R Lab 4 - IPTW & the Positivity Assumption

Introduction to Causal Inference

Goals:
1. Implement IPTW for statistical parameter equal to the average treatment effect under the necessary causal assumptions.
2. Implement IPTW for parameters of a MSM.
3. Understand how the IPTW estimator is affected by "near" positivity violations and weight stabilization.

Next lab:
We will implement targeted maximum likelihood estimation (TMLE).

1 Background Story

Given our previous success at predicting whether or not a pirate ship will find buried treasure, the Queen has hired us to estimate the causal effect of scurvy on mortality among pirates. The data available on n pirates are

- $W1$: possession of at least one awesome pirate characteristic (e.g. peg leg, eye patch, beard, parrot). This is a binary covariate measured when leaving port (yes:1, no:0).
- $W2$: a summary measure of route danger, including voyage length, hurricane season, travel through enemy waters, Bermuda triangle, etc. This is a categorical variable ranging from 0 as least dangerous to 3 as most dangerous.
- A: whether the pirate suffered from scurvy during the voyage. This is a binary exposure (yes:1, no:0).
- Y: pirate's mortality status. This is a binary outcome with 1 for deceased and 0 for alive.

1.1 Causal Road Map Rundown

Please note: We are doing a very fast review here. In practice, each step of the road map requires very careful thinking.

1. **Specify the Question:**
   What is the causal effect of scurvy on mortality among pirates?

2. **Specify the causal model:**
   - Endogenous nodes: $X = (W1, W2, A, Y)$, where $W1$ is an indicator for having an "awesome" pirate characteristic, $W2$ is a summary measure of route danger, A is scurvy status and Y is mortality.
   - Exogenous nodes: $U = (U_{W1}, U_{W2}, U_A, U_Y) \sim P_U$. We make no assumptions about the distribution P_U.
   - Structural equations F:

$$\begin{aligned} W1 &= f_{W1}(U_{W1}) \\ W2 &= f_{W2}(W1, U_{W2}) \\ A &= f_A(W1, W2, U_A) \\ Y &= f_Y(W1, W2, A, U_Y) \end{aligned}$$

   There are no exclusion restrictions or assumptions about functional form.

Image 1: https://www.flickr.com/photos/talklikeapirateday/3933458622/in/set-990505
Image 2: http://www.huffingtonpost.com/2013/09/24/sir-stuffington-one-eyed-kitten_n_3982907.html

3. **Specify the causal parameter of interest:**
   We are interested in the causal risk of death due to scurvy (i.e. the average treatment effect):

   $$\Psi^F(P_{U,X}) = E_{U,X}(Y_1) - E_{U,X}(Y_0) = P_{U,X}(Y_1 = 1) - P_{U,X}(Y_0 = 1)$$

   where Y_a denotes the counterfactual outcome (mortality), if possibly contrary to fact, the pirate had scurvy status $A = a$.

4. **Specify the link between the SCM and the observed data:**
   The observed data were generated by sampling n independent times from a data generating system compatible with the structural causal model $\mathcal{M}^F$. This yield n i.i.d. copies of random variable $O = (W1, W2, A, Y) \sim P_0$. The statistical model $\mathcal{M}$ for the set of allowed distributions of the observed data is non-parametric.

5. **Assess identifiability:**
   In the original SCM $\mathcal{M}^F$, the target causal parameter is not identified from the observed data distribution. A sufficient, but not minimal, identifiability assumption is that all of the exogenous errors are independent. Other possibilities include $U_A \perp\!\!\!\perp U_Y$ and (i) $U_A \perp\!\!\!\perp U_{W1}$, $U_A \perp\!\!\!\perp U_{W2}$ or (ii) $U_Y \perp\!\!\!\perp U_{W1}$, $U_Y \perp\!\!\!\perp U_{W2}$. We use $\mathcal{M}^{F*}$ to denote the original SCM augmented by the additional causal assumptions needed for identifiability. We introduce this "working" SCM to keep our real knowledge separate from our wished for identifiability assumptions. Under $\mathcal{M}^{F*}$, the backdoor criteria holds conditionally on $W = (W1, W2)$.

   For identifiability, we also need the positivity assumption to hold:

   $$min_{a \in \mathcal{A}}\; P_0(A = a|W = w) > 0$$

   for all w for which $P_0(W = w) > 0$. In words, we need that each possible exposure level is represented in every covariate strata. This condition on data support ensures that our statistical estimand is well-defined. Specifically, we need a non-zero probability of the conditioning set in the G-computation formula. In our example, there must be a positive probability of having scurvy or not, within strata of "awesome" pirate status and route danger. Here, we are using $W = (W1, W2)$ to denote the set of covariates that satisfy the backdoor criteria under the working SCM $\mathcal{M}^{F*}$.

6. **Specify the target parameter of the observed data distribution:**
   Under the working SCM $\mathcal{M}^{F*}$, the average treatment effect $\Psi^F(P_{U,X})$ is identified using the G-Computation formula:

   $$\begin{aligned}\Psi(P_0) &= E_0\big[E_0(Y|A = 1, W) - E_0(Y|A = 0W)\big]\\ &= \sum_w \big[E_0(Y|A = 1, W = w) - E_0(Y|A = 0, W = w)\big]P_0(W = w)\end{aligned}$$

   where $W = (W1, W2)$. This is our statistical estimand.

7. **Estimate the chosen parameter of the observed data distribution:**
   We have discussed two estimators of the statistical parameter. They rely on estimating different parts of the observed data distribution P_0:

   (a) Simple substitution estimator based on the G-Computation formula:

   $$\hat{\Psi}_{SS}(P_n) = \frac{1}{n}\sum_{i=1}^{n}\left[\bar{Q}_n(1,W_i) - \bar{Q}_n(0,W_i)\right]$$

   where P_n is the empirical distribution and $\bar{Q}_n(A,W)$ is the estimate of the conditional mean outcome given the exposure (scurvy) and baseline covariates $E_0(Y|A,W)$.

   (b) **Inverse probability weighted estimator (IPTW)**:

   $$\hat{\Psi}_{IPTW}(P_n) = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=1)}{g_n(A_i|W_i)}Y_i - \frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=0)}{g_n(A_i|W_i)}Y_i$$

   where $g_n(A_i|W_i)$ is an estimate of the conditional probability of scurvy given the baseline characteristics $P_0(A|W)$. This conditional distribution is often referred to as the exposure or treatment mechanism. IPTW is the focus of today's lab.

   (c) TMLE: coming soon :)

8. **Inference and interpret results:** Coming soon.

2 Import and explore data set `RLab4.IPTW.csv`.

1. Use the `read.csv` function to import the data set and assign it to data frame `ObsData`.

2. Use the `nrow` function to count the number of pirates in the data set. Assign this number as `n`.

3. Use the `names`, `tail` and `summary` functions to explore the data.

4. With the `table` function, the number of pirates in each covariate strata without scurvy $A = 0$ and the number of pirates in each covariate strata with scurvy $A = 1$. *Note: these tables are simply counting the number of observations within each strata of* $(W1, W2, A)$ *in a single sample of size n; we are not formally evaluating the positivity assumption, which is a statistical assumption on the true data generating process* P_0.

```
> table(ObsData$W1,ObsData$W2, ObsData$A)
```

Solution:

```
> ObsData<- read.csv('RLab4.IPTW.csv')
> # assign the number of pirates to random variable n
> n<- nrow(ObsData)
> n


[1] 5000


> # get the column names
> names(ObsData)


[1] "W1" "W2" "A"  "Y"
```

```
> # show the obsv data on the last six pirates
> tail(ObsData)


     W1 W2 A Y
4995  0  1 1 1
4996  1  2 1 1
4997  1  0 0 0
4998  1  2 0 1
4999  1  0 0 0
5000  0  2 1 1


> # recall: W1-awesomeness, W2-danger, A-scurvy, Y-mortality
> summary(ObsData)


       W1               W2               A                Y
 Min.   :0.0000   Min.   :0.000   Min.   :0.0000   Min.   :0.000
 1st Qu.:0.0000   1st Qu.:1.000   1st Qu.:0.0000   1st Qu.:1.000
 Median :1.0000   Median :2.000   Median :1.0000   Median :1.000
 Mean   :0.5158   Mean   :1.492   Mean   :0.6784   Mean   :0.762
 3rd Qu.:1.0000   3rd Qu.:2.000   3rd Qu.:1.0000   3rd Qu.:1.000
 Max.   :1.0000   Max.   :3.000   Max.   :1.0000   Max.   :1.000


> # use the table function to examine support in exposure-covariate strata
> table(ObsData$W1,ObsData$W2, ObsData$A)


, ,  = 0



      0   1   2   3
  0 258 118  12   0
  1 303 782 134   1

, ,  = 1



      0   1   2   3
  0  72 773 900 288
  1   2 191 861 305


> # top table corresponds to no scurvy A=0 and bottom table corresponds to scurvy A=1
> # the rows correspond to W1 (awesomeness) and W2 (danger)
```

There are certain covariate combinations with little or no variability in the exposure (scurvy). For example, there are zero "not-awesome" pirates ($W1 = 0$) without scurvy ($A = 0$) in the highest level of route danger ($W2 = 3$). Likewise, there are only 2 "awesome" pirates ($W1 = 1$) with scurvy ($A = 1$) in the lowest level of route danger ($W2 = 0$). In the following subsection, we will explore how sparsity (i.e. lack of data support) can affect estimator performance.

3 Implement the IPTW for the G-computation estimand

1. Estimate the treatment mechanism $P_0(A|W) = g_0(A|W)$, which is the conditional probability of scurvy, given the pirate's characteristics. Use the following *a priori*-specified parametric regression model:

   $$g_0(A = 1|W) = expit\big[\beta_0 + \beta_1 W1 + \beta_2 W2\big]$$

   Hint: Run `glm` with specifications `family='binomial'` for logistic regression and `data=ObsData`.
   In practice, we would generally use a data-adaptive estimator, such as SuperLearner. To save time during lab, we are using the correctly specified parametric regression.

2. Predict each pirate's probability of his observed exposure (scurvy status), given his covariates: $g_n(A_i|W_i)$...

   (a) Obtain the predicted probability of having scurvy, given the baseline covariates `pred.g1W`.
   Hint: Apply the `predict` function to the above fitted logistic regression function with `type='response'`.

   (b) Obtain the predicted probability of not having scurvy, given the baseline covariates `pred.g0W`:

   $$g_n(0|W) = 1 - g_n(1|W)$$

   (c) Create an empty vector `gAW` of length n.

   (d) Among pirates with scurvy, assign the appropriate predicted probability:

   ```
   > gAW[ObsData$A==1] <- pred.g1W[ObsData$A==1]
   ```

   (e) Among pirates without scurvy, assign the appropriate predicted probability:

   ```
   > gAW[ObsData$A==0] <- pred.g0W[ObsData$A==0]
   ```

   (f) Use the `summary` function to examine the distribution of the predicted probabilities. Any cause for concern?

3. Create a vector `wt` as the inverse of the predicted probability. Use the `summary` function to examine the distribution of the weights.

4. Evaluate the IPTW estimand by taking the empirical mean of the weighted outcomes:

   $$\hat{\Psi}_{IPTW}(P_n) = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i = 1)}{g_n(A_i|W_i)}Y_i - \frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i = 0)}{g_n(A_i|W_i)}Y_i$$

   Hint: We can code indicator functions with `as.numeric` applied to a logical statement.

   ```
   > #indicator that A_i=1 can be coded as
   > as.numeric(ObsData$A==1)
   ```

5. Comment on the results.

6. Arbitrarily truncate the weights at 10 and evaluate the IPTW estimand.
   Hint: The following code copies the weight vector (`wt`) into a new vector (`wt.trunc`) and truncates the weights at 10.

   ```
   > wt.trunc<- wt
   > wt.trunc[wt.trunc > 10]<- 10
   ```

7. Implement the stabilized IPTW estimator (a.k.a. the modified Horvitz-Thompson estimator):

   $$\hat{\Psi}_{St.IPTW}(P_n) = \frac{\frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=1)}{g_n(A_i|W_i)}Y_i}{\frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=1)}{g_n(A_i|W_i)}} - \frac{\frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=0)}{g_n(A_i|W_i)}Y_i}{\frac{1}{n}\sum_{i=1}^{n}\frac{\mathbb{I}(A_i=0)}{g_n(A_i|W_i)}}$$

   Dividing by the mean of the weights ensures that the IPTW estimator is bounded. This formula may look intimidating. Just divide empirical mean of the weighted outcomes by the empirical mean of the weights.

Solution:

```
> # 1.  Estimate the treatment mechanism g(A|W)=P(A|W)
> # run the logistic regression to estimate the treatment mechanism P(A|W)= g(A|W)
> gAW.reg<- glm(A ~ W1 +W2, family="binomial", data=ObsData)
> gAW.reg$coef

(Intercept)          W1          W2
  -1.332659   -3.224437    3.184136

> # 2. # predicted probability of having scurvy, given the obs cov P(A=1|W) = g(1|W)
> pred.g1W <- predict(gAW.reg, type= "response")
> # predicted probability of not having scurvy, given the obs cov P(A=0|W)=g(0|W)
> pred.g0W <- 1 - pred.g1W

> # we need the predicted prob of the observed treatment (scurvy), given covariates.
> # create an empty vector
> gAW <- rep(NA, n)
> # for pirates with scurvy, gAW = P(A=1 | W)
> gAW[ObsData$A==1] <- pred.g1W[ObsData$A==1]
> # for pirates without scurvy, gAW = P(A=0 | W)
> gAW[ObsData$A==0] <- pred.g0W[ObsData$A==0]
> # look at the distribution of predicted probabilities
> summary(gAW)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
0.00646 0.79790 0.86430 0.82190 0.99330 0.99970
```

IPTW is extremely sensitive to theoretical and practical positivity violations. From the summary of the estimated treatment mechanism $g_n(A|W)$, we see that there are certain covariate combinations with little variability in the exposure (scurvy status).

```
> # 3. Each subject gets a weight inverse weight inverse to pred prob
> wt<- 1/gAW
> # look at the distribution of weights
> summary(wt)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  1.000   1.007   1.157   2.064   1.253 154.800
```

"Near" violations of the positivity assumptions often yield poor finite sample performance. Here, at least one pirate is being up-weighted by 154.8.

```
> # 4. Point estimate:  weighted empirical mean outcome for pirates with scurvy
> # minus the weighted empirical mean for pirates without scurvy
> IPTW<- mean( wt*as.numeric(ObsData$A==1)*ObsData$Y) -
+   mean( wt*as.numeric(ObsData$A==0)*ObsData$Y)
> IPTW

[1] 0.1974928
```

5. The IPTW estimate of $\Psi(P_0)$ is 19.75%. The true value is $\psi_0 = 26.30\%$. Recall that the treatment mechanism $g_0(A|W)$ was estimated with a correctly *a priori*-specified model.

```
> # 6.  truncate  weights ARBITRARILY at 10
> # first let's see how many weights are greater than 10
> sum(wt>10)
```

```
[1] 15
```

```
> wt.trunc<- wt
> wt.trunc[ wt.trunc>10] =10
> # evaluate the IPTW estimand with the truncated weights
> mean( wt.trunc*as.numeric(ObsData$A==1)*ObsData$Y) -
+   mean( wt.trunc*as.numeric(ObsData$A==0)*ObsData$Y)
```

```
[1] 0.5437832
```

By bounding the predicted probabilities (weights), we are ensuring that the estimator of the treatment mechanism $g_0(A|W)$ is not consistent and thereby the resulting IPTW estimator will be biased. The point estimate from the IPTW estimator, using truncated weights, was 54.38%; the true value of the statistical estimand was $\psi_0 = 26.3\%$.

```
> # 7. Stabilized IPTW estimator - Modified Horvitz-Thompson estimator
> mean( wt*as.numeric(ObsData$A==1)*ObsData$Y)/mean( wt*as.numeric(ObsData$A==1)) -
+   mean( wt*as.numeric(ObsData$A==0)*ObsData$Y)/mean( wt*as.numeric(ObsData$A==0))
```

```
[1] 0.2783772
```

For this single sample of n pirates, the modified Horvitz-Thompson estimator yielded a point estimate 27.8% that was closest to the true value. *To evaluate the performance of these 3 IPTW estimators, we could draw another independent sample of size n, implement the three estimators, and repeat 500+ times. Then we could evaluate the bias, variance and mean squared error of these estimators for this data generating process. See the Appendix as well as `R` Lab2 and `R` homework 2.*

4 IPTW & Marginal Structural Models

Recall: Marginal structural models (MSMs) are just another way to define the target parameter. Specifically, they provide a summary measure of how the expected counterfactual outcome Y_a changes as a function of treatment A and possibly effect modifiers of interest, denoted V.

Suppose the Queen is interested in how the expected counterfactual mortality Y_a varies as function of scurvy A and route danger $V = W2$. Specifically, she believes that the effect of scurvy on death is modified by the route danger. Consider the following MSM to summarize how the average counterfactual outcome mortality Y_a varies of as a function scurvy A and route danger V:

$$E_{U,X}(Y_a|V) = m(a,V|\beta) = expit\big[\beta_0 + \beta_1 a + \beta_2 V + \beta_3 a^* V\big]$$

For simplicity, we are treating this MSM as the truth. In class, we will cover estimation of parameters of *working*

MSMs using IPTW with and without stabilized weights. This is step 1 of the causal road map: specifying the question. See `R` lab 1 for further discussion and worked examples.

4.1 Implement IPTW for the MSM parameter without stabilized weights:

1. Estimate the treatment mechanism $P_0(A|W) = g_0(A|W)$, which is the conditional probability of scurvy, given the pirate's characteristics.
   Hint: We already did this! Skip to next step.

2. Predict each pirate's probability of his observed exposure (scurvy status), given his covariates `gAW`.
   Hint: We already did this! Skip to next step.

3. Create the weight vector `wt` as the inverse of the predicted probabilities:

$$wt_i = \frac{g^*(A_i)}{g_n(A_i|W_i)}, \text{ where } g^*(A_i) = 1$$

   Hint: We already did this! Skip to the next step.

4. Estimate the β coefficients by regressing the outcome Y on the exposure A and effect modifier $V = W2$ according to the MSM.
   Hint: Use `glm` to run logistic regression. Specify the `weights`, the `family` and the `data`.

5. Interpret the results.

Solution:

```
> #########
> # IPTW estimation for MSM parameter
> ########
> # 1. we already estimated  the treatment mechanism g_0(A|W)
> # 2. we already obtained the predicted probability of the obsv exps, given cov
> # 3. we already created the weight vector
> tail(cbind(ObsData$A, gAW, wt))


                gAW       wt
[4995,] 1 0.8643004 1.157005
[4996,] 1 0.8595040 1.163462
[4997,] 0 0.9896165 1.010492
[4998,] 0 0.1404960 7.117638
[4999,] 0 0.9896165 1.010492
[5000,] 1 0.9935398 1.006502


> # 4. estimate the parameters of the MSM with IPTW
> IPTW.msm<- glm(Y~A + W2 + A:W2, weights=wt, family='binomial', data=ObsData )
> IPTW.msm$coef


(Intercept)           A          W2        A:W2
  -2.490257    2.386680    2.349990    2.253354
```

```
> # note on the warnings:
> # Warning messages:
> # 1: In eval(expr, envir, enclos) :
> #  non-integer #successes in a binomial glm!
>
> # these warnings are bc the weighted outcomes might be less than 0 or greater than 1.
> # IPTW is still working :)
```

5. The estimated parameters of the MSM are

$$\begin{aligned} m(a,V|\hat{\beta}) &= expit[\hat{\beta}_0 + \hat{\beta}_1 a + \hat{\beta}_2 V + \hat{\beta}_3 a^* V] \\ &= expit[-2.49 + 2.39^* a + 2.35^* V + 2.25^* a^* V] \end{aligned}$$

This suggests that route danger $V = W2$ does, indeed, modify the effect of scurvy A on mortality Y. The true parameters of the MSM are

$$m(a,V|\beta) = expit[-2.60 + 2.62^* a + 2.60^* V + 1.94^* a^* V]$$

See the Appendix for a detailed interpretation of these estimates, under the necessary causal assumptions and treating the MSM as true.

4.2 Weight stabilization in IPTW for an MSM parameter

For MSMs with effect modifiers $m(a,V|\beta)$, the numerator of the weights can be any function of the exposure A and the effect modifier V. A common choice is the conditional probability of the observed exposure, given the effect modifier:

$$st.wt_i = \frac{g_n^*(A_i|V_i)}{g_n(A_i|W_i)}, \text{ where } g_n^*(A_i|V_i) = P_n(A_i|V_i)$$

This can help reduce variability in the weights. If the covariates in W no longer predict the exposure (whether a pirate has scurvy) after controlling for the effect modifier (route's danger), then the weights will be 1. All the control for confounding is taken care by adjustment for $V = W2$ in the MSM. In less extreme cases, this choice of numerator can still increase efficiency. Pirates will get weighted based on the conditional probability of their observed exposure given the effect modifier $g_n(a|V)$ relative to the conditional probability of their observed exposure given all the baseline covariates $g_n(a|W)$.

The positivity assumption for a parameter defined with MSM $m(a,V|\beta)$ is also weakened:

$$sup_{a\in\mathcal{A}} \frac{g^*(a,V)}{g_0(a|w)} < \infty \text{ for all } w \text{ for which } P_0(W=w) > 0$$

For any (a,V) combinations that occur with a non-zero probability, we need that the conditional probability of that exposure a given the baseline covariates to also be non-zero. (We will cover this in more detail in class.)

4.3 Implement IPTW for a MSM parameter with stabilized weights:

1. Estimate the treatment mechanism $P_0(A|W) = g_0(A|W)$. *Hint: We already did this! Skip to next step.*

2. Predict each pirate's probability of his observed exposure (scurvy status), given his covariates `gAW`. *Hint: We already did this! Skip to next step.*

3. Create the stabilized weights `wt.MSM`:

$$st.wt_i = \frac{g_n^*(A_i|V_i)}{g_n(A_i|W_i)}, \text{ where } g_n^*(A_i|V_i) = P_n(A_i|V_i)$$

(a) Estimate the conditional probability of the scurvy, given route danger $g_0(A = 1|V)$ with a saturated logistic regression:

```
> gAV.reg <- glm(A ~ as.factor(W2), family="binomial", data=ObsData)
```

We are using the `as.factor` function to create dummy variables for each level of route danger $W2$.

(b) Obtain the predicted probability of having scurvy, given the route danger: `pred.g1V`.
Hint: Apply the `predict` function to the above fitted logistic regression function with `type='response'`.

(c) Calculate the predicted probability of not having scurvy, given the route danger: `pred.g0V`:

$$g_n(0|V) = 1 - g_n(1|V)$$

(d) Create an empty vector `gAV` of length n.

(e) Among pirates with scurvy, assign the appropriate predicted probability:

```
> gAV[ObsData$A==1] <- pred.g1V[ObsData$A==1]
```

(f) Among pirates without scurvy, assign the appropriate predicted probability:

```
> gAV[ObsData$A==0] <- pred.g0V[ObsData$A==0]
```

(g) Create the stabilized weights:

```
> wt.MSM<- gAV/gAW
```

(h) Look at the distribution of the stabilized weights.

4. Estimate the parameters of the MSM by regressing the observed outcome Y on the exposure A and effect modifier V according to the MSM. Specify the `weights`, the `family` and the `data`.

5. Comment on the resulting estimates.

Solution:

```
> #########
> # IPTW estimation for MSM parameter with stabilized weights
> ########
> # 1.  we already estimated  the treatment mechanism g_0(A|W)
> # 2. we already obtained the predicted probability of the obsv exps, given cov


> # 3. creating the stabilized weights
> # running the saturated logistic regression
> gAV.reg <- glm(A ~ as.factor(W2), family="binomial", data=ObsData)
> gAV.reg


Call:  glm(formula = A ~ as.factor(W2), family = "binomial", data = ObsData)

Coefficients:
   (Intercept)  as.factor(W2)1  as.factor(W2)2  as.factor(W2)3
        -2.026           2.094           4.516           8.411

Degrees of Freedom: 4999 Total (i.e. Null);  4996 Residual
Null Deviance:            6281
Residual Deviance: 4085         AIC: 4093
```

```
> # predicting the probability of scurvy, given the route danger g(1|V)
> pred.g1V<- predict(gAV.reg, type="response")
> # predicted probability of no scurvy, given the route danger g(0|V)
> pred.g0V<- 1- pred.g1V


> # create vector for numerator
> gAV<- rep(NA, n)
> # for pirates with scurvy g1V = P(A=1|V)
> gAV[ObsData$A==1] <- pred.g1V[ObsData$A==1]
> # for pirates without scurvy g0V = P(A=0|V)
> gAV[ObsData$A==0] <- pred.g0V[ObsData$A==0]


> # creating the stabilized weights
> wt.MSM<- gAV/gAW
> summary(wt.MSM)


   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.2504  0.6052  0.9294  1.0070  1.0740 11.8500


> # much less extreme!
```

This numerator choice leads to less variable weights.

```
> # 4. estimate the parameters of the MSM with IPTW
> IPTW.msm.st<- glm(Y~A + W2 + A:W2, weights=wt.MSM, family='binomial', data=ObsData )
> IPTW.msm.st$coef


(Intercept)           A          W2        A:W2
  -2.608820    2.551799    2.479631    2.012014
```

5. The estimated parameters of the MSM using IPTW with stabilized weights are

$$\begin{aligned} m(a,V|\hat{\beta}) &= expit[\hat{\beta}_0 + \hat{\beta}_1 a + \hat{\beta}_2 V + \hat{\beta}_3 a^* V] \\ &= expit[-2.61 + 2.55^* a + 2.48^* V + 2.01^* a^* V] \end{aligned}$$

Here, we are using a parametric MSM to smooth over values of the effect modifier V. So the estimates are not identical.

To compare performance metrics (such as bias, variance and MSE), we should sample another n pirates from the data generating system, implement the two IPTW estimators of the MSM parameter, and repeat many times. See Appendix.

Solution:

Appendix A: a specific data generating process

The following code was used to generate the data set `RLab4.IPTW.csv`. In this data generating process (one of many compatible with the SCM $\mathcal{M}^F$), all exogenous errors are independent.

```
> #-----------
> # genData: function to generate the data
> # input: sample size n
> # output: data frame with W1, W2, A, Y, as well as counterfactuals Y.1 & Y.0
> #-------
> genData<- function(n){
+   W1 <- rbinom(n, size=1, prob=.5) # characteristics
+   W2 <- rbinom(n, size=3, prob=.5) # route safety from 0(safe)- 3 (dangerous)
+   A <- rbinom(n, size =1, prob=plogis(-1.3 - 3*W1 +3*W2))
+   Y<- rbinom(n, size=1, prob= plogis(-2 - 2*W1 +3*W2 +3*A+ 2*A*W2 ))
+   Y.1<- rbinom(n, size=1, prob= plogis(-2 - 2*W1 +3*W2 +3*1+ 2*1*W2 ))
+   Y.0<- rbinom(n, size=1, prob= plogis(-2 - 2*W1 +3*W2 +3*0+ 2*0*W2 ))
+   data.frame(W1,W2, A, Y, Y.1,Y.0)
+ }


> # create the RLab4.IPTW.csv
> set.seed(252)
> Full<- genData(n=5000)
> ObsData<- subset(Full, select=c(W1,W2,A,Y))
> write.csv(ObsData, file="RLab4.IPTW.csv", row.names=F)
```

- Given this specific data generating process, we could estimate the true value of the average treatment effect by drawing a huge number of observations and taking the mean difference in the counterfactual outcomes.

```
> set.seed(252)
> # calculate true ATE by drawing a huge number of observations
> nTot=100000
> Full <- genData(n=nTot)
> Psi.F <- mean(Full$Y.1)- mean(Full$Y.0)
> Psi.F


[1] 0.26432
```

The counterfactual risk of mortality would 26.3% higher if all pirates had scurvy than if all pirates did not have scurvy.

- Now consider again the MSM to summarize how the average counterfactual outcome mortality (Y_a) varies of as a function scurvy (A) and route danger (V):

$$E_{U,X}(Y_a|V) = m(a, V|\beta) = expit\big[\beta_0 + \beta_1 a + \beta_2 V + \beta_3 a^* V\big]$$

Given the above data generating process, we would you calculate the value of the target causal parameter, which is the vector of coefficients β that summarize how the counterfactual outcome changes as a function of the exposure and effect modifier, by regressing the counterfactual outcomes Y_a on A and V according to the MSM.

```
> # calculate true parameters of MSM
> # recall the Full data set consist of (W1,W2,A, Y, Y.1, Y.0) on 100,000 pirates
> Y.A<- c(Full$Y.1, Full$Y.0) # create a vector of the stacked counterfactuals
> A <- c(rep(1, nTot), rep(0, nTot)) # the corresponding exposure levels
> V <- rep(Full$W2, 2) # effect modifier repeated twice
> true.MSM <- glm(Y.A~ A*V, family="binomial") #regress according to MSM
> true.MSM$coef


(Intercept)           A           V         A:V
  -2.595871    2.616669    2.601609    1.937133
```

The true parameters (coefficients) of the MSM are

$$E_{U,X}(Y_a|V) = m(a, V|\beta) = expit[-2.60 + 2.62^{*}a + 2.60^{*}V + 1.94^{*}a^{*}V]$$

This suggests that route danger $V = W2$ does, indeed, modify the effect of scurvy A on counterfactual mortality Y_a:

$$\begin{aligned}
logit[m(a, V = 0|\beta)] &= -2.60 + 2.62^{*}a + 2.60^{*}0 + 1.94^{*}a^{*}0 = -2.60 + 2.62^{*}a \\
logit[m(a, V = 1|\beta)] &= -2.60 + 2.62^{*}a + 2.60^{*}1 + 1.94^{*}a^{*}1 = 4.56^{*}a \\
logit[m(a, V = 2|\beta)] &= -2.60 + 2.62^{*}a + 2.60^{*}2 + 1.94^{*}a^{*}2 = 2.60 + 6.38^{*}a \\
logit[m(a, V = 3|\beta)] &= -2.60 + 2.62^{*}a + 2.60^{*}3 + 1.94^{*}a^{*}3 = 5.2 + 9.92^{*}a
\end{aligned}$$

Because we are assuming our MSM represents real knowledge about the form of $E_{U,X}[Y_a|V]$ (rather than a projection of the truth as in a working MSM), we may interpret the coefficients as in classical linear logistic regression:
- In the lowest level of route danger ($V = 0$), the counterfactual odds of mortality would be $e^{-2.60+2.62^{*}1}/e^{-2.60} = 13.74$ times higher if all pirate had scurvy than if no pirates had scurvy.
- In the first level of route danger ($V = 1$), the counterfactual odds of mortality would be $e^{4.56}/e^{0} = 95.56$ times higher if all pirate had scurvy than if no pirates had scurvy.
- In the second level of route danger ($V = 2$), the counterfactual odds of mortality would be $e^{2.60+6.38}/e^{2.60} = 589.93$ times higher if all pirate had scurvy than if no pirates had scurvy.
- In the highest level of route danger ($V = 3$), the counterfactual odds of mortality would be $e^{5.2+9.92}/e^{5.52} = 14,764.78$ times higher if all pirate had scurvy than if no pirates had scurvy.
Avast! Pirates need their vitamin C!

- If the MSM is considered a "working" model, then the choice of the numerator of the weights changes the projection. For example, if we set the numerator to be 1, then all areas of the causal curve are weighted equally. If, instead, we set the numerator to be the conditional probability of the exposure (scurvy), given the effect modifier (route danger) $g(A_i|V_i)$, then we would be giving more weight to areas with better support.

```
> # the true value of the MSM parameter with stabilized weights is calculated as follows
> # recall the Full data consists (W1, W2, A, Y, Y.1, Y.0) for 100,000 pirates
> #
> # calculate the numerator
> gAV.reg.true<- glm(A~ as.factor(W2), family="binomial", data=Full)
> pred.g1V.true<- predict(gAV.reg.true, type="response")
> pred.g0V.true<- 1- pred.g1V.true
> wt.st<- c(pred.g1V.true, pred.g0V.true) # corresponding to counterfactual exposure
> # weighted regression
> true.MSM.st<- glm(Y.A~ A*V, weights=wt.st ,family="binomial")
> true.MSM.st$coef
```

```
(Intercept)           A           V         A:V 
  -2.573335    2.593638    2.578216    1.961764 
```

5 Appendix B: Evaluating the performance of the IPTW estimators

```
> # number of iterations
> R<- 500
> # matrix for estimates from IPTW for the G-computation estimand
> estimates<- matrix(NA, ncol=3, nrow=R)
> # two matrices for the coef estimates from IPTW for MSM estimands
> est.msm<- est.msm.st<- matrix(NA, ncol=4, nrow=R)
> for(r in 1:R){
+   # 0. redraw the data
+   NewData<- genData(n)
+ 
+   # 1.  Estimate the treatment mechanism g(A|W)=P(A|W)
+   gAW.reg<- glm(A ~ W1 +W2, family="binomial", data=NewData)
+ 
+   # 2. # predicted probability of having scurvy, given the obs cov P(A=1|W) = g(1|W)
+   pred.g1W <- predict(gAW.reg, type= "response")
+   # predicted probability of not having scurvy, given the obs cov P(A=0|W)=g(0|W)
+   pred.g0W <- 1 - pred.g1W
+ 
+   # we need the predicted prob of the observed treatment (scurvy), given covariates.
+   # create an empty vector
+   gAW <- rep(NA, n)
+ 
+   # for pirates with scurvy, gAW = P(A=1 | W)
+   gAW[NewData$A==1] <- pred.g1W[NewData$A==1]
+   # for pirates without scurvy, gAW = P(A=0 | W)
+   gAW[NewData$A==0] <- pred.g0W[NewData$A==0]
+ 
+   # 3. Each subject gets a weight inverse weight inverse to pred prob
+   wt<- 1/gAW
+ 
+   # 4. # the weighted empirical mean outcome for pirates with scurvy
+   # minus the weighted empirical mean for pirates without scurvy
+   IPTW<- mean( wt*as.numeric(NewData$A==1)*NewData$Y) -
+    mean( wt*as.numeric(NewData$A==0)*NewData$Y)
+ 
+   # 6.  truncate  weights ARBITRARILY at 10
+   wt.trunc<- wt
+   wt.trunc[ wt.trunc>10] =10
+   # evaluate the IPTW estimand with the truncated weights
+   IPTW.tr<- mean( wt.trunc*as.numeric(NewData$A==1)*NewData$Y) -
+     mean( wt.trunc*as.numeric(NewData$A==0)*NewData$Y)
+ 
+   # 7. Stabilized IPTW estimator - Modified Horvitz-Thompson estimator
+   IPTW.HT<- mean( wt*as.numeric(NewData$A==1)*NewData$Y)/mean( wt*as.numeric(NewData$A==1)) -
+     mean( wt*as.numeric(NewData$A==0)*NewData$Y)/mean( wt*as.numeric(NewData$A==0))
+
+   estimates[r,]<-c(IPTW, IPTW.tr, IPTW.HT)
+
+   # NOW FOR MSM COEF
+   IPTW.msm<- glm(Y~A + W2 + A:W2, weights=wt, family='binomial', data=NewData)
+   est.msm[r, ]<- IPTW.msm$coef
+
+   # for MSM with stabilized weights
+   gAV.reg <- glm(A ~ as.factor(W2), family="binomial", data=NewData)
+   # predicting the probability of scurvy, given the route danger g(1|V)
+   pred.g1V<- predict(gAV.reg, type="response")
+   # predicted probability of no scurvy, given the route danger g(0|V)
+   pred.g0V<- 1- pred.g1V
+
+   # create vector for numerator
+   gAV<- rep(NA, n)
+   # for pirates with scurvy g1V = P(A=1|V)
+   gAV[NewData$A==1] <- pred.g1V[NewData$A==1]
+   # for pirates without scurvy g0V = P(A=0|V)
+   gAV[NewData$A==0] <- pred.g0V[NewData$A==0]
+
+   # creating the stabilized weights
+   wt.MSM<- gAV/gAW
+
+   # 4. estimate the parameters of the MSM with IPTW
+   IPTW.msm.st<- glm(Y~A + W2 + A:W2, weights=wt.MSM, family='binomial', data=NewData )
+   est.msm.st[r, ]<- IPTW.msm.st$coef
+ }
> colnames(estimates)<- c("IPTW", "IPTW.trunc", "IPTW.stab")
> colnames(est.msm) <- colnames(est.msm.st) <- c('beta0', 'beta1', 'beta2', 'beta3')
```

Performance of IPTW estimators of the G-Computation estimand

```
> # Average value of the estimates over R repetitions
> colMeans(estimates)
```

```
     IPTW IPTW.trunc  IPTW.stab
0.2603658  0.5355686  0.2718150
```

```
> Psi.F
```

```
[1] 0.26432
```

```
> # Bias: ave deviation from Psi.F and point estimates
> # recall that the treatment mechanism g0(A|W) was estimated with the correctly
> # specified parametric model
> colMeans(estimates - Psi.F)
```

```
        IPTW   IPTW.trunc    IPTW.stab
-0.003954195  0.271248560  0.007495004
```

```
> # Variance
> diag(var(estimates))


        IPTW   IPTW.trunc    IPTW.stab
0.0329454103 0.0001467801 0.0027522719
```

We see a huge reduction in variance from truncating or using stabilized weights However there is a corresponding increase in bias with truncation.

```
> # MSE
> colMeans( (estimates-Psi.F)^2)


       IPTW  IPTW.trunc   IPTW.stab
0.032895155 0.073722268 0.002802942


> # Overall lowest MSE was with IPTW using stabilized weights...
```

See Alex's presentation about why a small MSE is not necessarily the best metric if we are interested in inference (i.e. hypothesis testing and confidence interval construction)...

Performance of IPTW estimators of MSM parameters

```
> # ave coef estimate for IPTW without stabilized weights
> round(colMeans(est.msm), 3)


 beta0  beta1  beta2  beta3
-2.651  2.716  2.650  2.086


> # true values with numerator gAV=1
> round(true.MSM$coef, 3)


(Intercept)           A           V         A:V
     -2.596       2.617       2.602       1.937


> # bias in coef estimates for MSM without stabilized
> # m(a,V|Beta) = expit[ beta0 + beta1*a + beta2*V + beta3*a*V]
> colMeans(est.msm) - true.MSM$coef


      beta0       beta1       beta2       beta3
-0.05552913  0.09907288  0.04791903  0.14880728


> # variance
> diag( var(est.msm))


    beta0     beta1     beta2     beta3
0.0967366 0.4120130 0.1011498 1.5804522
```

```
> #~~~~~~~~~~~~~~~
> # ave coef estimate for IPTW with stabilized weights
> round(colMeans(est.msm.st), 3)

 beta0  beta1  beta2  beta3
-2.615  2.669  2.614  2.138

> # true values with numerator gAV= g_n(A|V)
> round(true.MSM.st$coef, 3)

(Intercept)           A           V         A:V
     -2.573       2.594       2.578       1.962

> # bias in coef estimates for MSM with stabilized weights
> # m(a,V|Beta) = expit[ beta0 + beta1*a + beta2*V + beta3*a*V]
> colMeans(est.msm.st) - true.MSM.st$coef

      beta0       beta1       beta2       beta3
-0.04197652  0.07568259  0.03564063  0.17643535

> # variance
> diag( var(est.msm.st))

     beta0      beta1      beta2      beta3
0.02844070 0.33488474 0.02921898 1.44231805

> # less variable
```