Data analysis

Hege Michiels

4/27/2023

```
##   id gender      age Z     time status    shift          ARM
## 1  1      1 55.20165 1 47.88922      1  0.00000      Control
## 2  2      0 55.21952 1 60.00000      0  0.00000      Control
## 3  3      1 58.12572 0 49.93390      1  0.00000 Experimental
## 4  4      1 53.51899 1 56.75231      1 34.24845      Control
## 5  5      0 52.67932 0 50.13366      1  0.00000 Experimental
## 6  6      1 55.69778 1 47.75965      1  0.00000      Control
```

The following variables are needed in the dataset:

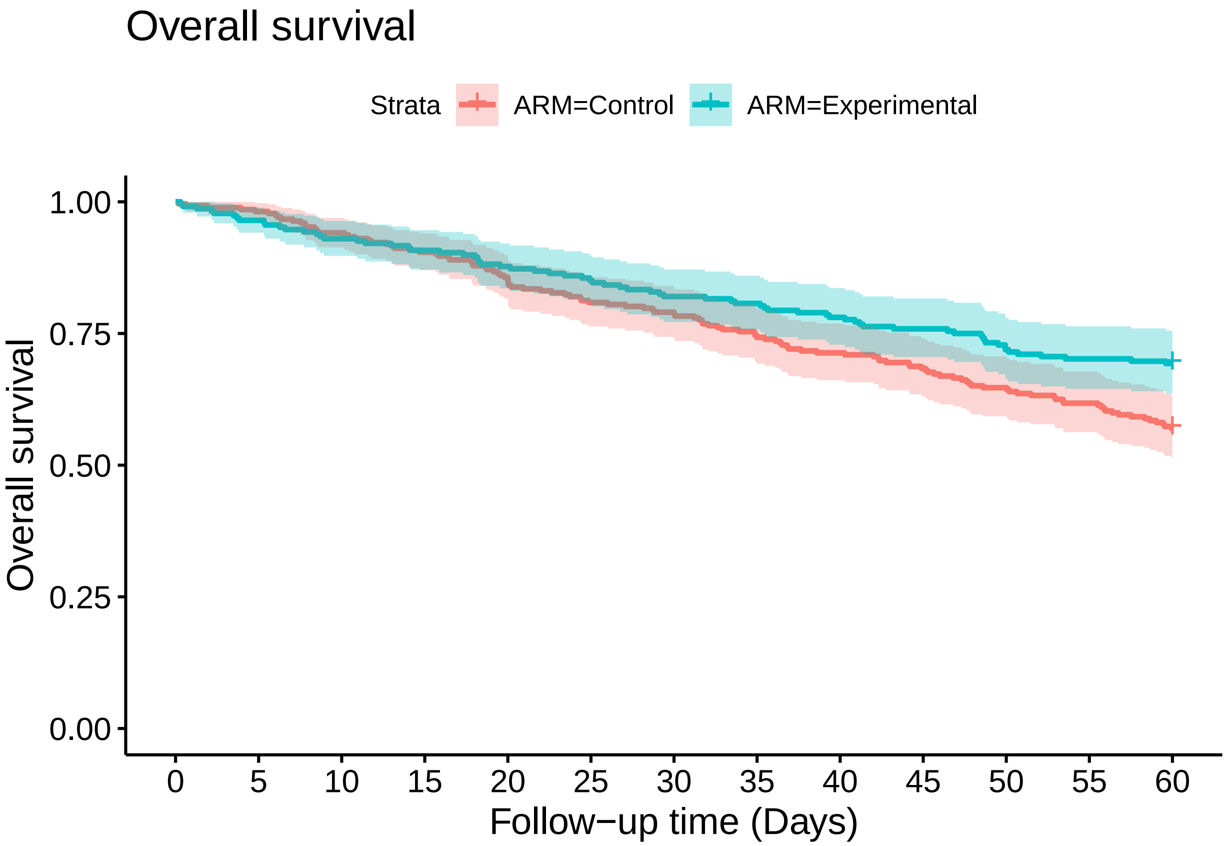
Z: indicating the treatment arm (with Z = 1: control and Z = 0: experimental)

time: right-censored survival time

status: indication whether (status = 1) or not (status = 0) the patient had an event

shift: possible time of treatment crossover from one treatment to the other, 0 if the patient did not cross over

Data



Adding an indicator for crossover

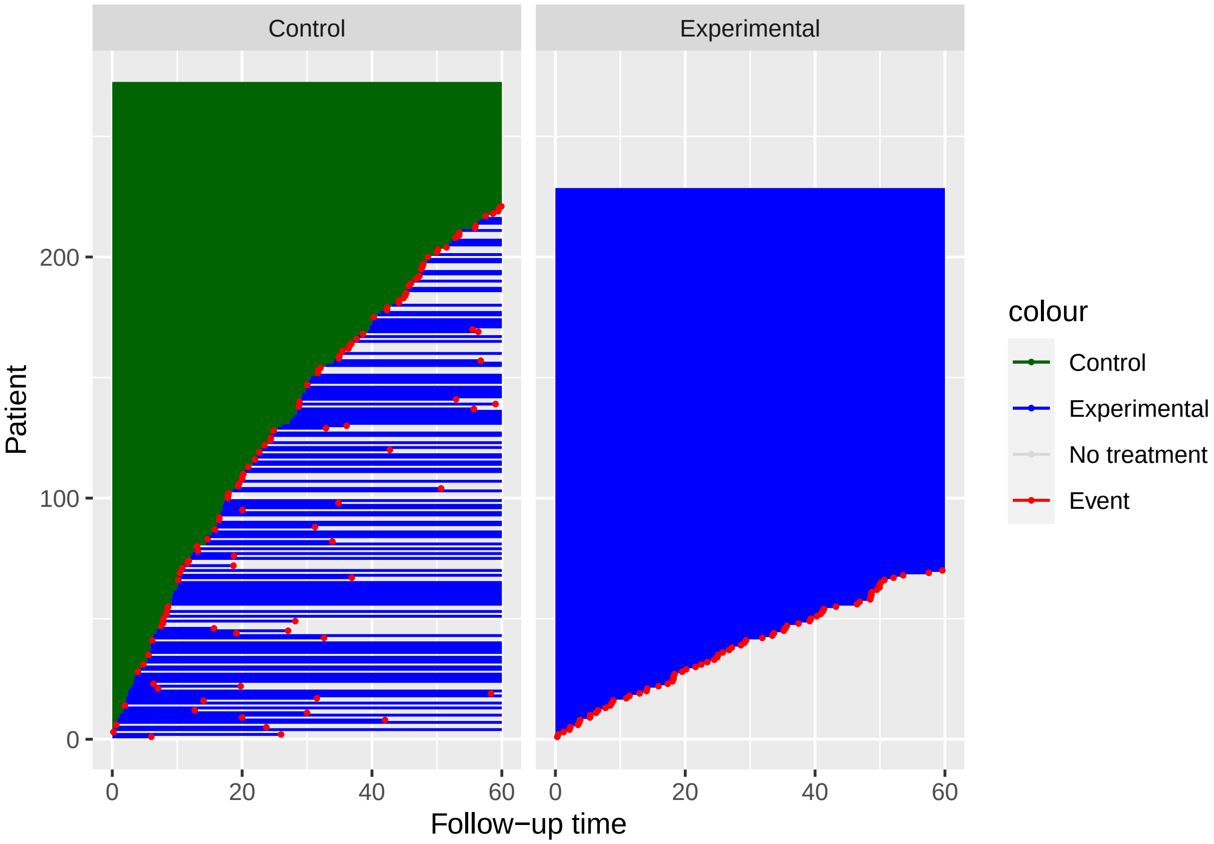
```
data$CO.indicator = ifelse(data$shift>0,yes="Yes",no="No") #Variable to indicate whether or not the pat
table(data$ARM)
```

```
##
##      Control Experimental
##          272          228
```

```
table(data$ARM,data$CO.indicator)
```

```
##
##                 No Yes
##   Control      132 140
##   Experimental 228   0
```

Patient pathway plot



Data analysis on Overall Survival endpoint

Treatment-policy estimand

```
## Additive Aalen Model
##
## Test for nonparametric terms
##
## Test for non-significant effects
##             Supremum-test of significance p-value H_0: B(t)=0
## (Intercept)                          8.31                   0
##
## Test for time invariant effects
##                   Kolmogorov-Smirnov test p-value H_0:constant effect
## (Intercept)                        0.0256                       0.546
##                     Cramer von Mises test p-value H_0:constant effect
## (Intercept)                        0.0108                       0.386
##
## Parametric terms :
##            Coef.      SE Robust SE    z   P-val lower2.5% upper97.5%
## const(Z) 0.00297 0.00112   0.00112 2.66 0.00777  0.000775    0.00517
##
##   Call:
## aalen(formula = Surv(time, status) ~ const(Z), data = data)
```

Wald type standard error and confidence interval

```
## Additive Aalen Model
##
## Test for nonparametric terms
##
## Test for non-significant effects
##             Supremum-test of significance p-value H_0: B(t)=0
## (Intercept)                          8.31                   0
##
## Test for time invariant effects
##                   Kolmogorov-Smirnov test p-value H_0:constant effect
## (Intercept)                        0.0256                       0.546
##                     Cramer von Mises test p-value H_0:constant effect
## (Intercept)                        0.0108                       0.386
##
## Parametric terms :
##            Coef.      SE Robust SE    z   P-val lower2.5% upper97.5%
## const(Z) 0.00297 0.00112   0.00112 2.66 0.00777  0.000775    0.00517
##
##   Call:
## aalen(formula = Surv(time, status) ~ const(Z), data = data)
```

```
##                   Estimand              Estimator Measure Value SE.type    SE
## const(Z) Treatment policy Additive Hazards Model    beta 0.003    Wald 0.001
##               p    LB    UB
## const(Z) 0.008 0.001 0.005
```

```
##                    Estimand              Estimator
## const(Z)  Treatment policy Additive Hazards Model
## const(Z)1 Treatment policy Additive Hazards Model
##                                                Measure Value SE.type    SE
## const(Z)                                          beta 0.003    Wald 0.001
## const(Z)1 Relative risk of surviving (exp. vs control) 1.195    Wald 0.080
##                p    LB    UB
## const(Z)  0.008 0.001 0.005
## const(Z)1 0.008 1.048 1.363
```

Score type standard error and confidence interval

Estimate β by minimizing the score equations

```
hazards = hazards.marginal(data)
beta = beta.estimate.itt(data=data,lower=-0.01,upper=0.01,hazards=hazards) #the lower and upper bounds
beta
```

```
## [1] 0.002977041
```

Now we estimate the p-value

```
fit.ivsacim = ivsacim(time=data$time, event=data$status, instrument=data$Z, IV_valid = TRUE, treatment_
fit.ivsacim.by_prod = fit.ivsacim$by_prod
stime = fit.ivsacim$stime


p.value = p.itt.score(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,stime=stime,hazards=hazards)
p.value
```

```
## [1] 0.00520822
```

SE for β:

```
beta.se = beta.itt.se.function(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,beta.est=beta,hazards=
beta.se
```

```
## [1] 0.001098524
```

Now we determine a 95% CI by minimising the quadratic difference between the p-value and 5%

```
LB = beta.bound.itt(lower=beta-10*beta.se,upper=beta,fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,
UB = beta.bound.itt(lower=beta,upper=beta+10*beta.se,fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,
```

```
##                    Estimand              Estimator
## const(Z)  Treatment policy Additive Hazards Model
## const(Z)1 Treatment policy Additive Hazards Model
## 1         Treatment policy Additive Hazards Model
## 11        Treatment policy Additive Hazards Model
##                                            Measure Value SE.type    SE
## const(Z)                                      beta 0.003    Wald 0.001
## const(Z)1 Relative risk of surviving (exp. vs control) 1.195    Wald 0.080
## 1                                             beta 0.003   Score 0.001
## 11        Relative risk of surviving (exp. vs control) 1.195   Score 0.079
##               p    LB    UB
## const(Z)  0.008 0.001 0.005
## const(Z)1 0.008 1.048 1.363
## 1         0.005 0.001 0.005
## 11        0.005 1.054 1.366
```

Using Instrumental Variables (g-estimation): method by Ying and Tchetgen Tchetgen 2022

(No baseline covariates are included)

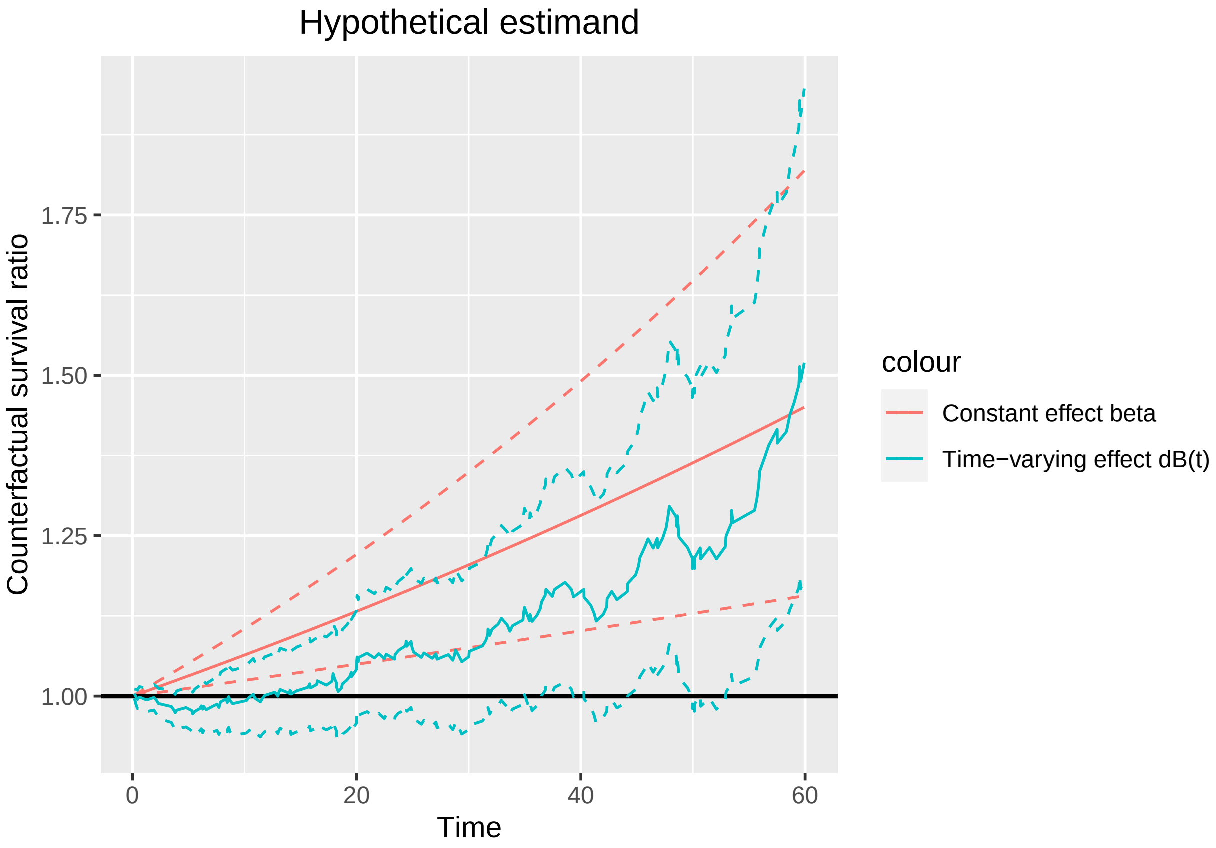
```
library(ivsacim) #install this package first if needed
fit.ivsacim <- ivsacim(time=data$time, event=data$status, instrument=data$Z, IV_valid = TRUE, treatment
summary(fit.ivsacim)
```

```
##
## Instrumental Variables Structural Additive Cumulative Intensity Model with a Valid IV
##
## Test for non-significant exposure effect. H_0: B_D(t) = 0
```

```
##
##          Supremum-test pval
## Exposure                  0
##
##
## Goodness-of-fit test for constant effects model. H_0: B_D(t) = beta_D * t
##          Supremum-test pval
##                       0.114
##
## Constant effect model: B_D(t) = beta_D * t
##                 coef se(coef) z-value p-value
## Exposure      0.0062  0.00193    3.22  0.0013
##
```

```
p.value = summary(fit.ivsacim)$pval_beta_D
```

Hypothetical survival risk ratio over time:



```
##                          Estimand              Estimator
## const(Z)       Treatment policy Additive Hazards Model
## const(Z)1      Treatment policy Additive Hazards Model
## 1              Treatment policy Additive Hazards Model
## 11             Treatment policy Additive Hazards Model
## 12          Hypothetical estimand                ivsacim
## 13          Hypothetical estimand                ivsacim
##                                                                Measure Value
## const(Z)                                                        beta 0.003
## const(Z)1               Relative risk of surviving (exp. vs control) 1.195
## 1                                                               beta 0.003
## 11                      Relative risk of surviving (exp. vs control) 1.195
## 12                                                              beta 0.006
## 13        Relative risk of surviving (always exp. vs always control) 1.450
##           SE.type    SE     p    LB    UB
## const(Z)     Wald 0.001 0.008 0.001 0.005
## const(Z)1    Wald 0.080 0.008 1.048 1.363
## 1           Score 0.001 0.005 0.001 0.005
## 11          Score 0.079 0.005 1.054 1.366
## 12           Wald 0.002 0.001 0.002 0.010
## 13           Wald 0.168 0.001 1.156 1.819
```

Score (Sandwich estimator) β estimate:

```
fit.ivsacim.by_prod = fit.ivsacim$by_prod
stime = fit.ivsacim$stime
beta = beta.estimate.iv.score(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,lower=-0.01,upper=0.01,s
beta
```

```
## [1] 0.005421263
```

Next, we estimate the SE of β using the Sandwich estimator

```
beta.se = beta.iv.se(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,beta.est=fit.ivsacim.by_prod$beta
beta.se
```

```
## [1] 0.002738459
```

Now we estimate the p-value

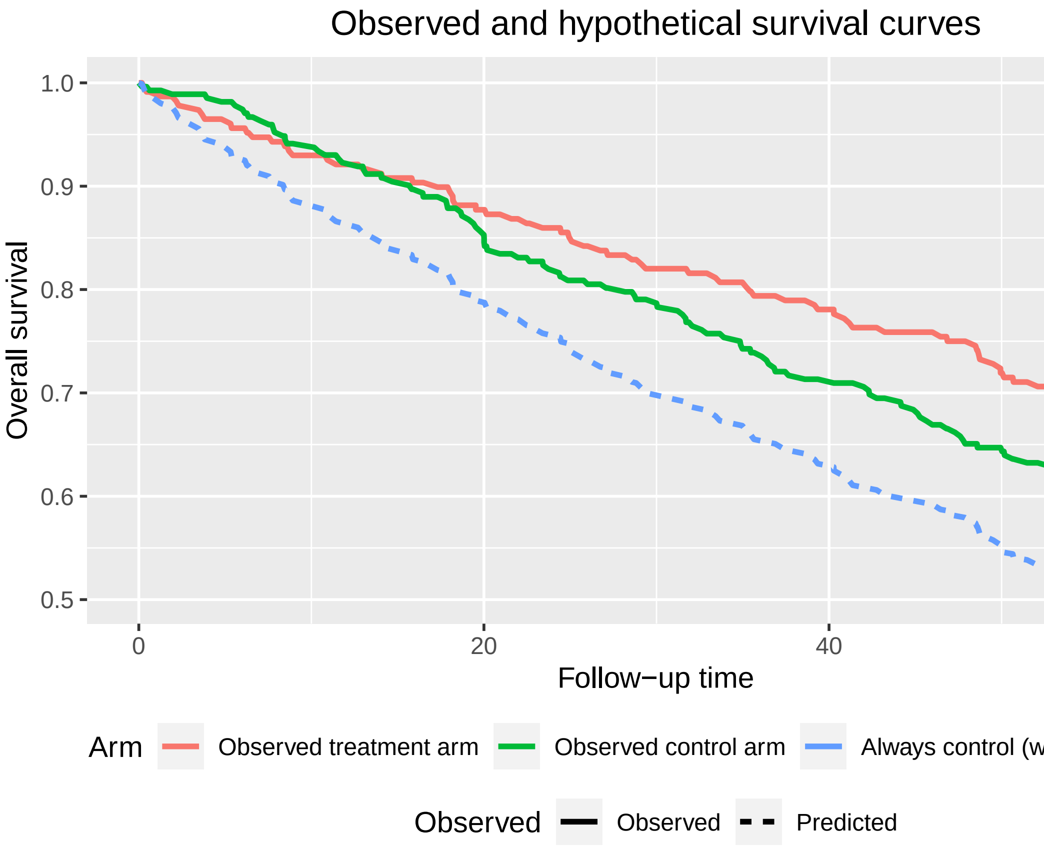
```
p.value = p.iv.score(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,stime=stime)
p.value
```

```
## [1] 0.02272636
```

Now we determine a 95% CI by minimising the quadratic difference between the p-value and 5%

```
LB = beta.bound.iv(lower=beta-10*beta.se,upper=beta,fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,st
UB = beta.bound.iv(lower=beta,upper=beta+10*beta.se,fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,st
```

```
##                         Estimand              Estimator
## const(Z)       Treatment policy Additive Hazards Model
## const(Z)1      Treatment policy Additive Hazards Model
## 1              Treatment policy Additive Hazards Model
## 11             Treatment policy Additive Hazards Model
## 12        Hypothetical estimand                ivsacim
## 13        Hypothetical estimand                ivsacim
## 14        Hypothetical estimand                ivsacim
## 15        Hypothetical estimand                ivsacim
##                                                                 Measure Value
## const(Z)                                                           beta 0.003
## const(Z)1               Relative risk of surviving (exp. vs control) 1.195
## 1                                                                  beta 0.003
## 11                      Relative risk of surviving (exp. vs control) 1.195
## 12                                                                 beta 0.006
## 13           Relative risk of surviving (always exp. vs always control) 1.450
## 14                                                                 beta 0.005
## 15           Relative risk of surviving (always exp. vs always control) 1.384
##           SE.type    SE     p    LB    UB
## const(Z)     Wald 0.001 0.008 0.001 0.005
## const(Z)1    Wald 0.080 0.008 1.048 1.363
## 1           Score 0.001 0.005 0.001 0.005
## 11          Score 0.079 0.005 1.054 1.366
## 12           Wald 0.002 0.001 0.002 0.010
## 13           Wald 0.168 0.001 1.156 1.819
## 14          Score 0.003 0.023 0.001 0.009
## 15          Score 0.227 0.023 1.052 1.759
```



Hypothetical survival curve

One-step estimator

1. Initial estimate for β using the IV estimator of Ying and Tchetgen Tchetgen

```
library(ivsacim)
fit.ivsacim <- ivsacim(time=data$time, event=data$status, instrument=data$Z, IV_valid = TRUE, treatment
summary(fit.ivsacim)
```

```
##
## Instrumental Variables Structural Additive Cumulative Intensity Model with a Valid IV
##
## Test for non-significant exposure effect. H_0: B_D(t) = 0
##
##          Supremum-test pval
## Exposure              0.002
##
##
## Goodness-of-fit test for constant effects model. H_0: B_D(t) = beta_D * t
##          Supremum-test pval
##                       0.106
##
## Constant effect model: B_D(t) = beta_D * t
##                 coef  se(coef) z-value p-value
## Exposure      0.0062  0.00193    3.22  0.0013
##
```

```
beta.start = fit.ivsacim$beta_D
stime = fit.ivsacim$stime
```

2. Calculate score function for every patient, evaluated in the inital β estimate and take the sample average.

```
#Z.formula=Z~1 #use this if you dont want to condition on baseline covariates
#L.values=c() #use this if you dont want to condition on baseline covariates
Z.formula=Z~gender+age #change the names of the baseline covariates
L.values=c("gender","age") #change the names of the baseline covariates

fit.ivsacim.by_prod = fit.ivsacim$by_prod
stime = fit.ivsacim$stime
#Sample average of the score function, evaluated in beta.start
av = av.function(fit.ivsacim.by_prod = fit.ivsacim.by_prod,data=data,beta.start=beta.start,stime=stime
av
```

```
## [1] -0.008687571
```

3. Calculate the derivative of the score function for every patient, evaluated in the inital β estimate and take the sample average.

```
## [1] -8.231402
```

4. Update the initial β estimate

```
beta.update = beta.start - av/av.deriv
beta.update
```

```
## [1] 0.005147617
```

```
beta = beta.update
```

Now we estimate the p-value

```
p.value = p.score(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,stime=stime,Z.formula=Z.formula,L.va
p.value
```

```
## [1] 0.04847756
```

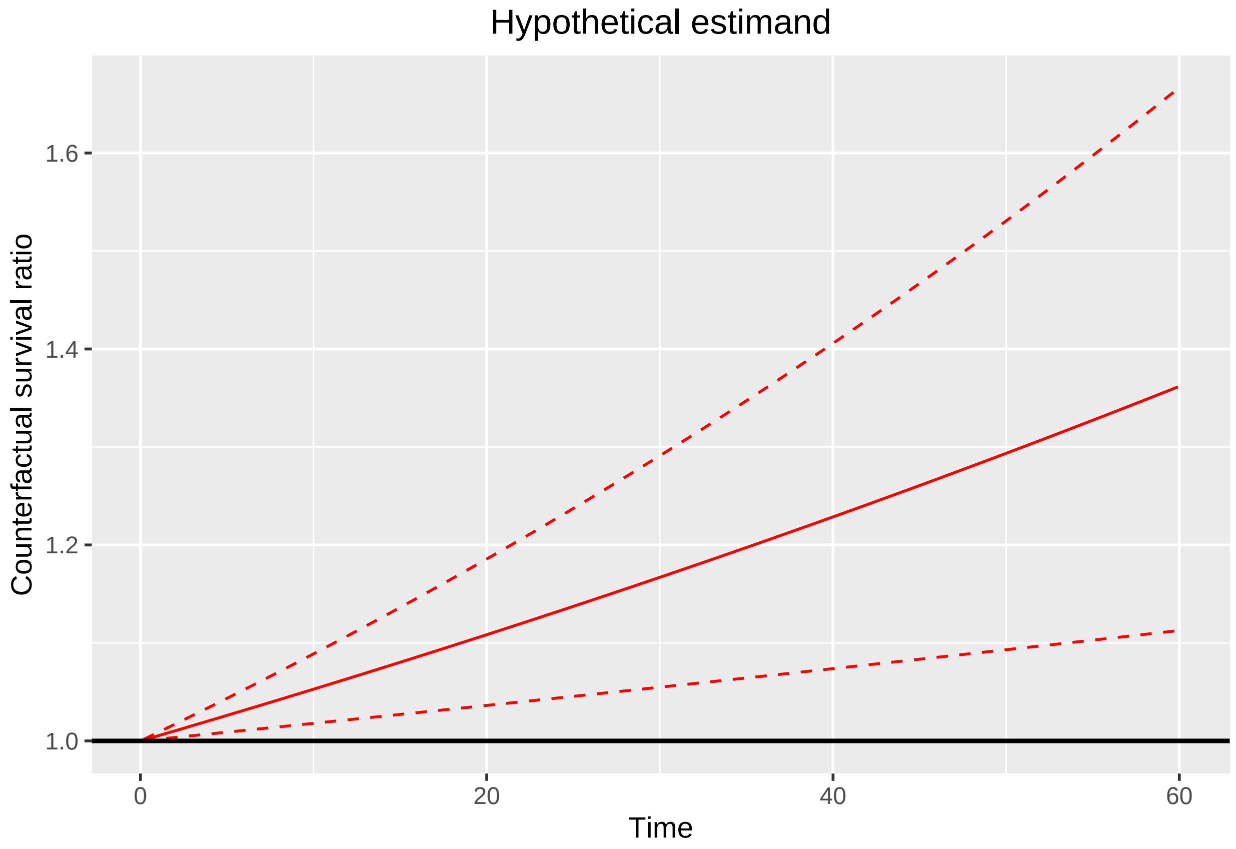
SE for β:

```
beta.se = beta.se.function(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,beta.est=beta,Z.formula=Z.
beta.se
```

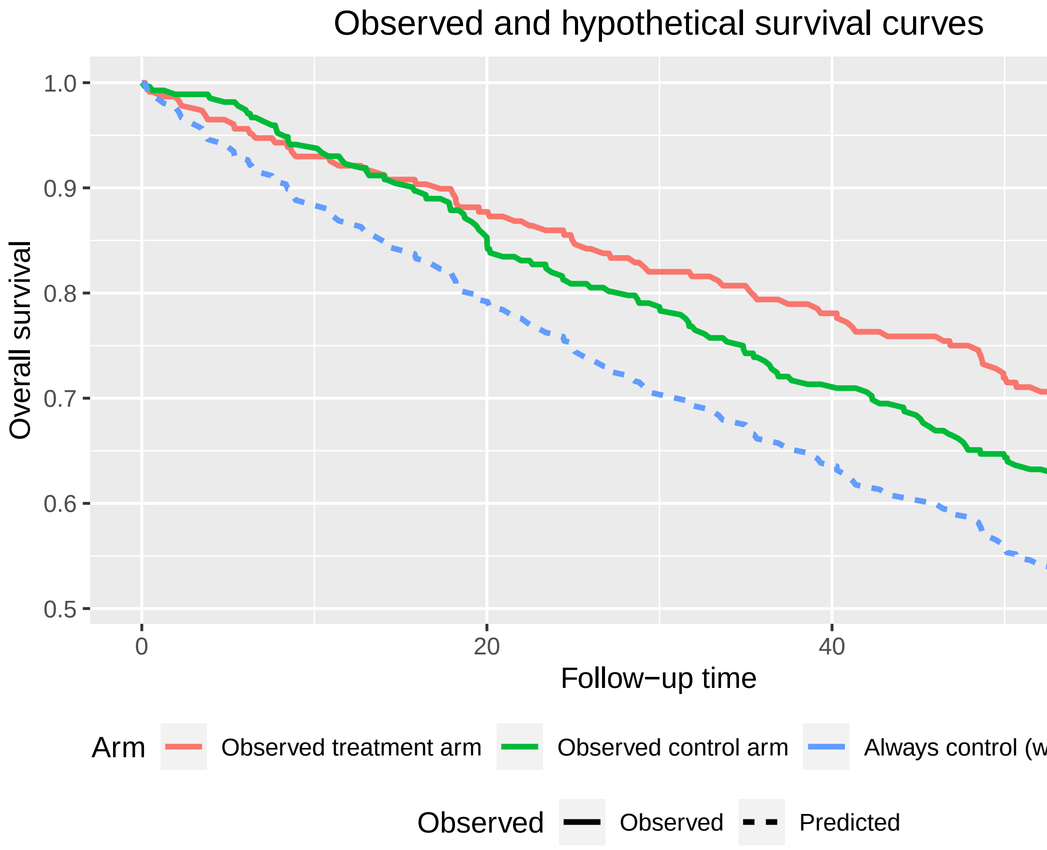
```
## [1] 0.001718241
```

Now we determine a 95% CI by minimising the quadratic difference between the p-value and 5%

```
LB = beta.bound(lower=beta-10*beta.se,upper=beta,fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,stime
UB = beta.bound(lower=beta,upper=beta+10*beta.se,fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,stime
```



```
##                         Estimand              Estimator
## const(Z)       Treatment policy Additive Hazards Model
## const(Z)1      Treatment policy Additive Hazards Model
## 1              Treatment policy Additive Hazards Model
## 11             Treatment policy Additive Hazards Model
## 12        Hypothetical estimand                ivsacim
## 13        Hypothetical estimand                ivsacim
## 14        Hypothetical estimand                ivsacim
## 15        Hypothetical estimand                ivsacim
## 16        Hypothetical estimand                Onestep
## 17        Hypothetical estimand                Onestep
##                                                              Measure Value
## const(Z)                                                        beta 0.003
## const(Z)1               Relative risk of surviving (exp. vs control) 1.195
## 1                                                               beta 0.003
## 11                      Relative risk of surviving (exp. vs control) 1.195
## 12                                                              beta 0.006
## 13        Relative risk of surviving (always exp. vs always control) 1.450
## 14                                                              beta 0.005
## 15        Relative risk of surviving (always exp. vs always control) 1.384
## 16                                                              beta 0.005
## 17        Relative risk of surviving (always exp. vs always control) 1.361
##           SE.type    SE     p    LB    UB
## const(Z)     Wald 0.001 0.008 0.001 0.005
## const(Z)1    Wald 0.080 0.008 1.048 1.363
## 1           Score 0.001 0.005 0.001 0.005
## 11          Score 0.079 0.005 1.054 1.366
## 12           Wald 0.002 0.001 0.002 0.010
## 13           Wald 0.168 0.001 1.156 1.819
## 14          Score 0.003 0.023 0.001 0.009
## 15          Score 0.227 0.023 1.052 1.759
## 16          Score 0.002 0.048 0.000 0.009
## 17          Score 0.140 0.048 1.001 1.718
```



Hypothetical survival curve

One-step estimator with hazard

1. Initial estimate for β using the IV estimator of Ying and Tchetgen Tchetgen

```
library(ivsacim)
fit.ivsacim <- ivsacim(time=data$time, event=data$status, instrument=data$Z, IV_valid = TRUE, treatment
summary(fit.ivsacim)
```

```
##
## Instrumental Variables Structural Additive Cumulative Intensity Model with a Valid IV
##
## Test for non-significant exposure effect. H_0: B_D(t) = 0
##
##           Supremum-test pval
## Exposure                 0
##
##
## Goodness-of-fit test for constant effects model. H_0: B_D(t) = beta_D * t
##           Supremum-test pval
##                      0.109
##
## Constant effect model: B_D(t) = beta_D * t
##                  coef se(coef) z-value p-value
## Exposure      0.0062  0.00193    3.22  0.0013
##
```

```
beta.start = fit.ivsacim$beta_D
stime = fit.ivsacim$stime
```

2. Calculate score function for every patient, evaluated in the inital β estimate and take the sample average.

```
hazards = hazards.conditional(data,T.formula = T.formula)

fit.ivsacim.by_prod = fit.ivsacim$by_prod
stime = fit.ivsacim$stime
#Sample average of the score function, evaluated in beta.start
av = av.function.hazard(fit.ivsacim.by_prod = fit.ivsacim.by_prod,data=data,beta.start=beta.start,stime=
av
```

```
## [1] -0.009675167
```

3. Calculate the derivative of the score function for every patient, evaluated in the inital β estimate and take the sample average.

```
## [1] -8.252618
```

4. Update the initial β estimate

```
beta.update = beta.start - av/av.deriv
beta.update
```

```
## [1] 0.005030659
```

```
beta = beta.update
```

Now we estimate the p-value

```
p.value = p.score.hazard(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,stime=stime,Z.formula=Z.formu
p.value
```

```
## [1] 0.007696374
```

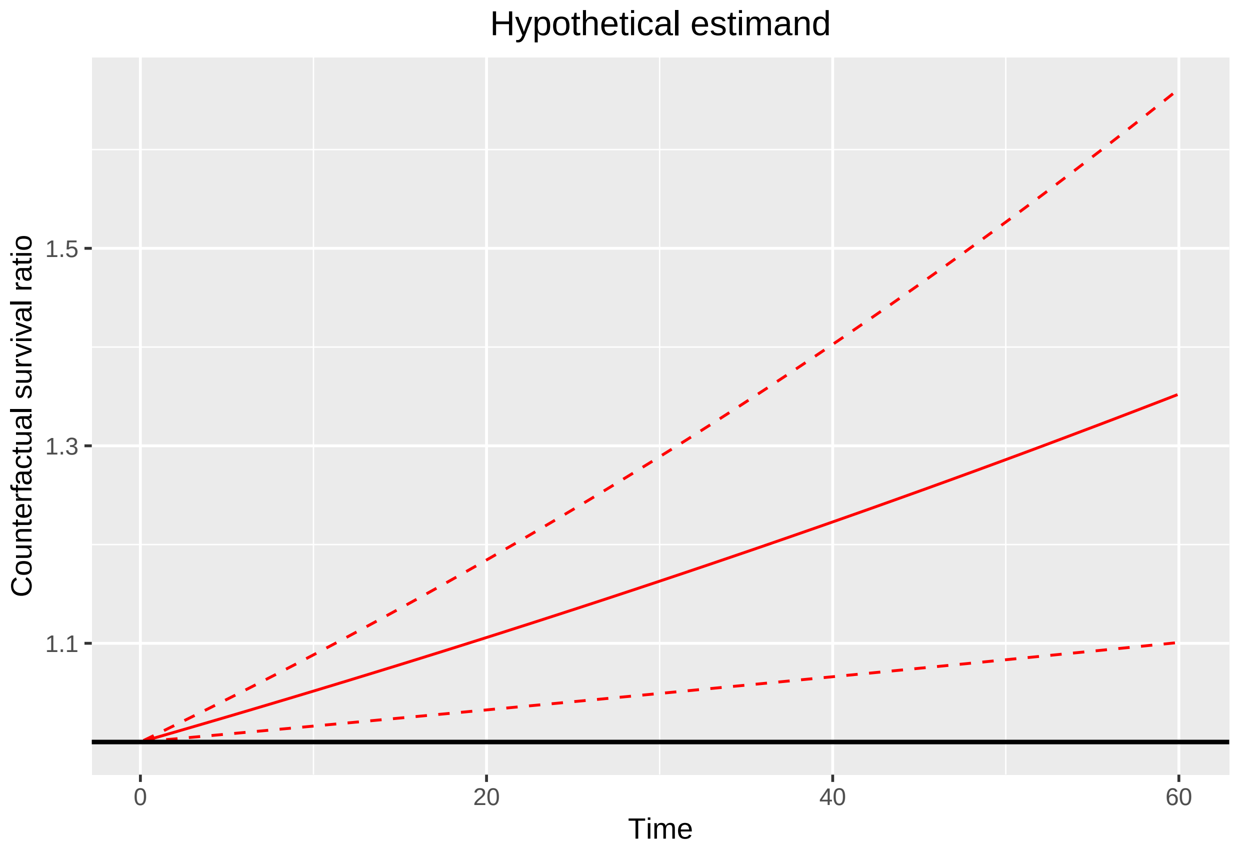
SE for β:

```
beta.se = beta.se.function.hazard(fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=data,beta.est=beta,Z.for
beta.se
```

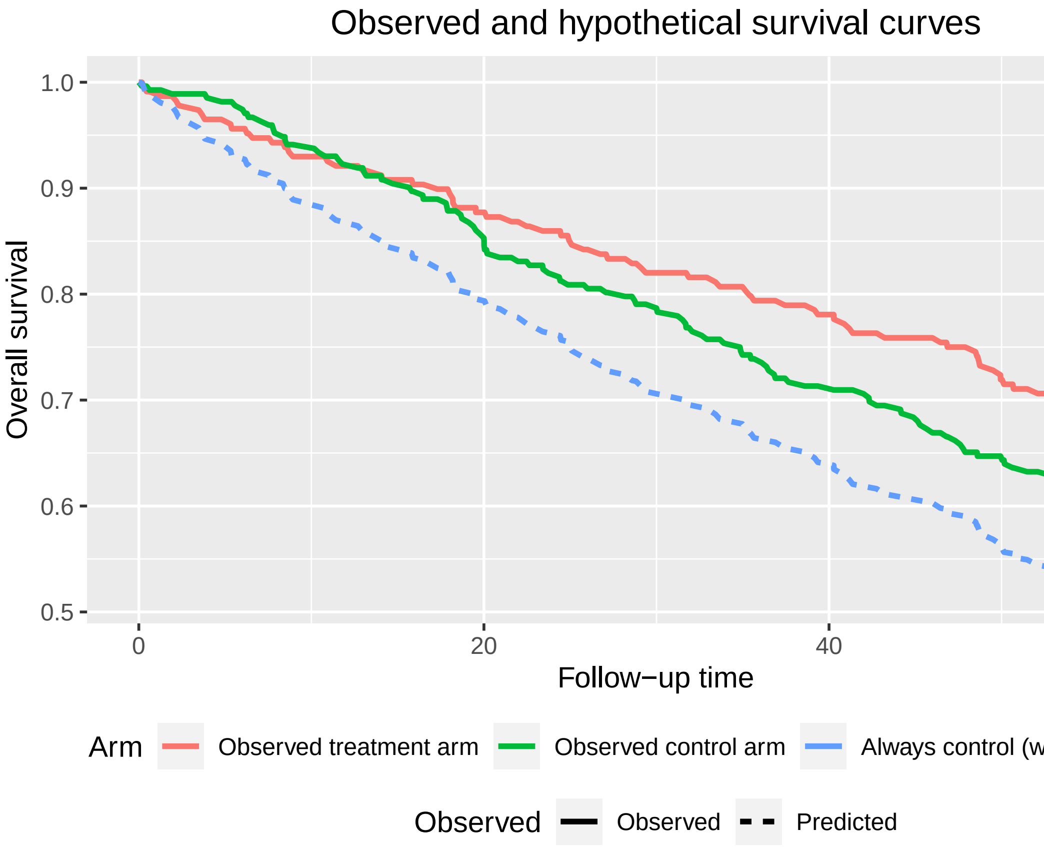
```
## [1] 0.001749919
```

Now we determine a 95% CI by minimising the quadratic difference between the p-value and 5%

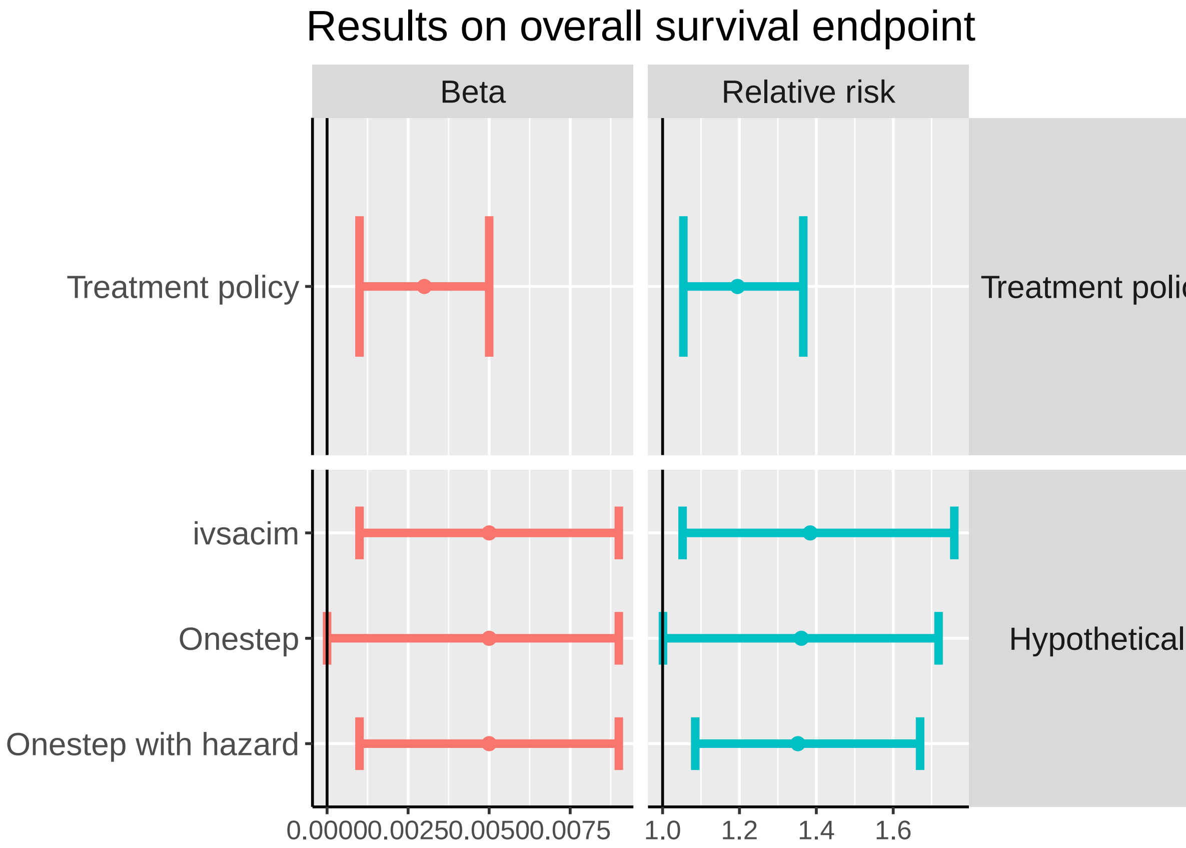
```
LB = beta.bound.hazard(lower=beta-10*beta.se,upper=beta,fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=da
UB = beta.bound.hazard(lower=beta,upper=beta+10*beta.se,fit.ivsacim.by_prod=fit.ivsacim.by_prod,data=da
```



```
##                         Estimand              Estimator
## const(Z)       Treatment policy Additive Hazards Model
## const(Z)1      Treatment policy Additive Hazards Model
## 1              Treatment policy Additive Hazards Model
## 11             Treatment policy Additive Hazards Model
## 12        Hypothetical estimand                ivsacim
## 13        Hypothetical estimand                ivsacim
## 14        Hypothetical estimand                ivsacim
## 15        Hypothetical estimand                ivsacim
## 16        Hypothetical estimand                Onestep
## 17        Hypothetical estimand                Onestep
## 18        Hypothetical estimand    Onestep with hazard
## 19        Hypothetical estimand    Onestep with hazard
##                                                               Measure Value
## const(Z)                                                        beta 0.003
## const(Z)1               Relative risk of surviving (exp. vs control) 1.195
## 1                                                               beta 0.003
## 11                      Relative risk of surviving (exp. vs control) 1.195
## 12                                                              beta 0.006
## 13        Relative risk of surviving (always exp. vs always control) 1.450
## 14                                                              beta 0.005
## 15        Relative risk of surviving (always exp. vs always control) 1.384
## 16                                                              beta 0.005
## 17        Relative risk of surviving (always exp. vs always control) 1.361
## 18                                                              beta 0.005
## 19        Relative risk of surviving (always exp. vs always control) 1.352
##           SE.type    SE     p    LB    UB
## const(Z)     Wald 0.001 0.008 0.001 0.005
## const(Z)1    Wald 0.080 0.008 1.048 1.363
## 1           Score 0.001 0.005 0.001 0.005
## 11          Score 0.079 0.005 1.054 1.366
## 12           Wald 0.002 0.001 0.002 0.010
## 13           Wald 0.168 0.001 1.156 1.819
## 14          Score 0.003 0.023 0.001 0.009
## 15          Score 0.227 0.023 1.052 1.759
## 16          Score 0.002 0.048 0.000 0.009
## 17          Score 0.140 0.048 1.001 1.718
## 18          Score 0.002 0.008 0.001 0.009
## 19          Score 0.142 0.008 1.085 1.670
```



Hypothetical survival curve



Plot of the results