Implementation Description for: How to transport causal effects across populations?

Vincent Brugger

2022-12-07

## Code for the Elements of good practice

The dataset was aquired from the freely available medicaldata package in R. We used all variables in the analysis except for one on post-operational bleeding that was removed due to excessive missingness. We used only complete cases and recoded the categorical variables as dummies. The sample was limited to the University of Michigan (UM) and the Indiana University (IU) trial site. As sample size we ended up with 577 individuals out of which 164 were from the target population (UM) and 413 from the source population (IU).  
This is the table we would expect to be seeing in transportability applications, the tests used here are homogeneity tests for categorical and binary variables and Mann-Whitney-U tests for continous variables, as we do not assume normality.

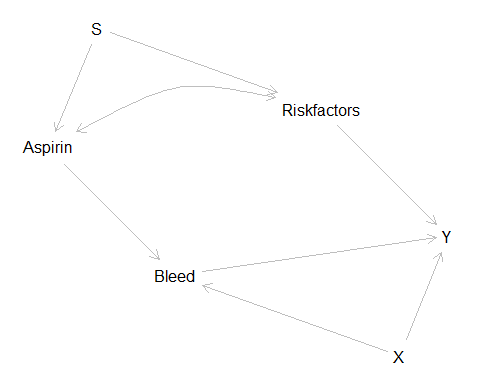
#include the Indomethacin trial dataset  
  
data\_read<-function(){  
 #read in data  
 dat=indo\_rct  
 #limit to the relevant sites   
 dat=dat[(dat$site=="1\_UM"|dat$site=="2\_IU"),]  
 #make it binary  
 dat$site=ifelse(dat$site=="2\_IU",1,0)  
 dat$rx=ifelse(dat$rx=="1\_indomethacin",1,0)  
 #remove bleed  
 dat=dat[,1:32]  
 #recode factors as dummies  
 for (k in 6:29){  
 dat[,k]=ifelse(dat[,k]=="0\_no",0,1)  
   
 }  
 dat$gender=ifelse(dat$gender=="1\_female",1,0) #gender==1 is female  
  
 dat$status=ifelse(dat$status=="0\_inpatient",0,1)  
 #make categorial variable into dummies  
 dat$type1=ifelse(dat$type=="1\_type 1",1,0)  
 dat$type2=ifelse(dat$type=="2\_type 2",1,0)  
 dat$type3=ifelse(dat$type=="3\_type 3",1,0)  
 #remove the ID and the type variable  
 dat=dat[,-31]  
 dat=dat[,-1]  
 #####  
 #omit rows with missings  
 data=na.omit(data.frame(subset(dat,select=-c(site,rx,outcome)),  
 Y=dat$outcome,Z=dat$rx,S=dat$site))  
 Y=data$Y  
 S=data$S  
 Z=data$Z  
 X=subset(data,select = -c(Z,S,Y))  
 return(list(S,Y,Z,X,data))  
}  
sol=data\_read()  
#the Population Dummy  
S=sol[[1]];  
#the Output data  
Y=sol[[2]];  
#the treatment variable  
Z=sol[[3]];  
#the matrix of covariates   
X=sol[[4]];  
#the dataset as whole  
data=sol[[5]];  
  
data$S=ifelse(data$S==1,"Source Population","Target Population")  
colnames(data)=c(  
 "Age in Years",  
 "Post hoc risk score",  
 "Female Gender",  
 "Sphincter of oddi dysfunction was present",  
 "Previous post-ERCP pancreatitis (PEP)",  
 "Recurrent Pancreatitis",  
 "Pancreatic Sphincterotomy was performed",  
 "Sphincter pre-cut was needed",  
 "Cannulation of the papilla was difficult",  
 "Pneumatic dilation of the papilla was performed",  
 "Ampullectomy was performed for dysplasia or cancer",  
 "Contrast was injected into the pancreas during the procedure",  
 "Pancreas appeared to have acinarization on imaging",  
 "Brushings were taken from the pancreatic duct",  
 "Aspirin was used at a dose of 81 mg per day",  
 "Aspirin was used at a dose of 325 mg per day",  
 "Aspirin was used at any dose",  
 "Pancreatic duct stent was placed at the end of the procedure per the judgement of the endoscopist",  
 "Pancreatic duct stent was placed in order to treat a clinically significant narrowing of the pancreatic duct",  
 "pancreatic duct stent was placed at the end of the procedure for any reason",  
 "Sphincter of oddi manometry was performed during the procedure for SOD",  
 "Biliary sphincterotomy was performed",  
 "Biliary stent was placed",  
 "Choledocholithiasis was present",  
 "Malignancy of the biliary duct or pancreas was found",  
 "A trainee participated in the ERCP",  
 "Outpatient status",  
 "Sphincter of Oddi dysfunction type 1",  
 "Sphincter of Oddi dysfunction type 2",  
 "Sphincter of Oddi dysfunction type 3",  
 "Outcome of post-ercp pancreatitis",  
 "Treatment arm",  
 "Site")

|  | Source Population | Target Population | p | test |
| --- | --- | --- | --- | --- |
| n | 413 | 164 |  |  |
| Age in Years (mean (SD)) | 44.45 (12.91) | 47.22 (14.20) | 0.024 |  |
| Post hoc risk score (mean (SD)) | 2.52 (0.85) | 2.06 (0.89) | <0.001 |  |
| Female Gender = 1 (%) | 345 (83.5) | 110 (67.1) | <0.001 |  |
| Sphincter of oddi dysfunction was present = 1 (%) | 392 (94.9) | 85 (51.8) | <0.001 |  |
| Previous post-ERCP pancreatitis (PEP) = 1 (%) | 61 (14.8) | 32 (19.5) | 0.203 |  |
| Recurrent Pancreatitis = 1 (%) | 102 (24.7) | 66 (40.2) | <0.001 |  |
| Pancreatic Sphincterotomy was performed = 1 (%) | 290 (70.2) | 43 (26.2) | <0.001 |  |
| Sphincter pre-cut was needed = 1 (%) | 15 ( 3.6) | 16 ( 9.8) | 0.006 |  |
| Cannulation of the papilla was difficult = 1 (%) | 84 (20.3) | 68 (41.5) | <0.001 |  |
| Pneumatic dilation of the papilla was performed = 1 (%) | 2 ( 0.5) | 0 ( 0.0) | 0.914 |  |
| Ampullectomy was performed for dysplasia or cancer = 1 (%) | 6 ( 1.5) | 11 ( 6.7) | 0.002 |  |
| Contrast was injected into the pancreas during the procedure = 1 (%) | 35 ( 8.5) | 23 (14.0) | 0.065 |  |
| Pancreas appeared to have acinarization on imaging = 1 (%) | 7 ( 1.7) | 19 (11.6) | <0.001 |  |
| Brushings were taken from the pancreatic duct = 1 (%) | 0 ( 0.0) | 1 ( 0.6) | 0.632 |  |
| Aspirin was used at a dose of 81 mg per day = 1 (%) | 30 ( 7.3) | 14 ( 8.5) | 0.730 |  |
| Aspirin was used at a dose of 325 mg per day = 1 (%) | 7 ( 1.7) | 6 ( 3.7) | 0.262 |  |
| Aspirin was used at any dose = 1 (%) | 36 ( 8.7) | 20 (12.2) | 0.264 |  |
| Pancreatic duct stent was placed at the end of the procedure per the judgement of the endoscopist = 1 (%) | 356 (86.2) | 82 (50.0) | <0.001 |  |
| Pancreatic duct stent was placed in order to treat a clinically significant narrowing of the pancreatic duct = 1 (%) | 24 ( 5.8) | 16 ( 9.8) | 0.133 |  |
| pancreatic duct stent was placed at the end of the procedure for any reason = 1 (%) | 379 (91.8) | 98 (59.8) | <0.001 |  |
| Sphincter of oddi manometry was performed during the procedure for SOD = 1 (%) | 281 (68.0) | 26 (15.9) | <0.001 |  |
| Biliary sphincterotomy was performed = 1 (%) | 239 (57.9) | 90 (54.9) | 0.575 |  |
| Biliary stent was placed = 1 (%) | 21 ( 5.1) | 11 ( 6.7) | 0.571 |  |
| Choledocholithiasis was present = 1 (%) | 5 ( 1.2) | 20 (12.2) | <0.001 |  |
| Malignancy of the biliary duct or pancreas was found = 1 (%) | 1 ( 0.2) | 1 ( 0.6) | 1.000 |  |
| A trainee participated in the ERCP = 1 (%) | 147 (35.6) | 124 (75.6) | <0.001 |  |
| Outpatient status = 1 (%) | 410 (99.3) | 145 (88.4) | <0.001 |  |
| Sphincter of Oddi dysfunction type 1 = 1 (%) | 29 ( 7.0) | 47 (28.7) | <0.001 |  |
| Sphincter of Oddi dysfunction type 2 = 1 (%) | 235 (56.9) | 28 (17.1) | <0.001 |  |
| Sphincter of Oddi dysfunction type 3 = 1 (%) | 128 (31.0) | 9 ( 5.5) | <0.001 |  |
| Outcome of post-ercp pancreatitis = 1 (%) | 41 ( 9.9) | 36 (22.0) | <0.001 |  |
| Treatment arm = 1 (%) | 206 (49.9) | 77 (47.0) | 0.588 |  |

data=sol[[5]];

The following chunk will generate the selection diagram and solve the query to display the transport formula.

SD\_input=dagitty("dag{  
 Riskfactors ->Y;  
 Aspirin-> Bleed;  
 X->Bleed;  
 Bleed->Y;  
 S-> Riskfactors;  
 X -> Y;  
 S-> Aspirin;  
 Aspirin->Bleed;  
 Riskfactors<->Aspirin;  
 X [exposure]  
 Y [outcome]  
}")  
  
plot(SD\_input)



data="p(Riskfactors,Y,Aspirin|do(X),S)  
 p(Riskfactors,Aspirin)"  
  
res=dosearch(data,  
 query= "p(Y|do(X))",graph=SD\_input,transportability = "S")  
res

## \sum\_{Aspirin,Riskfactors}\left(p(Aspirin,Riskfactors)p(Y|do(X),Aspirin,Riskfactors,S)\right)

In our analysis we include into Z all variables that were labelled as risk factors for Y together with gender and age. Further there is a variable bleeding that occurs after the treatment that was suspected to be a potential adverse event of the treatment in the original study. In case this adverse treatment is related to the outcome, we included it in the causal structure as mediator on the path from X to Y. In the trial, excessive aspirin intake is used as exclusion criterion it is believed to be a risk factor for post-surgery bleeding, which was unobserved for most cases in the data set and therefore excluded from the analysis. Though the test does not indicate any differing distributions in aspirin intake, we still include the respective node as it might be related to other risk factors. The *do-search* command yields the transport formula, thus indicating having to control for mentioned risk factors and aspirin.

In the following the assumptions for the transportability approach are tested using the data sample. The Omnibustest is used to verify the common outcome model as in our case the outcome data is available from both populations. It’s adapted from the implementation in Rudolph et al. (2017).

set.seed(1)  
run\_omnibus<-function(W.=X,A.=S,Y.=Y){  
 mmd\_test = function(R, S, D.R, D.S, sig.meth = 'eig', num.reps = 1e4,  
 return.cutoff = FALSE) {  
 n = length(R)  
  
 D.R.mat1 = matrix(rep(D.R,n),nrow=n)  
 D.R.mat2 = matrix(rep(D.R,each=n),nrow=n)  
  
 D.S.mat1 = matrix(rep(D.S,n),nrow=n)  
 D.S.mat2 = matrix(rep(D.S,each=n),nrow=n)  
  
 R.mat1 = matrix(rep(R,n),nrow=n)  
 R.mat2 = matrix(rep(R,each=n),nrow=n)  
  
 S.mat1 = matrix(rep(S,n),nrow=n)  
 S.mat2 = matrix(rep(S,each=n),nrow=n)  
  
 EE = ((2\*(R.mat1-R.mat2)\*(D.R.mat2-D.R.mat1) + 1 - (4\*(R.mat1-R.mat2)^2-2)\*D.R.mat1\*D.R.mat2)\*exp(-(R.mat1-R.mat2)^2)  
 - ((2\*(S.mat1-R.mat2)\*(D.R.mat2-D.S.mat1) + 1 - (4\*(S.mat1-R.mat2)^2-2)\*D.S.mat1\*D.R.mat2)\*exp(-(S.mat1-R.mat2)^2))  
 - ((2\*(R.mat1-S.mat2)\*(D.S.mat2-D.R.mat1) + 1 - (4\*(R.mat1-S.mat2)^2-2)\*D.R.mat1\*D.S.mat2)\*exp(-(R.mat1-S.mat2)^2))  
 + (2\*(S.mat1-S.mat2)\*(D.S.mat2-D.S.mat1) + 1 - (4\*(S.mat1-S.mat2)^2-2)\*D.S.mat1\*D.S.mat2)\*exp(-(S.mat1-S.mat2)^2))  
  
 # EE = exp(-(R.mat1-R.mat2)^2) - 2\*exp(-(S.mat1-R.mat2)^2) + exp(-(S.mat1-S.mat2)^2)  
  
 if(sig.meth=='eig'){  
 line.means = rowMeans(EE)  
 EE.ctrd = EE - matrix(rep(line.means,n),nrow=n) - matrix(rep(line.means,each=n),nrow=n) + matrix(rep(mean(line.means),n^2),nrow=n)  
 num.eigs = min(200,n)  
 # tmp = eigen(EE.ctrd)$values/n  
 tmp = eigs\_sym(EE.ctrd,num.eigs,which='LA')$values/n  
 num.pos.eigs = num.eigs # sum(tmp>0)  
 draws=c(matrix(rnorm(num.reps\*num.pos.eigs)^2-1,nrow=num.reps,ncol=num.pos.eigs)%\*%cbind(tmp[1:num.pos.eigs]))  
 }  
  
 # U-statistic  
 diag(EE) = 0  
 est = (rbind(rep(1/(n-1),n)) %\*% EE %\*% cbind(rep(1/n,n)))[1,1]  
 # V-statistic  
 # est = (rbind(rep(1/n,n)) %\*% EE %\*% cbind(rep(1/n,n)))[1,1]  
  
 if(sig.meth=='eig'){  
 pval = mean(draws>n\*est)  
 } else if(sig.meth=='var'){  
 pval = pchisq(est/(2\*var(D.R)/n)+1,df=1,lower.tail=FALSE)  
 }  
  
 return(if(!return.cutoff){  
 c(est,pval)  
 }else{  
 c(est,pval,  
 if(sig.meth=='eig'){  
 quantile(draws,0.95)  
 }else{  
 2\*var(D.R)\*(qchisq(0.95,df=1)-1)})})  
 }  
 est\_psi\_prob\_binom =  
 function(W=W., A=A., Y=Y., W.train = NULL, A.train = NULL, Y.train = NULL,  
 sig.meth = 'var', est.g = TRUE,  
 g0 = NULL,  
 SL.library = c('SL.glm', 'SL.step', 'SL.glm.interaction')) {  
 n=length(A)  
  
 if(is.null(W.train) | is.null(A.train) | is.null(Y.train)){  
 W.train = W  
 A.train = A  
 Y.train = Y  
 }  
  
 # Estimate outcome regressions  
 Qbar.est = SuperLearner(Y=Y.train,X=data.frame(W=W.train,A=A.train),newX=data.frame(W=rbind(W,W),A=rep(c(0,1),each=n)),SL.library=SL.library, family='binomial')  
 Qbar.est.0 = Qbar.est$SL.predict[,1][1:n]  
 Qbar.est.1 = Qbar.est$SL.predict[,1][(n+1):(2\*n)]  
  
 if(est.g){  
 gg = SuperLearner(Y=A,X=W,SL.library=SL.library,family='binomial')$SL.predict[,1]  
 } else {  
 gg = g0(W)  
 }  
  
 # Plug-in estimate of blip  
 R = Qbar.est.1 - Qbar.est.0  
 S = rep(0,n)  
  
 D.R = A/gg \* (Y-Qbar.est.1) - (1-A)/(1-gg) \* (Y-Qbar.est.0)  
 D.S = rep(0,n)  
  
 return(mmd\_test(R,S,D.R,D.S,sig.meth=sig.meth))  
 }  
  
 test=est\_psi\_prob\_binom(W=X,A=S,Y)  
 res=data.frame(qestimate=test[1],pvalue=test[2])  
 return(res)  
}  
run\_omnibus()

## qestimate pvalue  
## 1 -0.01763653 0.6826096

Now that the assumptions were not falsified, we turn to the estimation process. The targeted maximum likelihood estimator for transportability is adapted from Rudolph et al. (2017). This includes estimating the sampling, treatment and outcome model to correct the outcome model for bias. This was carried out using the SuperLearner package. To set up the SuperLearner algorithm (Van der Laan et al. 2007) details are given in the following. Firstly, if one has prior knowledge on the functional relationship between either outcome or treatment assignment or selection into the respective population of any observation, it should be leveraged. For example, if one knows from the experiment how the outcome or the treatment assignment was modeled, it could be a good guess to start from there. This case occurs for example if there was adjustment for imbalances in covariates. Including these in the modelling can be done through selecting libraries, as the “Superlearner” will need a list of such as an input, as candidates to find the best prediction model. Regarding the libraries, only realistically feasible libraries should be included into the consideration as others might by chance fit to the model and therefore, with little explanatory power, be included and bias the estimate. If more flexible libraries are included, it will be required to choose tuning parameters which can be estimated by the learner itself using cross validation. Further, entirely new learners or built-in ones with different tuning settings can be supplied as is described in the documentation. The ensembler then weights them together giving better tuning settings higher weights for the ensembled estimate. Secondly, feature selection naturally poses a challenge and different screening possibilities can be considered to pick a suitable set of variables before applying the estimators. This task can be carried out with several built-in solutions. However, in the transportability setting, there is little room for selecting covariates other than the sufficient set outlined in the transport formula, because if an optimizer would suggest using other features than the sufficient set, the causal relationship might violate the S-admissibility condition, given the causal structure was correctly specified. Therefore we recommend to base variable selection on theory rather than using the learner to suggest these. Thirdly, the user needs to choose an ensembling technique to weight the individual libraries and cross validation parameters to find the correct tuning for the individual libraries. The amount of cross validation has to be considered from a variance-bias trade-off. In the application, we used built-in functions of the “SuperLearner” package for generalized linear models (GLM), GLM with all multiplicative interactions between variables included as regressors and generalized additive models (GAM). Further, we use 5-fold cross validation, first internally (within-library settings) and then again externally (to weight the libraries) to obtain the best ensemble learner, leaving the other arguments at default setting. All three application cases were covered with this technique and similar settings. The function in the code yields an estimate of the transported average treatment effect and applies to the different situations by changing the data inputs. The data inputs for the three situations are reiterated in the results table.

Simulation of the Overcome Confounding case: A dataset was simulated using R, -the code is available below. It was set up so that the treatment and outcome depend on confounders in the target sample. Then, one of the confounders U was omitted from the target sample. In the source sample, the same process was run with different distributions for confounders, except for U. A random treatment dummy was set and outcomes were generated using the same model as in the target population, complying to our assumptions by construction. The sufficient set for transportability was naturally all confounders except for U, as S does not point to U in the selection diagram Figure 1. Further, another set of random treatment dummies and respective outcomes were generated in the target population to obtain the target average treatment effect as estimand.

#this function is needed to recover the results in the paper there is no further use for other projects   
simulate\_confounding\_example\_data<-function(){  
 #Simulation code for the Avoid confounding case  
 #S=0  
 set.seed(300)  
 n=5000  
 t\_conf1=rbinom(n,1,0.6)  
 t\_conf2=runif(n,-5,5)   
 t\_conf3=rbinom(n,1,0.2)  
   
 t\_treat=rbinom(n,1,t\_conf3\*0.2+t\_conf2\*0.01+0.3+t\_conf1\*0.05)  
 t\_rand=rbinom(n,1,0.5)  
 t\_outcome\_treat=rbinom(n,1,t\_conf3\*0.3+t\_conf2\*0.02+0.3+t\_conf1\*0.1\*t\_treat)  
 t\_outcome\_rand=rbinom(n,1,t\_conf3\*0.3+t\_conf2\*0.02+0.3+t\_conf1\*0.1\*t\_rand)  
 ######  
 #S=1  
 n=5000  
 s\_conf1=rbinom(n,1,0.3)  
 s\_conf2=runif(n,-5,5)   
 s\_conf3=rbinom(n,1,0.6)  
 s\_rand=rbinom(n,1,0.2)  
 s\_outcome=rbinom(n,1,s\_conf3\*0.3+s\_conf2\*0.02+0.3+s\_conf1\*0.1\*s\_rand)  
 conf1=c(s\_conf1,t\_conf1)  
 conf2=c(s\_conf2,t\_conf2)  
 conf3=c(s\_conf3,t\_conf3)  
 out=c(s\_outcome,t\_outcome\_treat)  
 treat=c(s\_rand,t\_treat)  
   
 data=data.frame(Y=out,S=c(rep(1,n),rep(0,n)),Z=treat,conf1,conf2)  
 S=data$S  
 Y=data$Y  
 Z=data$Z  
 X=subset(data,select = -c(Z,S,Y))  
   
 estcol=numeric(0)  
 cicol1=numeric(0)  
 cicol2=numeric(0)  
   
 return(list(S=S,Z=Z,Y=Y,X=X,estcol=estcol,cicol1=cicol1,cicol2=cicol2,  
 t\_conf1=t\_conf1,t\_conf2=t\_conf2,t\_conf3=t\_conf3,t\_treat=t\_treat,t\_rand=t\_rand,  
 t\_outcome\_treat=t\_outcome\_treat,t\_outcome\_rand=t\_outcome\_rand,s\_outcome=s\_outcome,  
 s\_rand=s\_rand))  
}  
  
set.seed(1)  
  
data=sol[[5]];  
lib=c("SL.glm","SL.glm.interaction","SL.gam")  
#this function generates the bootstrap samples and estimates the CI  
run\_bootstrap<-function(R=10,purpose,data,method,lib=c("SL.glm","SL.glm.interaction","SL.gam")){  
   
   
 validate\_boot<-function(data,indices,method,lib){  
 dat <- data[indices,]  
 S=data$S  
 X=subset(data,select = -c(Z,S,Y))  
 Y=data$Y  
 Z=data$Z   
 res<-Validate\_RCT\_results(S=S,Z=Z,Y=Y,X=X,method=method,lib.=lib)  
 res1=as.numeric(res[2])  
   
 return(res1)  
 }  
 overcome\_boot<-function(data,indices,method,lib){  
 dat <- data[indices,]  
 S=data$S  
 X=subset(data,select = -c(Z,S,Y))  
 Y=data$Y  
 Z=data$Z   
 res<-Avoid\_confoundig(Y=Y,Z=Z,S=S,X=X)  
 re1=res$est  
 return(re1)  
 }  
 gennew\_boot<-function(data,indices,method=parent.frame()$method,lib=parent.frame()$lib){  
 dat <- data[indices,]  
 S=data$S  
 X=subset(data,select = -c(Z,S,Y))  
 Y=data$Y  
 Z=data$Z   
 res<-Generate\_new\_Evidence()  
 res1=as.numeric(res[2])  
 return(res1)  
 }  
 Estimate\_validate=NULL  
 CI\_validate=NULL  
 Estimate\_gennew=NULL  
 CI\_gennew=NULL  
 Estimate\_overcome=NULL  
 CI\_overcome=NULL  
   
   
 if(purpose=="gennew"){  
 bootstrapIntervalls\_gennew=boot(data,gennew\_boot,R=R,method=method,lib=lib)  
 CI\_gennew=c(bootstrapIntervalls\_gennew$t0-1.96\*sd(bootstrapIntervalls\_gennew$t),bootstrapIntervalls\_gennew$t0+1.96\*sd(bootstrapIntervalls\_gennew$t))  
 Estimate\_gennew=bootstrapIntervalls\_gennew$t0  
 #res<-Generate\_new\_Evidence()  
 #ate=res[1]  
 ret=list(  
 #source\_ATE=ate,  
 Estimate\_gennew=Estimate\_gennew,CI\_gennew=CI\_gennew)  
 }  
 if(purpose=="validate"){  
 bootstrapIntervalls=boot(data,validate\_boot,R=R,method=method,lib=lib)  
 CI\_validate=c(bootstrapIntervalls$t0-1.96\*sd(bootstrapIntervalls$t),bootstrapIntervalls$t0+1.96\*sd(bootstrapIntervalls$t))  
 Estimate\_validate=bootstrapIntervalls$t0  
 #res<-Validate\_RCT\_results()  
 #ate=res[1]  
 ret=list(  
 #source\_ate=ate,  
 Estimate\_validate=Estimate\_validate,CI\_validate=CI\_validate)  
 }  
 if(purpose=="overcome"){  
 bootstrapIntervalls\_overcome=boot(data,overcome\_boot,R=R,method=method,lib=lib)  
 CI\_overcome=c(mean(bootstrapIntervalls\_overcome$t0)-1.96\*sd(bootstrapIntervalls\_overcome$t0),mean(bootstrapIntervalls\_overcome$t)+1.96\*sd(bootstrapIntervalls\_overcome$t))  
 Estimate\_overcome=bootstrapIntervalls\_overcome$t0  
 #res=Avoid\_confoundig(Y=Y,Z=Z,S=S,X=X)  
 #cate=res$CATE  
 #sate=res$SATE  
 ret=list(  
 #source\_ate=sate,confounded\_ate=cate,  
 Estimate\_overcome=Estimate\_overcome,CI\_overcome=CI\_overcome)  
 }  
   
 return(ret)  
}  
  
#the following functions are for the different Purposes of transportability  
Generate\_new\_Evidence<-function(S=parent.frame()$S,  
 Z=parent.frame()$Z,  
 Y=parent.frame()$Y,  
 X=parent.frame()$X,  
 lib.=parent.frame()$lib,  
 method=parent.frame()$method){  
 ATE=mean(Y[S==1&Z==1])-mean(Y[S==1&Z==0])  
 #TATE=mean(Y[S==0&Z==1])-mean(Y[S==0&Z==0])  
 est=numeric(0)  
 if (method=="TMLE"){  
 #Outcome and treatment set to zero giving no information on these cases to the learner  
 #Will not work with NA's as the Super Learner function cannot handle these  
 Z[S==0]=0  
 Y[S==0]=0  
 n=transport\_compare(z=Z,y=Y,site=S,w=X,lib=lib.)  
 est=n$est  
 }  
 return(list(ATE,est))  
}  
  
  
Validate\_RCT\_results<-function(S,Z,Y,X,method=method,lib.=lib){  
 ATE=mean(Y[S==1&Z==1])-mean(Y[S==1&Z==0])  
  
 TATE=mean(Y[S==0&Z==1])-mean(Y[S==0&Z==0])  
 est=numeric(0)  
 if (method=="TMLE"){  
 n=transport\_compare(z=Z,y=Y,site=S,w=X,lib=lib.)  
 est=n$est  
 }  
 return(list(ATE,est))  
}  
Avoid\_confoundig<-function(S,Z,Y,X,  
 lib.=parent.frame()$lib,  
 method=parent.frame()$method){  
 CATE=mean(Y[S==0&Z==1])-mean(Y[S==0&Z==0])  
 SATE=mean(Y[S==1&Z==1])-mean(Y[S==1&Z==0])  
 if (method=="TMLE"){  
  
 #TATE=mean(t\_outcome\_rand[t\_rand==1])-mean(t\_outcome\_rand[t\_rand==0])  
 p=transport\_compare(z=parent.frame()$Z,y=parent.frame()$Y,site=parent.frame()$S,w=parent.frame()$X,lib=lib.)  
 est=p$est  
 }  
 Out=list(est=est,SATE=SATE,CATE=CATE)  
 return(Out)  
}  
  
#here the functions are called and the results are generated  
purpose="validate"  
g=run\_bootstrap(R=100,purpose=purpose,lib=lib,method="TMLE",data=data)  
print(g)

## $Estimate\_validate  
## [1] -0.1374948  
##   
## $CI\_validate  
## [1] -0.1539089 -0.1210806

purpose="gennew"  
g=run\_bootstrap(R=100,purpose=purpose,lib=lib,method="TMLE",data=data)  
print(g)

## $Estimate\_gennew  
## [1] -0.1453351  
##   
## $CI\_gennew  
## [1] -0.1595674 -0.1311028

purpose="overcome"  
set.seed(1)  
simdat=simulate\_confounding\_example\_data()  
data=data.frame(cbind(X=simdat$X,Y=simdat$Y,S=simdat$S,Z=simdat$Z))  
g=run\_bootstrap(R=100,purpose=purpose,lib=lib,method="TMLE",data=data)

CATE=mean(data$Y[data$S==0&data$Z==1])-mean(data$Y[data$S==0&data$Z==0])

SATE=mean(data$Y[data$S==1&data$Z==1])-mean(data$Y[data$S==1&data$Z==0])

TATE=mean(simdat$t\_outcome\_rand[simdat$t\_rand==1])-mean(simdat$t\_outcome\_rand[simdat$t\_rand==0])

print(g);CATE;SATE;TATE;

## $Estimate\_overcome  
## [1] 0.0258065  
##   
## $CI\_overcome  
## [1] 0.00570169 0.04591131

## [1] 0.1092836

## [1] 0.02093532

## [1] 0.0281078

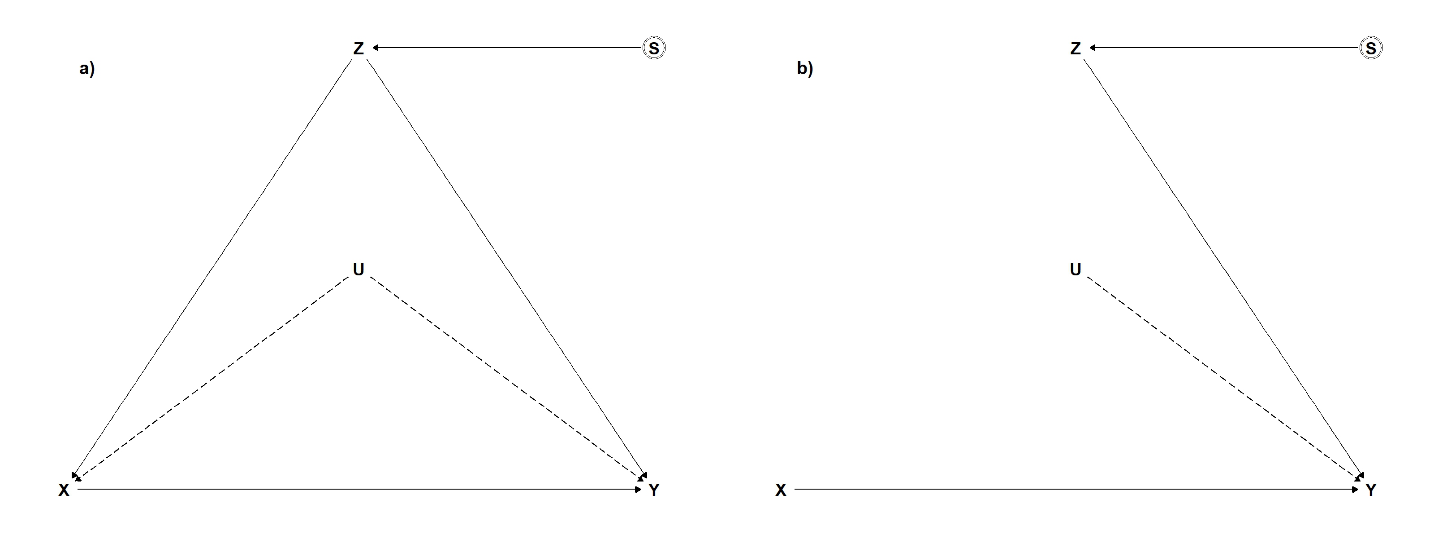
## Simulation

The simulation code for the overcome confounding bias purpose can be seen in the estimation step. A treatment variable is generated building on three confounders. In the source population all of these are observed, in the target population only two of them are. Importantly the distribution of these three is different only in the two observed in both populations and the same in the third that is unobserved in the target. From these confounders and the treatment assignment, an outcome variable is computed using a common outcome model. The confounders are distributed according to:

## Further cases

Additionally to the case in the main text, here a general case is displayed that might be of value for researchers applying the method in their own case. On the left a pre-intervention diagram is displayed and on the right one where the treatment was randomized (post-treatment diagram) is shown.  
In the diagrams, all covariates are grouped together as set Z, as the dependencies between them are neither known nor relevant as long as the variables are effect modifiers and not mediators and all Z’s are controlled for. As can be seen form the p-values in Table 1, the S-node does point to Z as some distributions differ significantly between the two populations. Note that there is no arrow pointing from S to U i.e. the distributions of unobserved confounders U do not differ between the two populations. To wrap up the 3 core assumptions depicted in the selection diagrams:

1. Some or all of the covariates have differing distributions in the target and source population (Z containing all observed covariates and S pointing in Z)
2. The distributions of any non-observed confounder do not differ between the populations (S is not pointing into U)
3. Outcomes are generated by a common causal mechanism in both populations given the covariates (The S-node does not point into Y)

* 

The respective transport formula is given by: The derivation can be found in Manke-Reimers et al. 2023.