Causal Inference with Continuous Exposures: A Tutorial with Application to ICU Data: Real Data Analysis

M Ehsan Karim

0.1 Download and Prepare RHC Data

```r
# Download data
ObsData <- read.csv("https://hbiostat.org/data/repo/rhc.csv", header = TRUE)

# Calculate Length of Stay
ObsData$Length.of.Stay <- ObsData$dschdte - ObsData$sadmdte
ObsData$Length.of.Stay[is.na(ObsData$Length.of.Stay)] <-
  ObsData$dthdte[is.na(ObsData$Length.of.Stay)] -
  ObsData$sadmdte[is.na(ObsData$Length.of.Stay)]

# Binary outcome
ObsData$death <- ifelse(ObsData$death == "Yes", 1, 0)

# Remove unwanted outcome variables
ObsData <- dplyr::select(ObsData, !c(dthdte,
                                     lstctdte,
                                     dschdte,
                                     t3d30,
                                     dth30,
                                     surv2md1))

# Remove problematic variables
ObsData <- dplyr::select(ObsData, !c(sadmdte,
                                     ptid,
                                     X,
                                     adld3p,
                                     urin1,
                                     cat2))

# Convert categorical variables
factors <- c("cat1", "ca", "death", "cardiohx", "chfhx", "dementhx", "psychhx",
             "chrpulhx", "renalhx", "liverhx", "gibledhx", "malighx", "immunhx",
             "transhx", "amihx", "sex", "dnr1", "ninsclas", "resp", "card", "neuro",
             "gastr", "renal", "meta", "hema", "seps", "trauma", "ortho", "race",
             "income")
ObsData[factors] <- lapply(ObsData[factors], as.factor)

# Recode RHC use
ObsData$RHC.use <- ifelse(ObsData$swang1 == "RHC", 1, 0)
ObsData <- dplyr::select(ObsData, -swang1)
```

```r
# Recode and factor levels
ObsData$age <- cut(ObsData$age, breaks=c(-Inf, 50, 60, 70, 80, Inf), right=FALSE)
ObsData$race <- factor(ObsData$race, levels=c("white", "black", "other"))
ObsData$sex <- relevel(as.factor(ObsData$sex), ref = "Male")
ObsData$cat1 <- factor(ObsData$cat1, levels = unique(ObsData$cat1))
levels(ObsData$cat1) <- c("ARF", "CHF", "Other", "Other", "Other", "Other", "Other", "MOSF", "MOSF")
ObsData$ca <- factor(ObsData$ca, levels = c("No", "Yes"), labels = c("None", "Metastatic"))

# Rename variables
names(ObsData) <- c("Disease.category", "Cancer", "Death", "Cardiovascular", "Congestive.HF",
                    "Dementia", "Psychiatric", "Pulmonary", "Renal", "Hepatic", "GI.Bleed",
                    "Tumor", "Immunosupperssion", "Transfer.hx", "MI", "age", "sex", "edu",
                    "DASIndex", "APACHE.score", "Glasgow.Coma.Score", "blood.pressure", "WBC",
                    "Heart.rate", "Respiratory.rate", "Temperature", "PaO2vs.FIO2", "Albumin",
                    "Hematocrit", "Bilirubin", "Creatinine", "Sodium", "Potassium", "PaCo2",
                    "PH", "Weight", "DNR.status", "Medical.insurance", "Respiratory.Diag",
                    "Cardiovascular.Diag", "Neurological.Diag", "Gastrointestinal.Diag",
                    "Renal.Diag", "Metabolic.Diag", "Hematologic.Diag", "Sepsis.Diag",
                    "Trauma.Diag", "Orthopedic.Diag", "race", "income", "Length.of.Stay",
                    "RHC.use")

str(ObsData)
```

```
## 'data.frame':    5735 obs. of  52 variables:
##  $ Disease.category     : Factor w/ 4 levels "ARF","CHF","Other",..: 1 2 3 3 2 1 3 3 3 3 ...
##  $ Cancer               : Factor w/ 2 levels "None","Metastatic": 2 1 2 1 1 1 NA 1 2 2 ...
##  $ Death                : Factor w/ 2 levels "0","1": 1 2 1 2 2 1 1 2 1 1 ...
##  $ Cardiovascular       : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 1 1 1 ...
##  $ Congestive.HF        : Factor w/ 2 levels "0","1": 1 2 1 1 1 2 1 1 1 1 ...
##  $ Dementia             : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Psychiatric          : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Pulmonary            : Factor w/ 2 levels "0","1": 2 1 1 1 1 2 1 1 1 1 ...
##  $ Renal                : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Hepatic              : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ GI.Bleed             : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Tumor                : Factor w/ 2 levels "0","1": 2 1 2 1 1 1 2 1 1 2 ...
##  $ Immunosupperssion    : Factor w/ 2 levels "0","1": 1 2 2 2 1 1 1 1 1 1 ...
##  $ Transfer.hx          : Factor w/ 2 levels "0","1": 1 2 1 1 1 1 1 2 1 1 ...
##  $ MI                   : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
##  $ age                  : Factor w/ 5 levels "[-Inf,50)","[50,60)",..: 4 4 1 4 3 5 2 1 1 1 ...
##  $ sex                  : Factor w/ 2 levels "Male","Female": 1 2 2 2 1 2 1 1 2 2 ...
##  $ edu                  : num  12 12 14.07 9 9.95 ...
##  $ DASIndex             : num  23.5 14.8 18.1 22.9 21.1 ...
##  $ APACHE.score         : int  46 50 82 48 72 38 29 25 47 48 ...
##  $ Glasgow.Coma.Score   : int  0 0 0 0 41 0 26 100 0 0 ...
##  $ blood.pressure       : num  41 63 57 55 65 115 67 128 53 73 ...
##  $ WBC                  : num  22.1 28.9 0.05 23.3 29.7 ...
##  $ Heart.rate           : int  124 137 130 58 125 134 135 102 118 141 ...
##  $ Respiratory.rate     : num  10 38 40 26 27 36 10 34 30 40 ...
##  $ Temperature          : num  38.7 38.9 36.4 35.8 34.8 ...
##  $ PaO2vs.FIO2          : num  68 218 276 157 478 ...
##  $ Albumin              : num  3.5 2.6 3.5 3.5 3.5 ...
##  $ Hematocrit           : num  58 32.5 21.1 26.3 24 ...
##  $ Bilirubin            : num  1.01 0.7 1.01 0.4 1.01 ...
##  $ Creatinine           : num  1.2 0.6 2.6 1.7 3.6 ...
##  $ Sodium               : int  145 137 146 117 126 138 136 136 136 146 ...
##  $ Potassium            : num  4 3.3 2.9 5.8 5.8 ...
##  $ PaCo2                : num  40 34 16 30 17 68 45 26 40 30 ...
##  $ PH                   : num  7.36 7.33 7.36 7.46 7.23 ...
##  $ Weight               : num  64.7 45.7 0 54.6 78.4 ...
##  $ DNR.status           : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 1 1 1 1 1 ...
##  $ Medical.insurance    : Factor w/ 6 levels "Medicaid","Medicare",..: 2 6 5 6 2 2 5 5 5 1 ...
##  $ Respiratory.Diag     : Factor w/ 2 levels "No","Yes": 2 1 1 2 1 2 1 2 1 1 ...
##  $ Cardiovascular.Diag  : Factor w/ 2 levels "No","Yes": 2 1 2 1 2 1 1 1 1 1 ...
##  $ Neurological.Diag    : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 2 1 2 ...
##  $ Gastrointestinal.Diag: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 2 ...
##  $ Renal.Diag           : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 1 1 1 ...
##  $ Metabolic.Diag       : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Hematologic.Diag     : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 2 1 ...
##  $ Sepsis.Diag          : Factor w/ 2 levels "No","Yes": 1 2 1 1 1 1 1 2 1 1 ...
##  $ Trauma.Diag          : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ Orthopedic.Diag      : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...
##  $ race                 : Factor w/ 3 levels "white","black",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ income               : Factor w/ 4 levels "$11-$25k","$25-$50k",..: 4 4 2 1 4 4 2 2 4 4 ...
##  $ Length.of.Stay       : int  9 45 60 37 2 7 42 34 11 19 ...
##  $ RHC.use              : num  0 1 1 0 1 0 0 0 0 1 ...
```

```r
# Save cleaned dataset
saveRDS(ObsData, file = "rhcAnalytic.RDS")
```

0.2 Load RHC Data

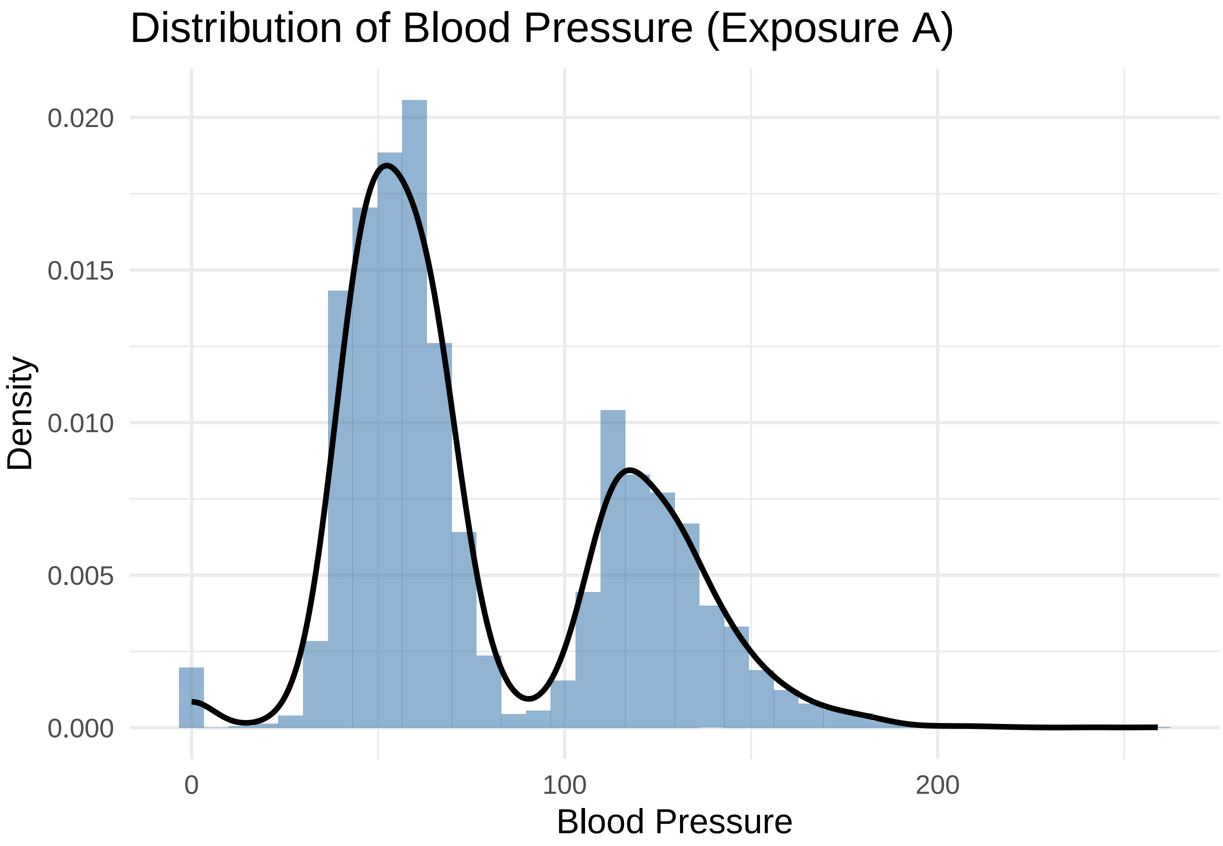
```r
rhc <- readRDS("rhcAnalytic.RDS")

# Define comprehensive confounder set
covariates <- c(
  "age", "sex", "Cardiovascular", "Pulmonary",
  "Renal", "Congestive.HF", "Cancer",
  "APACHE.score", "DASIndex", "Albumin",
  "Creatinine", "Sodium", "Heart.rate", "WBC",
  "DNR.status", "Transfer.hx"
)

# Prepare variables
rhc <- rhc %>%
  mutate(
    A = blood.pressure,
    Y = as.numeric(as.character(Death))
  ) %>%
  filter(!is.na(A), !is.na(Y)) %>%
  filter(complete.cases(select(., all_of(covariates))))
```

0.3 Summarize the Exposure

```
ggplot(rhc, aes(x = A)) +
  geom_histogram(aes(y = ..density..), bins = 40, fill = "steelblue", alpha = 0.6) +
  geom_density(color = "black", linewidth = 1) +
  labs(
    title = "Distribution of Blood Pressure (Exposure A)",
    x = "Blood Pressure",
    y = "Density"
  ) +
  theme_minimal(base_size = 13)
```



0.4 Method 1: IPW with Normal Exposure Model

```
# Numerator: marginal exposure model
mod_num <- lm(A ~ 1, data = rhc)
mu_num <- predict(mod_num)
sd_num <- sd(residuals(mod_num))

# Denominator: exposure conditional on confounders
form_denom <- as.formula(paste("A ~",
                               paste(covariates, collapse = " + ")))
```

```
mod_denom <- lm(form_denom, data = rhc)
mu_denom <- predict(mod_denom)
sd_denom <- sd(residuals(mod_denom))

# Densities and weights
f_num <- dnorm(rhc$A, mean = mu_num, sd = sd_num)
f_denom <- dnorm(rhc$A, mean = mu_denom, sd = sd_denom)
rhc$sw_normal <- f_num / f_denom

# Weighted model
mod_w_normal <- glm(Y ~ A, family = binomial(),
                    data = rhc, weights = sw_normal)
publish(mod_w_normal)
```

```
##  Variable Units OddsRatio       CI.95 p-value
##         A            1.00 [1.00;1.00] < 1e-04
```

0.5 Method 2: IPW with Quantile Binning

```
# Create quantile bins
rhc$qbin <- cut(rhc$A, breaks = quantile(rhc$A,
                                         probs = seq(0, 1, 0.1),
                                         na.rm = TRUE),
                include.lowest = TRUE, labels = FALSE)

# Multinomial model for conditional bin probability
form_q <- as.formula(paste("factor(qbin) ~",
                           paste(covariates, collapse = " + ")))
mod_q <- multinom(form_q, data = rhc, trace = FALSE)
p_denom <- predict(mod_q, type = "probs")
row_idx <- cbind(1:nrow(rhc), rhc$qbin)
p_denom_val <- p_denom[row_idx]
p_num_val <- 1 / 10
rhc$sw_qbin <- p_num_val / p_denom_val

# Weighted model
mod_w_qbin <- glm(Y ~ A, family = binomial(),
                  data = rhc, weights = sw_qbin)
publish(mod_w_qbin)
```

```
##  Variable Units OddsRatio       CI.95  p-value
##         A            1.00 [1.00;1.00]   0.2348
```

0.6 Method 3: TMLE with Shift Intervention

```
# TMLE setup
node_list <- list(W = covariates, A = "A", Y = "Y")
glm_learner <- make_learner(Lrnr_glm)
learner_list <- list(Y = glm_learner, A = glm_learner)
```

```r
tmle_spec <- tmle_shift(
  delta = 0.1,
  shift_fxn = function(tmle_task, delta, ...) {
    a <- tmle_task$get_tmle_node("A")
    a + delta
  },
  max_shift = 1
)

# Run TMLE
future::plan(future::sequential)
tmle_fit <- tmle3(
  tmle_spec,
  data = as.data.table(rhc),
  node_list = node_list,
  learner_list = learner_list
)

# Extract estimate
psi <- tmle_fit$estimates[[1]]$psi
IC <- tmle_fit$estimates[[1]]$IC
se <- sd(IC) / sqrt(length(IC))
ci <- psi + c(-1.96, 1.96) * se

# Output
data.frame(
  logOR = psi,
  OR = exp(psi),
  se = se,
  lower = exp(ci[1]),
  upper = exp(ci[2])
)
```

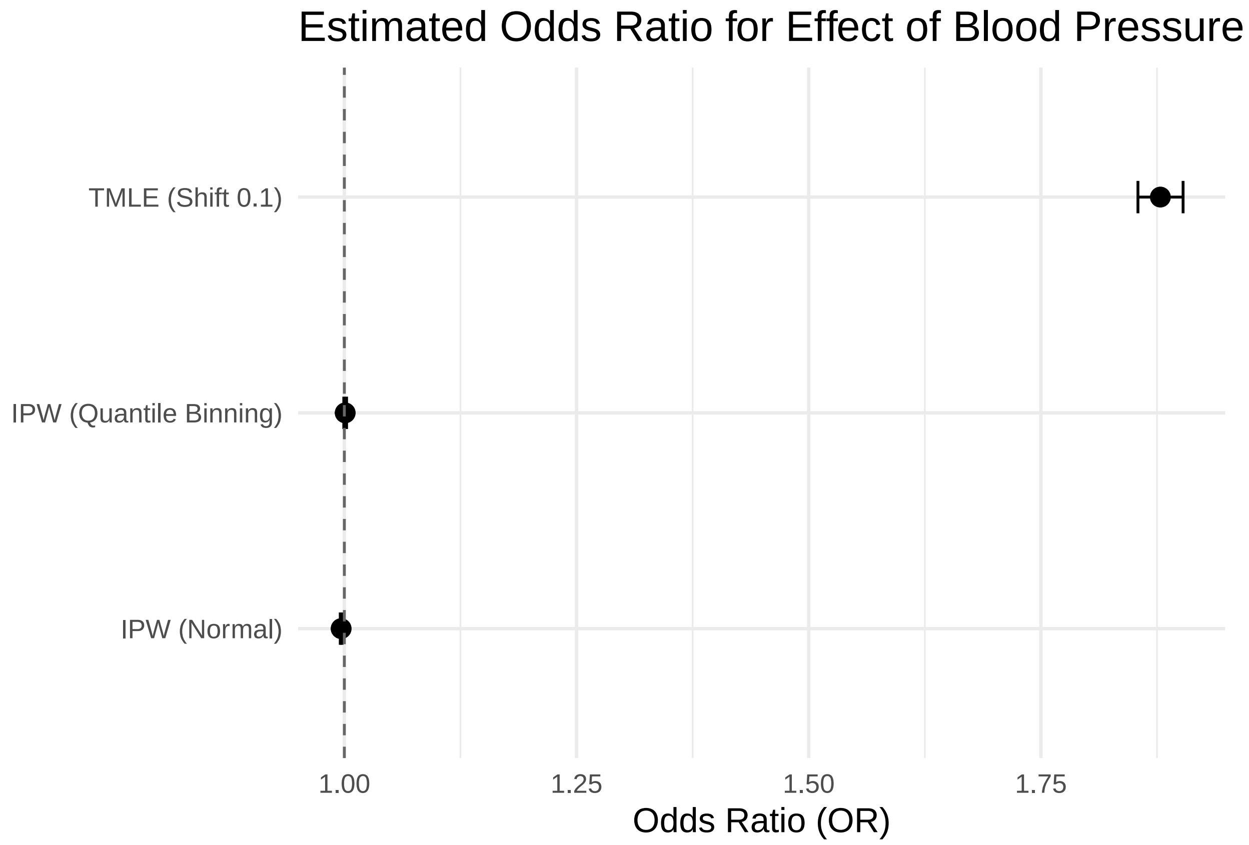
```
##       logOR       OR          se    lower    upper
## 1 0.6305272 1.878601 0.00659851 1.854461 1.903055
```

0.7 Results

```r
# Manually collect results into a data frame
estimates <- data.frame(
  Method = c("IPW (Normal)", "IPW (Quantile Binning)", "TMLE (Shift 0.1)"),
  logOR = c(
    coef(mod_w_normal)["A"],
    coef(mod_w_qbin)["A"],
    psi
  ),
  SE = c(
    sqrt(vcov(mod_w_normal)["A", "A"]),
    sqrt(vcov(mod_w_qbin)["A", "A"]),
    sd(IC) / sqrt(length(IC))
  )
)

# Compute OR and 95% CI
estimates <- estimates %>%
  mutate(
    OR = exp(logOR),
    lower = exp(logOR - 1.96 * SE),
    upper = exp(logOR + 1.96 * SE)
  )

# Plot
ggplot(estimates, aes(x = Method, y = OR)) +
  geom_point(size = 3) +
  geom_errorbar(aes(ymin = lower, ymax = upper), width = 0.15) +
  geom_hline(yintercept = 1, linetype = "dashed", color = "gray40") +
  coord_flip() +
  labs(
    title = "Estimated Odds Ratio for Effect of Blood Pressure on Death",
    y = "Odds Ratio (OR)",
    x = NULL
  ) +
  theme_minimal(base_size = 13)
```



```r
library(knitr)
library(kableExtra)
```

```
# Prepare table of OR estimates
results_table <- estimates %>%
  transmute(
    Method,
    OR = sprintf("%.3f", OR),
    `95% CI` = sprintf("%.3f - %.3f", lower, upper),
    SE = sprintf("%.3f", SE)
  )

# Show as styled table
kable(results_table, format = "latex", booktabs = TRUE, caption = "Estimated Odds Ratios and 95% Confid
```

\begin{table}

\caption{Estimated Odds Ratios and 95% Confidence Intervals}

Method	OR	95% CI	SE
IPW (Normal)	0.997	0.996 – 0.997	0.000
IPW (Quantile Binning)	1.001	0.999 – 1.002	0.001
TMLE (Shift 0.1)	1.879	1.854 – 1.903	0.007

\end{table}

0.8 Save plot

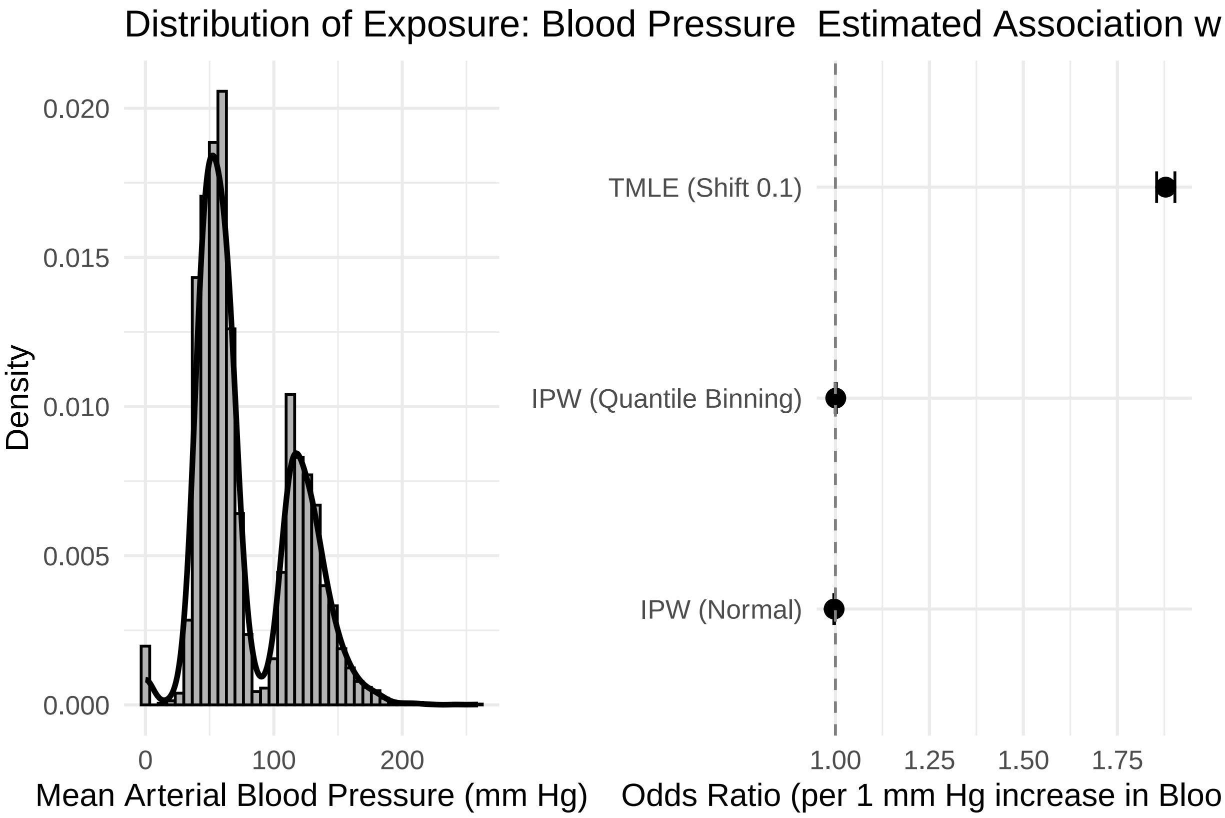
```
# Load required libraries
library(ggplot2)
library(patchwork)

# Plot 1: Distribution of the exposure A (Blood Pressure)
p1 <- ggplot(rhc, aes(x = A)) +
  geom_histogram(aes(y = ..density..), bins = 40, fill = "grey70", color = "black") +
  geom_density(color = "black", linewidth = 1) +
  labs(
    title = "Distribution of Exposure: Blood Pressure",
    x = "Mean Arterial Blood Pressure (mm Hg)",
    y = "Density"
  ) +
  theme_minimal(base_size = 12)

# Plot 2: Forest plot of estimated odds ratios for death
p2 <- ggplot(estimates, aes(x = Method, y = OR)) +
  geom_point(size = 3, color = "black") +
  geom_errorbar(aes(ymin = lower, ymax = upper), width = 0.15, color = "black") +
  geom_hline(yintercept = 1, linetype = "dashed", color = "grey50") +
  coord_flip() +
  labs(
    title = "Estimated Association with In-Hospital Death",
    y = "Odds Ratio (per 1 mm Hg increase in Blood Pressure)",
    x = NULL
  ) +
  theme_minimal(base_size = 12)

# Combine plots side by side
combined_plot <- p1 + p2 + plot_layout(ncol = 2)

# Display
print(combined_plot)
```



```
# Optional: Save the plot
ggsave("rhc_bw_combined_plot.png", combined_plot, width = 12, height = 5, dpi = 600)
```

0.9 Diagnostic Checks

To assess the reliability of causal estimates, we conduct several diagnostic checks related to the validity of assumptions and model performance. These include checking weight stability, covariate balance, and the positivity assumption.

0.9.1 Weight Diagnostics

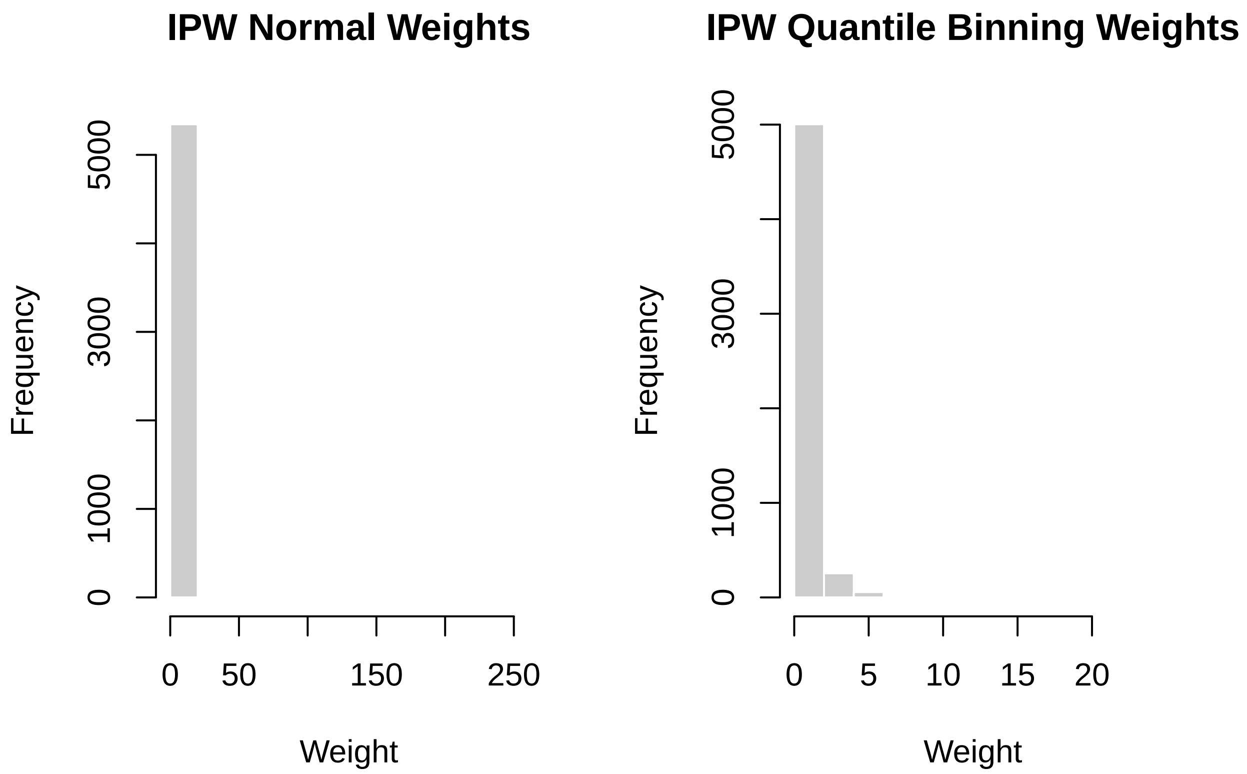
```
# Histograms of weights
summary(rhc$sw_normal)
```

```
##     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
##   0.1041    0.7208    0.8650    1.0907    1.0737  243.6645
```

```
summary(rhc$sw_qbin)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1136  0.5673  0.7700  1.0083  1.0925 22.7556
```

```
par(mfrow = c(1, 2))
hist(rhc$sw_normal, main = "IPW Normal Weights", xlab = "Weight", col = "grey80", border = "white")
hist(rhc$sw_qbin, main = "IPW Quantile Binning Weights", xlab = "Weight", col = "grey80", border = "whi
```



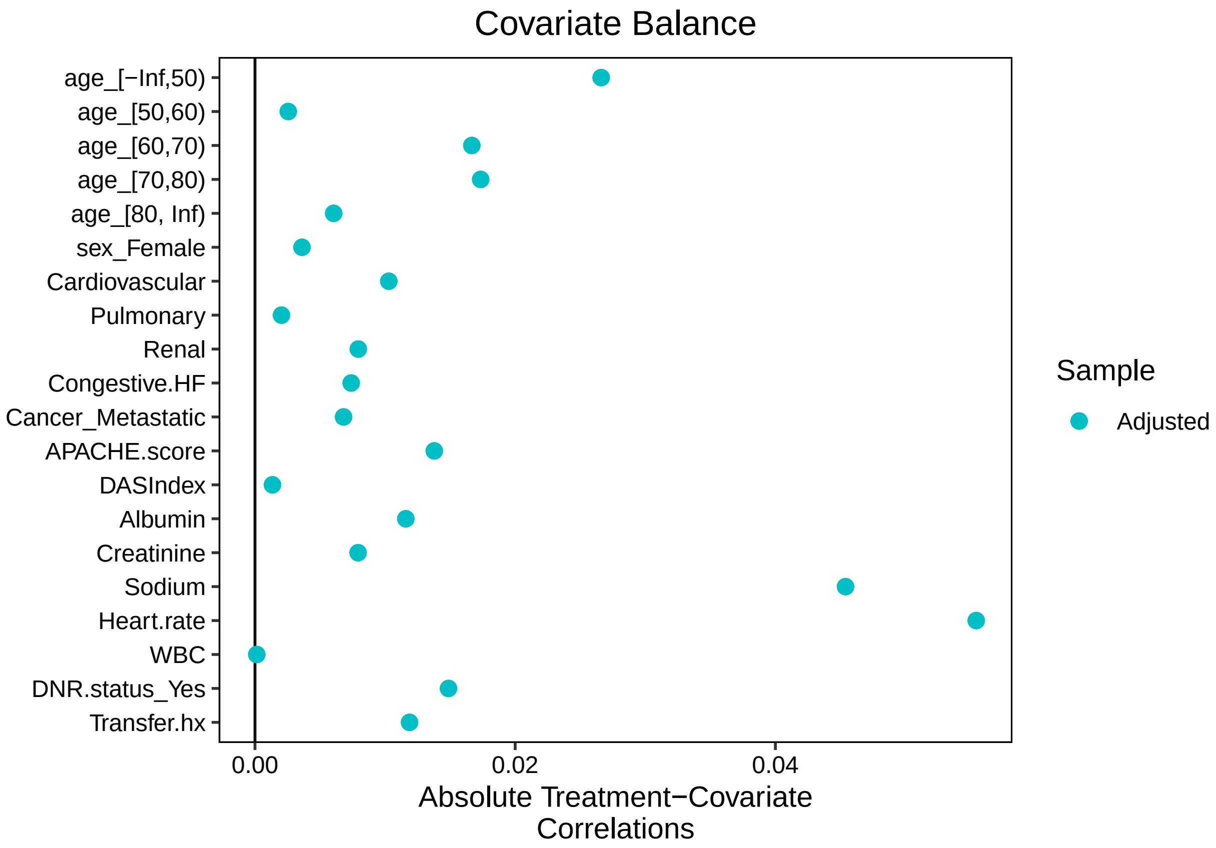
Extreme or highly variable weights can indicate misspecification in the exposure model or violations of the positivity assumption.

0.9.2 Covariate Balance with IPW

```
# Covariate balance before and after IPW using quantile binning
bal <- bal.tab(
  as.formula(paste("A ~", paste(covariates, collapse = " + "))),
  data = rhc,
  weights = rhc$sw_qbin,
  method = "weighting"
)
bal
```

```
## Balance Measures
##                        Type Corr.Adj
## age_[-Inf,50)        Binary  -0.0266
## age_[50,60)          Binary  -0.0026
## age_[60,70)          Binary   0.0167
## age_[70,80)          Binary   0.0173
## age_[80, Inf)        Binary  -0.0060
## sex_Female           Binary   0.0036
## Cardiovascular       Binary   0.0103
## Pulmonary            Binary   0.0020
## Renal                Binary   0.0079
## Congestive.HF        Binary  -0.0074
## Cancer_Metastatic    Binary   0.0068
## APACHE.score        Contin.  -0.0138
## DASIndex            Contin.   0.0013
## Albumin             Contin.   0.0116
## Creatinine          Contin.   0.0079
## Sodium              Contin.  -0.0454
## Heart.rate          Contin.  -0.0554
## WBC                 Contin.   0.0001
## DNR.status_Yes       Binary   0.0149
## Transfer.hx          Binary  -0.0119
##
## Effective sample sizes
##              Total
## Unadjusted 5351.
## Adjusted   2382.93
```

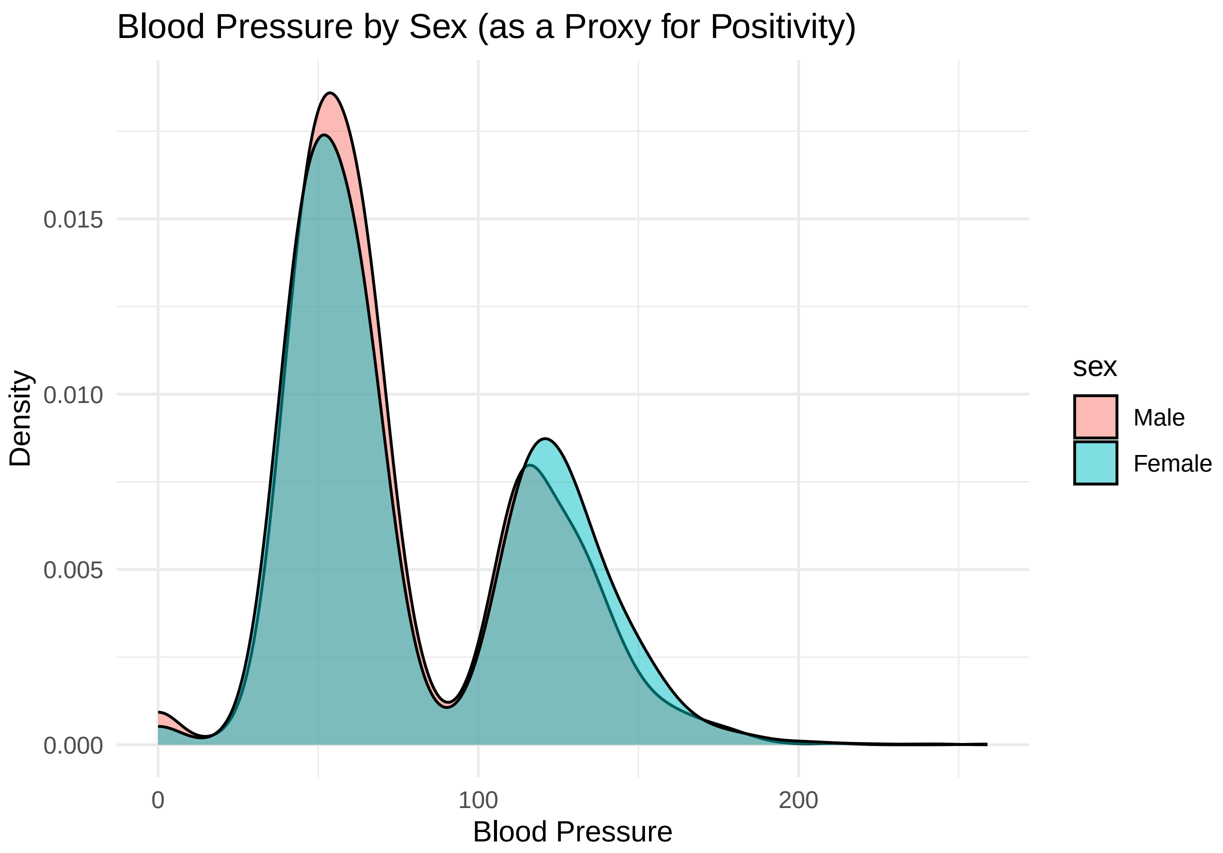
```
love.plot(bal, abs = TRUE, thresholds = c(m = 0.1))
```



The love plot shows performance before and after weighting. Good balance is typically indicated by values below 0.1.

0.9.3 Positivity Check

```
# Exposure distribution by confounder category
ggplot(rhc, aes(x = A, fill = sex)) +
  geom_density(alpha = 0.5) +
  labs(title = "Blood Pressure by Sex (as a Proxy for Positivity)", x = "Blood Pressure", y = "Density"
  theme_minimal()
```



This density plot illustrates whether the exposure is sufficiently variable across levels of key confounders. Lack of overlap may indicate a positivity violation.