bhbpdias Bobese Bthin phase3 phase4 [pw=sweight], cluster(id)
```

ii) R codes for structural accelerated failure time model

```
packages <- c("tidyverse", "GGally", "data.table", "broom",
             "ggplot2", "tabplot", "gridExtra", "here", "parallel",
             "VIM", "dummies", "foreign", "cmprsk", "lubridate")
```

```
for (package in packages) {
```

```

if (!require(package, character.only=T, quietly=T)) {
install.packages(package,repos='http://lib.stat.cmu.edu/R/CRAN')
}
}

for (package in packages) {
  library(package, character.only=T)
}

thm<- theme_classic() +
  theme(
legend.position = "top",
  #legend.title=element_blank(),
legend.background = element_rect(fill = "transparent", colour = NA),
legend.key = element_rect(fill = "transparent", colour = NA)
)
theme_set(thm)

set.seed(123)
# define grid
psitilde1 <- seq(-1,1,by=.025)
psitilde2 <- seq(0,1,by=.025)
psitilde<- cbind(expand.grid(psitilde1,psitilde2))

# define saftm
saftm2 <- function(id,psitilde1,psitilde2,y,ti,a,max_ti){
H_psi<- C_psi<- delta <- 0
ti<- data.frame(ti)
a <- data.frame(a)
for(m in 1:ncol(ti)){
  if(m==1){
H_psi<- H_psi + ti[,m]*exp(psitilde1*a[,m] + psitilde2*0)*y
  } else{
H_psi<- H_psi + ti[,m]*exp(psitilde1*a[,m] + psitilde2*a[,m-1])*y
  }
}
if(psitilde1>0&psitilde2>0){
C_psi<- max_ti*exp(psitilde1*0+psitilde2*0)*y
} else if(psitilde1 <= 0 & psitilde2 <= 0){
C_psi<- max_ti*exp(psitilde1*1 + psitilde2*1)*y
} else if(psitilde1 > 0 & psitilde2 <= 0){
C_psi<- max_ti*exp(psitilde1*0 + psitilde2*1)*y
} else if(psitilde1 <= 0 & psitilde2 > 0){
C_psi<- max_ti*exp(psitilde1*1 + psitilde2*0)*y
}
delta <- as.numeric(H_psi<= C_psi)*y
res <- data.frame(id,psitilde1,psitilde2,y,H_psi,C_psi,delta)
return(res)
}

```

```

# impute potential outcomes under different psitilde values
snm_res<- mclapply(1:nrow(psitilde),function(x)
snaftm2(snm$d$id,psitilde[x,1],psitilde[x,2],

snmd$mi,t_dat[,2:4],a_dat[,2:4],max(rowSums(t_dat[,2:4]))),mc.cores=detectCores()-
2)
snm_res<- do.call(rbind,snm_res)

# merge imputed POs with original data
g_est<- left_join(a,snm_res)

# fit pooled logistic exposure model for each combination of psitilde values
r <- mclapply(1:nrow(psitilde),function(x) c(summary(glm(cursmok ~
delta+as.factor(visit)+agebase
                                +fib75+hearta+highbp+diabet+chol75
                                +hbpsyst+hbpsdias+gout+obese+thin
                                +Lfib75+Lgout+Lhearta+Lhighbp+Ldiabet
                                +Lchol75+Lhbpsyst
                                +Lhbpdias+Lobese+Lthin,
                                data=g_est, family=binomial,

subset=visit>1&psitilde1==psitilde[x,1]&psitilde2==psitilde[x,2]))$coefficients[2,],p
sitilde[x,1],psitilde[x,2]),mc.cores=detectCores()-2)

r <- do.call(rbind,r)
r <- data.frame(r)

names(r) <- c("Estimate","SE","Z","pZ","psitilde1","psitilde2")

# find where z-statistic crosses zero, and upper and lower confidence bounds
min_z<- min(abs(r$Z))
z_ucl<- min(abs(r$Z-1.96))
z_lcl<- min(abs(r$Z+1.96))

res0 <- subset(r,abs(Z)==min_z)
res_ucl<- subset(r,abs(Z-1.96)==z_ucl)
res_lcl<- subset(r,abs(Z+1.96)==z_lcl)

res0 <- colMeans(res0)
res_ucl<- colMeans(res_ucl)
res_lcl<- colMeans(res_lcl)

time_ratio2 <- exp(-sum(res0[5:6]))
time_ucl2 <- exp(-sum(res_ucl[5:6]))
time_lcl2 <- exp(-sum(res_lcl[5:6]))

# find shape parameter for converting time ratios to hazard ratios
model<- exp((weibull_mod$scale)*(-log(time_ratio2)))

```

```
model_LCL<- exp((weibull_mod$scale)*(-log(time_lcl2)))
model_UCL<- exp((weibull_mod$scale)*(-log(time_ucl2)))
```

```
# estimated HRs with 95% CIs
```

```
model
model_LCL
model_UCL
```

```
iii) Stata codes for time-dependent Cox model
```

```
stcox cursmok agebase hearta gout highbp diabet fib75 chol75 hbpsyst hbpdias obese
thin L* B*
lincom cursmok+Lcursmok, hr
```

```
stcox cursmok agebase hearta gout highbp diabet fib75 chol75 hbpsyst hbpdias obese
thin L* Bhearta Bgout Bhighbp Bdiabet Bfib75 Bchol75 Bhbpsyst Bhbpdias Bobese
Bthin
lincom cursmok+Lcursmok, hr
```

```
stcox cursmok agebase hearta gout highbp diabet fib75 chol75 hbpsyst hbpdias obese
thin Lhearta Lgout Lhighbp Ldiabet Lfib75 Lchol75 Lhbpsyst Lhbpdias Lobese
Lthin B*
```

```
stcox cursmok agebase hearta gout highbp diabet fib75 chol75 hbpsyst hbpdias obese
thin Lhearta Lgout Lhighbp Ldiabet Lfib75 Lchol75 Lhbpsyst Lhbpdias Lobese
Lthin Bhearta Bgout Bhighbp Bdiabet Bfib75 Bchol75 Bhbpsyst Bhbpdias Bobese
Bthin
```

```
iv) R codes for Figure 2
```

```
packages <- c("tidyverse", "GGally", "data.table", "broom", "survival",
             "ggplot2", "tabplot", "gridExtra", "here", "parallel",
             "VIM", "dummies", "foreign", "cmprsk", "lubridate")
```

```
for (package in packages) {
  if (!require(package, character.only=T, quietly=T)) {
    install.packages(package, repos='http://lib.stat.cmu.edu/R/CRAN')
  }
}
```

```
for (package in packages) {
  library(package, character.only=T)
}
```

```
thm <- theme_classic() +
  theme(
    legend.position = "top",
    #legend.title=element_blank(),
    legend.background = element_rect(fill = "transparent", colour = NA),
    legend.key = element_rect(fill = "transparent", colour = NA)
  )
theme_set(thm)
```

```

aa <- subset(a,visit>1)

reg1 <- coxph(Surv(time=t0,time2=t, event=d) ~
  cursmok+agebase+as.numeric(hearta)-1+
  as.numeric(gout)-1+as.numeric(highbp)-1+as.numeric(diabet)-1+
  fib75+chol75+hbpsyst+hbpdias+obese+thin+phase3+phase4,data=aa)

KM_0 <- survfit(reg1,newdata=data.frame(cursmok=0,
  agebase=mean(aa$agebase),
  hearta=.05,
  gout=.09,
  highbp=.33,
  diabet=.05,
  fib75=.28,
  chol75=.25,
  hbpsyst=.59,
  hbpdias=.26,
  obese=.15,
  thin=.03,
  phase3=1,
  phase4=0),type="kaplan-meier",conf.type="log")
KM_1 <- survfit(reg1,newdata=data.frame(cursmok=1,
  agebase=mean(aa$agebase),
  hearta=.05,
  gout=.09,
  highbp=.33,
  diabet=.05,
  fib75=.28,
  chol75=.25,
  hbpsyst=.59,
  hbpdias=.26,
  obese=.15,
  thin=.03,
  phase3=1,
  phase4=0),type="kaplan-meier",conf.type="log")

table(as.numeric(aa$hearta))
plotDat_0 <- data.frame(cdf = 1 - KM_0$surv, cursmok="No", time =
  KM_0$time,model="adjusted")
plotDat_1 <- data.frame(cdf = 1 - KM_1$surv, cursmok="Yes", time =
  KM_1$time,model="adjusted")
plotDat <- rbind(plotDat_0,plotDat_1)
table(plotDat$cursmok)
plot_a <- ggplot(plotDat) +
  geom_line(aes(x=time,y=cdf,group=cursmok,linetype=cursmok)) + ylim(0,.3) +
  xlab("Follow Up (Years)") + ylab("Cumulative Proportion") +
  labs(linetype="Smoker") + ggtitle("Cox Regression*")
plot_a

```

```
aa$num <- glm(cursmok ~ agebase+phase3+phase4,data=a, family=binomial,
subset=visit>1)$fitted.values
aa$den <- glm(cursmok ~
agebase+hearta+gout+highbp+diabet+fib75+chol75+hbpsyst+hbpdias+obese+thin+p
hase3+phase4,
data=aa, family=binomial, subset=visit>1)$fitted.values
```

```
aa <- aa %>% group_by(id) %>% mutate(sw = cumprod( (num/den)*cursmok + (1-
num)/(1-den)*(1-cursmok) ))
```

```
#aa <- aa %>% ungroup(id) %>% mutate(sw =
ifelse(sw>quantile(sw,.99),quantile(sw,.99),sw))
```

```
aggregate(aa$sw,list(aa$visit),summary)
```

```
coxph(Surv(time=t0,time2=t, event=d) ~ cursmok, weight=sw, ties="efron", data=aa)
```

```
KM1 <- survfit(Surv(time=t0,time2=t, event=d) ~ cursmok, weight=sw,
type="kaplan-meier", conf.type="log", data=aa)
```

```
cursmok <- c(rep(0,KM1$strata[[1]]),rep(1,KM1$strata[[2]]))
plotDat <- data.frame(cdf = 1 - KM1$surv, cursmok =
ifelse(cursmok==0,"No","Yes"), time = KM1$time,model="weighted")
plot_b <- ggplot(plotDat) +
geom_line(aes(x=time,y=cdf,group=cursmok,linetype=cursmok)) + ylim(0,.3) +
xlab("Follow Up (Years)") + ylab("Cumulative Proportion") +
labs(linetype="Smoker") + ggtitle("MSM Regression")
plot_b
```

```
## SAFTM CURVES
```

```
snaftm2_curve0 <- function(id,y,ti,a,max_ti){
  H_psi <- C_psi <- delta <- 0
  ti <- data.frame(ti)
  a <- data.frame(a)
  for(m in 1:ncol(ti)){
    if(m==1){
      H_psi <- H_psi + ti[,m]*exp(.2*(a[,m]==1) + .725*0)*y
    } else{
      H_psi <- H_psi + ti[,m]*exp(.2*(a[,m]==1) + .725*(a[,m-1]==1))*y
    }
  }
  C_psi <- max_ti*exp(.2*0+.725*0)*y
  delta <- as.numeric(H_psi <= C_psi & H_psi < 13)*y
  res <- data.frame(id,y,H_psi,C_psi,delta)
  return(res)
}
```

```
snaftm2_curve1 <- function(id,y,ti,a,max_ti){
  H_psi <- C_psi <- delta <- 0
  ti <- data.frame(ti)
```

```

a <- data.frame(a)
for(m in 1:ncol(ti)){
  if(m==1){
    H_psi <- H_psi + ti[,m]*exp(-.2*(a[,m]==0) + .725*0)*y
  } else{
    H_psi <- H_psi + ti[,m]*exp(-.2*(a[,m]==0) - .725*(a[,m-1]==0))*y
  }
}
C_psi <- max_ti*exp(.2*0+.725*0)*y
delta <- as.numeric(H_psi <= C_psi)*y
res <- data.frame(id,y,H_psi,C_psi,delta)
return(res)
}

res_0 <-
snaftm2_curve0(snmd$id,snmd$mi,t_dat[,2:4],a_dat[,2:4],max(rowSums(t_dat[,2:4]))
)
a0 <- merge(snmd,res_0)
a0 <- a0 %>% mutate(t4=ifelse(is.na(t4),0,t4),t =
ifelse(delta==0,t2+t3+t4,H_psi),cursmok=0)

res_1 <-
snaftm2_curve1(snmd$id,snmd$mi,t_dat[,2:4],a_dat[,2:4],max(rowSums(t_dat[,2:4]))
)
a1 <- merge(snmd,res_1)
a1 <- a1 %>% mutate(t4=ifelse(is.na(t4),0,t4),t =
ifelse(delta==0,t2+t3+t4,H_psi),cursmok=1)

aa <- rbind(a0,a1)
head(aa)
nrow(aa)
max(aa$t,na.rm=T)

KM1 <- survfit(Surv(t, delta) ~ cursmok, type="kaplan-meier", conf.type="log",
data=aa)
plot(KM1)

cursmok <- c(rep(0,KM1$strata[1]),rep(1,KM1$strata[2]))
plotDat <- data.frame(cdf = 1 - KM1$surv, cursmok =
ifelse(cursmok==0,"No","Yes"), time = KM1$time,model="SAFTM")
plot_c <- ggplot(plotDat) +
geom_line(aes(x=time,y=cdf,group=cursmok,linetype=cursmok)) + ylim(0,.3) +
xlab("Follow Up (Years)") + ylab("Cumulative Proportion") +
labs(linetype="Smoker") + ggtitle("SAFTM Regression")
plot_c

pdf("./cdf.pdf",width=5,height=10)
grid.arrange(plot_a,plot_b,plot_c,ncol=1)
dev.off()

```