ProblemSet1

Siyu Zou

2024-04-11

```
library(tidyverse)
library(mvtnorm)
library(readr)
library(mice)
library(dplyr)
library(knitr)
library(biostat3)
library(ggplot2)
library(lmtest)
```

I. Evaluating the impact of different missing data mechanisms within a randomized trial

Q1

1. Write an R function to simulate a randomized trial of patients from the population defined above. Set the inputs to the R function as: n (the total sample size of the trial), and delta (the treatment effect, i.e. the improvement in the QOL scores when the patient's surgery is performed under the novel protocol vs. standard of care). An outline of the R function is given below:

```
expit <- function(x) {
  exp(x) / (1 + exp(x))
  }

my.sim <- function(n , delta ){
  sigma <- matrix(c(16,5,5,6), ncol=2)
  data <- rmvnorm(n, mean=c(65,10), sigma=sigma)
  e <- rnorm(n, mean = 0, sd = 3)
  age <- data[ , 1]
  severity <- data[ , 2]
  Y0 <- 75 - 0.35 * (age -65) - 0.15 * (severity -10) + e
  Y1 <- Y0 + delta
  # c.  Randomly assign patients to receive the standard of care or novel protocol.  HINT:  Define the
  A = rbinom(n,1,prob=0.5)
  # d.  Define the outcome as Y = A * Y1 + (1-A) * Y0
  Y =  A * Y1 + (1-A) * Y0
  # e.  Estimate the marginal treatment effect under no missing data; trteffect0 = lm(Y~A)$coeff[2]
  trteffect0 = lm(Y~A)$coeff[2]

  # f.  For values j = 1, 2, 3, 4, generate R_j, the indicator that the patient did not return for thei
    R_1 = rbinom(n,size=1,prob= 0.15)
    x2 = -1.75 + 0.5 * (age-65) + 0.02 * (severity - 10)
    R_2 = rbinom(n,size=1,prob= expit(x2))
    x3 = -1.75 + 0.5 * (1-A) * (age-65) + 0.01 * A * (age - 65) + 0.02 * (severity - 10)
    R_3 = rbinom(n,size=1,prob= expit(x3))
    x4 = -1.75 + 0.35 * (1-A) * (Y-75)
    R_4 = rbinom(n,size=1,prob= expit(x4))

    # g.    Create Y_j = Y with values set to NA for patients with R_j = 1
    Y_1 = ifelse(R_1 == 1, NA, Y)
    Y_2 = ifelse(R_2 == 1, NA, Y)
    Y_3 = ifelse(R_3 == 1, NA, Y)
    Y_4 = ifelse(R_4 == 1, NA, Y)
    # h.    Estimate the marginal treatment effect under the four missing data models, e.g. trteffect1
    trteffect1 = lm(Y_1 ~ A,na.action=na.omit)$coeff[2]
    trteffect2 = lm(Y_2 ~ A,na.action=na.omit)$coeff[2]
    trteffect3 = lm(Y_3 ~ A,na.action=na.omit)$coeff[2]
    trteffect4 = lm(Y_4 ~ A,na.action=na.omit)$coeff[2]
    # i.    Have the function output trteffect0, trteffect1, ..., trteffect4
    return(c(trteffect0, trteffect1, trteffect2, trteffect3, trteffect4))
}
```

Q2

2. Using the function you created above, perform a simulation study: simulate 5,000 randomized trials of 250 patients assuming a marginal treatment effect of 3. Separately for each of the missing data conditions (i.e. no missing data and models 1 through 4), estimate the bias (sample mean of the 5,000 estimates minus delta), variance (sample variance of the 5,000 estimates) and mean squared error (bias^2 + variance) of the marginal treatment effects. Compute the relative mean squared error comparing the estimator under no missing data (numerator) to the estimators generated under the missing data models (denominator).

```
# Simulation parameters
K <- 5000
n_patients <- 250
delta <- 3

# Simulation results storage
results <- matrix(NA, ncol=5, nrow=K)

# Running the simulation
set.seed(123)
for (i in 1:K) {
  results[i, ] <- my.sim(n_patients, delta)
}

# Calculate Bias, Variance, and MSE
estimated_means <- colMeans(results)
biases <- estimated_means - delta
variances <- apply(results, 2, var)
mses <- biases^2 + variances
# Calculate relative MSE
relative_mse <- mses[1] / mses
```

```
results <- data.frame(
"Missing Data Conditions" = c("No Missing Data", "Model 1", "Model 2", "Model 3", "Model4"),
"Means" = estimated_means,
"Bias" = biases,
"Variance" = variances,
"Mean Squared Error" = mses,
"Relative Mean Squared Error" = relative_mse
)
results
```

```
##   Missing.Data.Conditions    Means         Bias  Variance Mean.Squared.Error
## 1         No Missing Data 2.995296 -0.004703568 0.1847466          0.1847687
## 2                 Model 1 2.992902 -0.007098171 0.2191746          0.2192249
## 3                 Model 2 2.996725 -0.003274814 0.2358724          0.2358831
## 4                 Model 3 2.495203 -0.504796693 0.2207403          0.4755600
## 5                  Model4 3.661379  0.661379326 0.2068482          0.6442708
##   Relative.Mean.Squared.Error
## 1                   1.0000000
## 2                   0.8428271
## 3                   0.7833063
## 4                   0.3885287
## 5                   0.2867873
```

Summarize your findings in 1 or 2 paragraphs.

For each of the missing data conditions, we estimate the bias (sample mean of the 5,000 estimates minus delta), mean squared error, variance, and mean squared error of the marginal treatment effects. We could see the smallest bias, variance and mean squared error when there is no missing data. Then for the other four missing data conditions that the patient did not return for their 1-month assessment, the bias and mean squared error were lower in model 1 and the highest in model 4. Model 1 means the probability that patient did not return is 0.15 and the cause of missingness is unrelated to the observed data. Model 4 is a condition that the probabilities of patient did not return for their 1-month assessment dependent on both the treatment A and outcome Y. Missing data under this condition may be more likely due to the side-effect of treatment, lower the accuracy of the estimate of marginal treatment effect. For the relative mean squared error (RMSE) comparing the estimator under no missing data (numerator) to the estimators generated under the missing data models (denominator), we could see the model 4 has the lowest RMSE. The low RMSE imply the estimate is biased. This simulation study tells us the importance of considering the mechanisms of missing data.

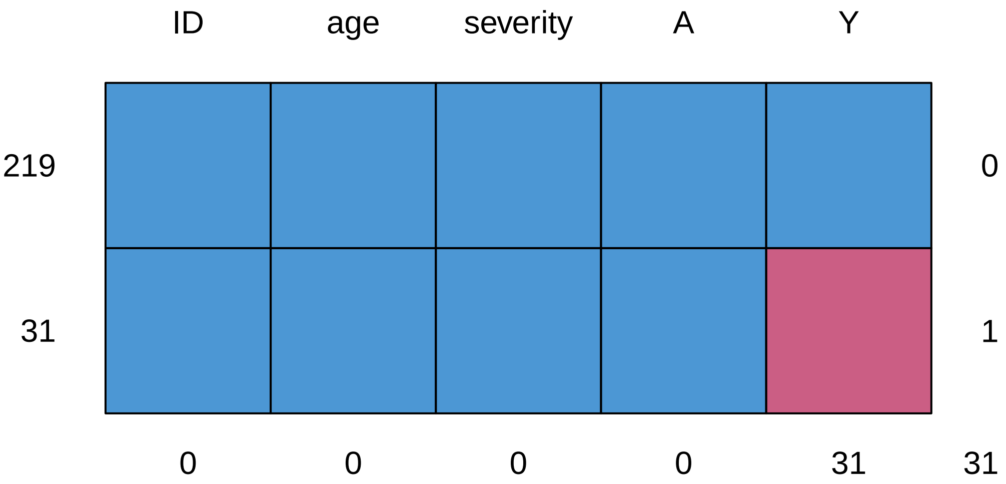
II. Analyze a dataset with missing data

1. Conduct an exploratory analysis of the missing data in your trial. Compute the proportion of missing outcomes (overall and within treatment group).

```
data <- read_csv("/Users/zousiyu/Library/CloudStorage/OneDrive-JohnsHopkins/Term 4/bios_654/Homework/Pr
```

```
## Rows: 250 Columns: 5
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## dbl (5): ID, age, severity, A, Y
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
md.pattern(data)
```



```
##     ID age severity A  Y
## 219  1   1        1 1  1  0
## 31   1   1        1 1  0  1
##      0   0        0 0 31 31
```

```
prop_missing_overall <-  sum(is.na(data$Y)) /  nrow(data)
print(prop_missing_overall)
```

```
## [1] 0.124
```

```
data %>%
  group_by(A) %>%
  summarise(
    Prevalence = sum(is.na(Y)) / n() )
```

```
## # A tibble: 2 x 2
##       A Prevalence
##   <dbl>      <dbl>
## 1     0     0.173
## 2     1     0.0732
```

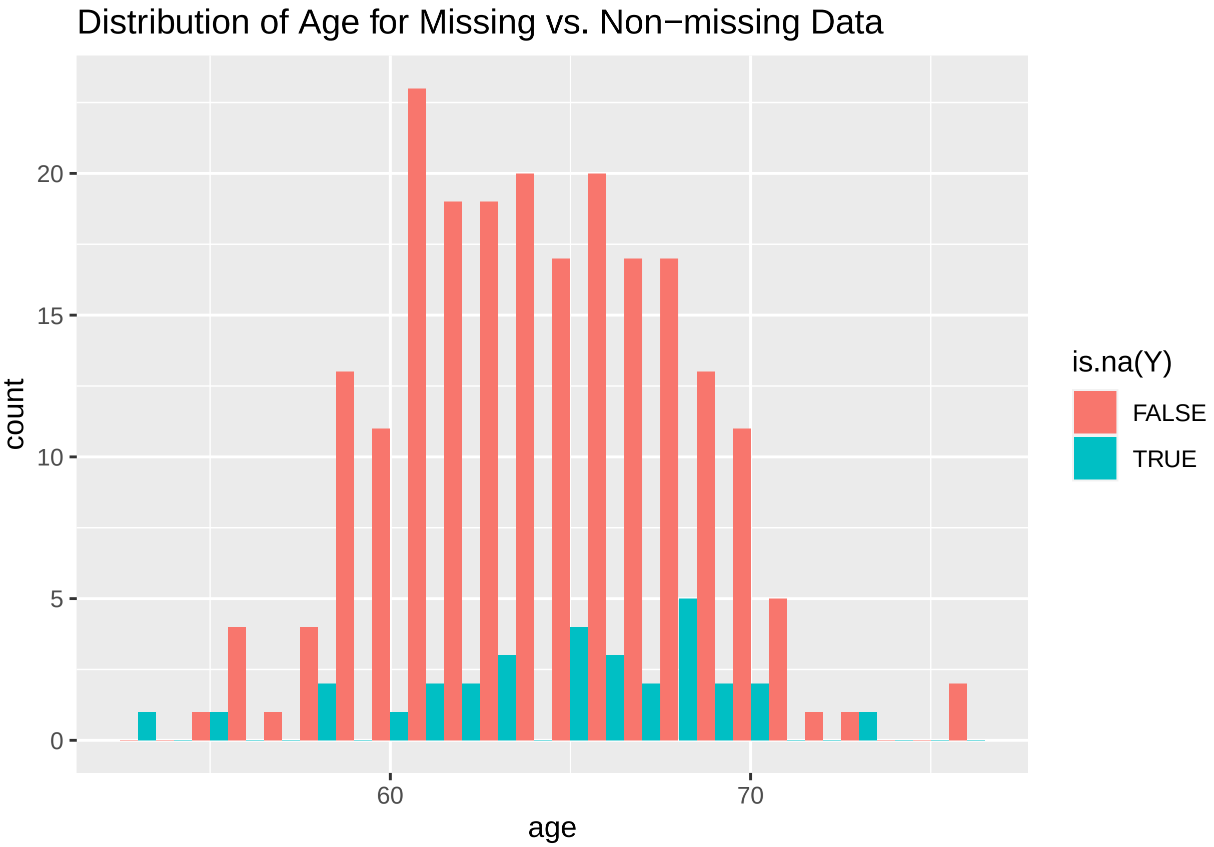
The proportion of missing outcomes is 0.124 overall, 0.173 for standard of care, 0.073 for the treatment group (surgery)

Look at the distribution of age and severity for those with and without missing data (overall and within treatment group).

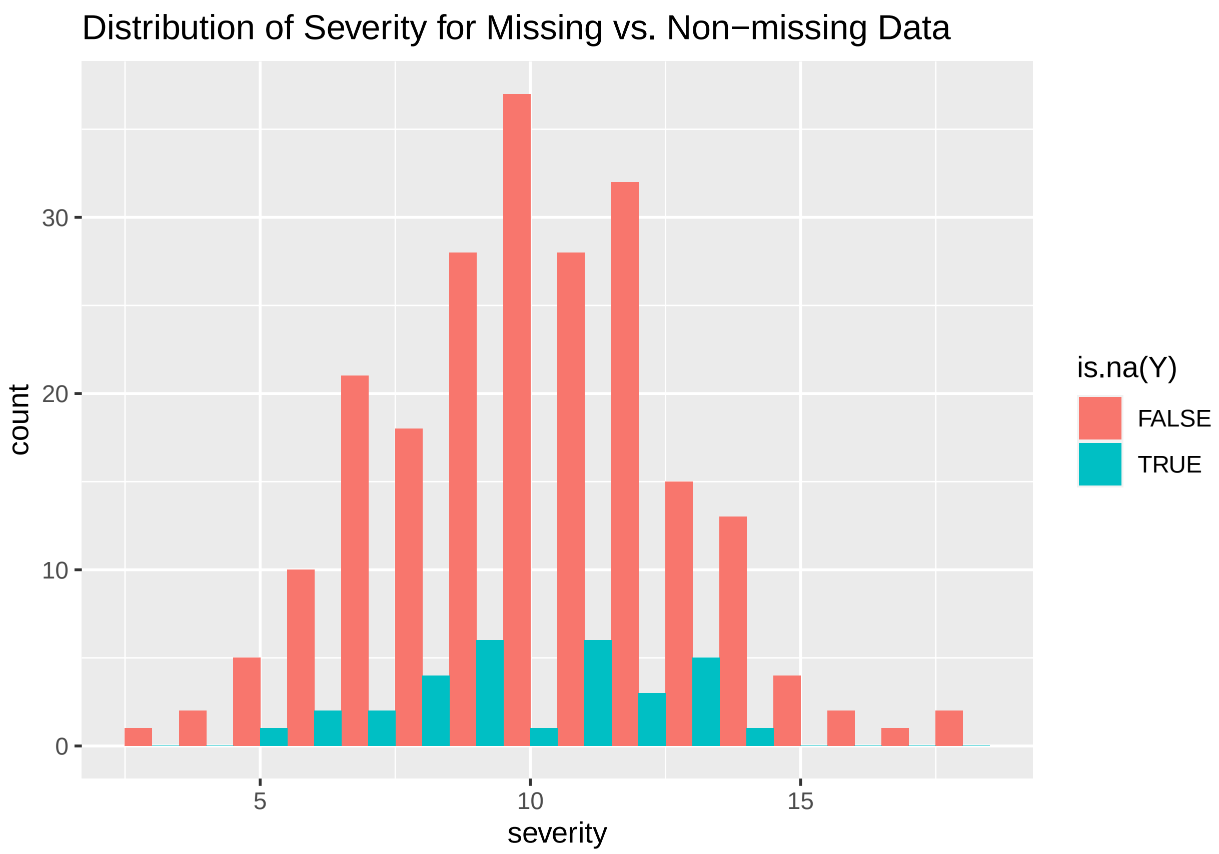
```
# Overall
updata <- data |>
  mutate(
    Y_missing = ifelse(is.na(Y) == T, 1, 0)
  )

updata |>
  group_by(Y_missing) |>
  summarise(Age_mean = mean(age),
            Age_sd = sd(age),
            Severity_mean = mean(severity),
            Severity_sd = sd(severity))
```

```
## # A tibble: 2 x 5
##   Y_missing Age_mean Age_sd Severity_mean Severity_sd
##       <dbl>    <dbl>  <dbl>         <dbl>       <dbl>
## 1         0     64.3   3.84         10.1         2.63
## 2         1     64.6   4.65          9.97        2.43
```

```
ggplot(data, aes(x = age, fill = is.na(Y))) +
  geom_histogram(binwidth = 1, position = "dodge") +
  labs(title = "Distribution of Age for Missing vs. Non-missing Data")
```



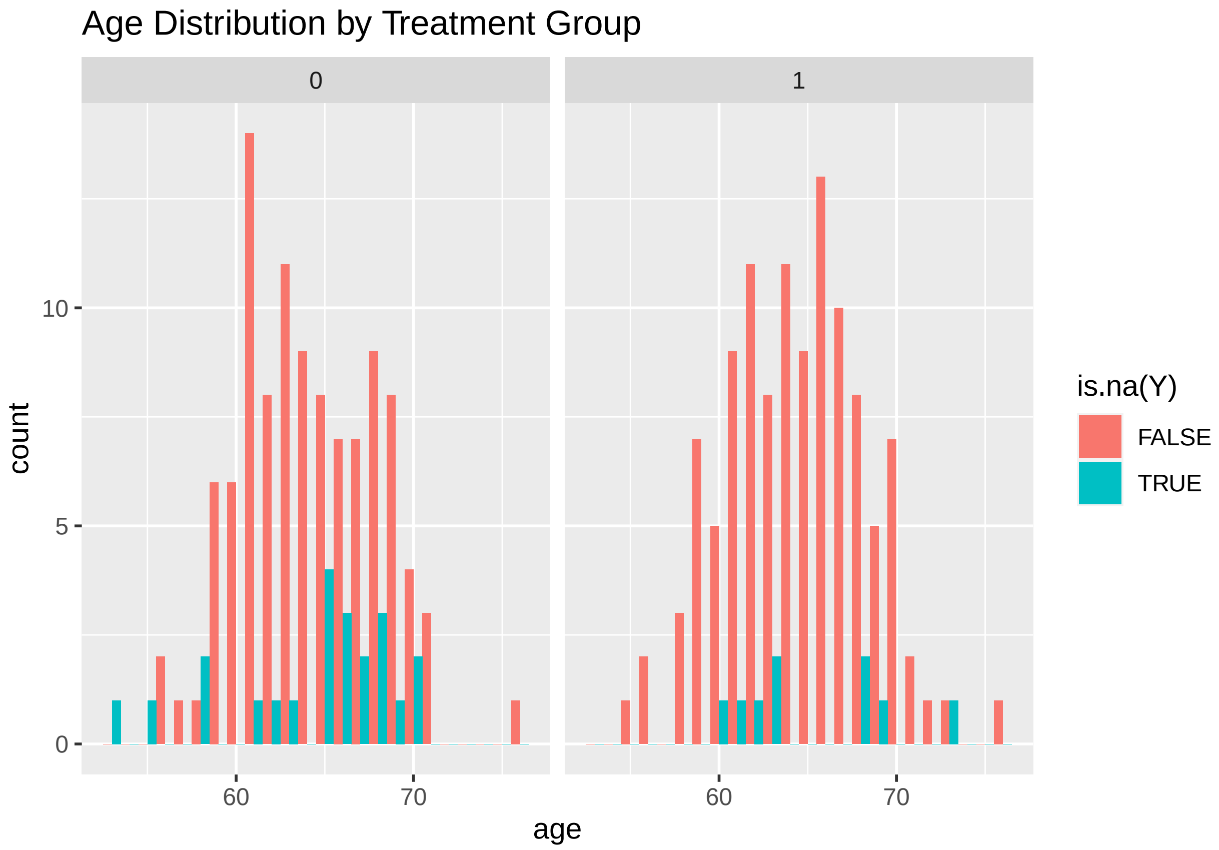
```
ggplot(data, aes(x = severity, fill = is.na(Y))) +
  geom_histogram(binwidth = 1, position = "dodge") +
  labs(title = "Distribution of Severity for Missing vs. Non-missing Data")
```



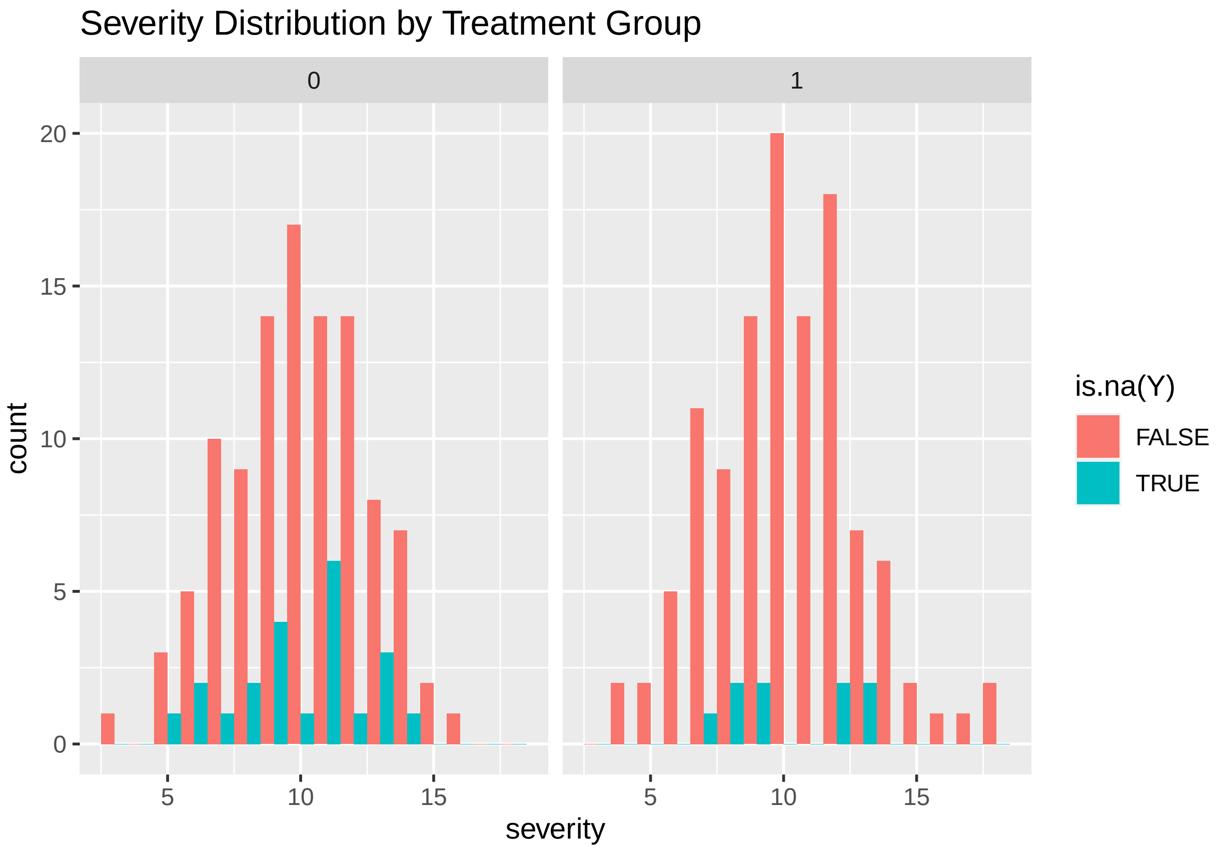
```
## Within treatment group
updata |>
  group_by( A, Y_missing) |>
  summarise(Age_mean = mean(age),
            Age_sd = sd(age),
            Severity_mean = mean(severity),
            Severity_sd = sd(severity))
```

```
## `summarise()` has grouped output by 'A'. You can override using the `.groups`
## argument.

## # A tibble: 4 x 6
## # Groups:   A [2]
##       A Y_missing Age_mean Age_sd Severity_mean Severity_sd
##   <dbl>     <dbl>    <dbl>  <dbl>         <dbl>       <dbl>
## 1     0         0     64.2   3.79         10.1         2.52
## 2     0         1     64.3   4.73          9.90        2.55
## 3     1         0     64.4   3.89         10.2         2.73
## 4     1         1     65.3   4.67         10.2         2.22
```

```
data %>%
  ggplot(aes(x = age, fill = is.na(Y))) +
  geom_histogram(binwidth = 1, position = "dodge") +
  facet_wrap(~ A) +
  labs(title = "Age Distribution by Treatment Group")
```



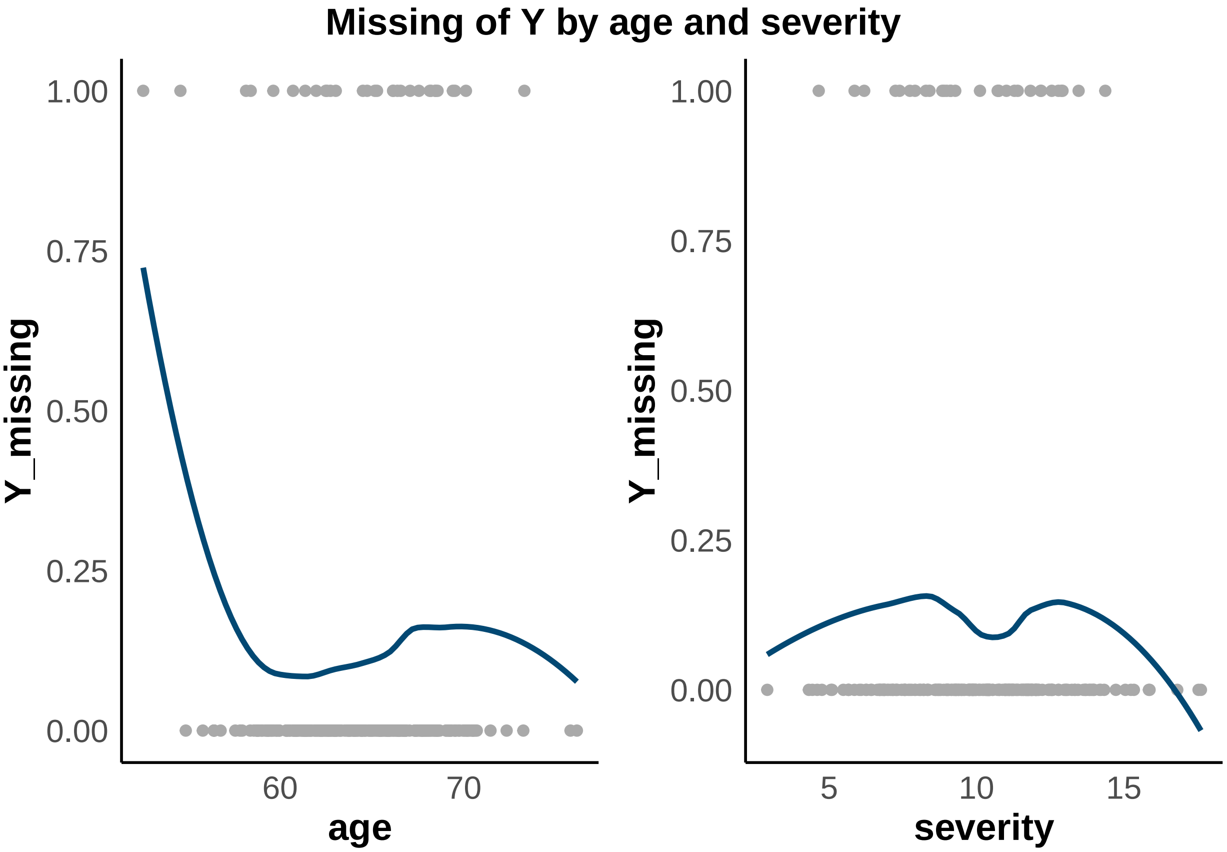
```r
data %>%
  ggplot(aes(x = severity, fill = is.na(Y))) +
  geom_histogram(binwidth = 1, position = "dodge") +
  facet_wrap(~ A) +
  labs(title = "Severity Distribution by Treatment Group")
```



```r
# Create custom theme
custom_theme <- theme(
  panel.background = element_blank(),
  axis.ticks = element_blank(),
  axis.text = element_text(size = 12),
  axis.title = element_text(size = 14, face = "bold"),
  axis.line = element_line(size = 0.5)
)
```

```
## Warning: The `size` argument of `element_line()` is deprecated as of ggplot2 3.4.0.
## i Please use the `linewidth` argument instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```r
# Scatterplot for age
updata <- data |>
  mutate(
    Y_missing = ifelse(is.na(Y) == T, 1, 0)
  )

scatter_age <- ggplot(updata, aes(x = age, y =Y_missing)) +
  geom_point(color = "darkgray") +
  geom_smooth( method = "loess",color = "#024873", se = FALSE) +
  custom_theme

scatter_severity <- ggplot(updata, aes(x = severity, y = Y_missing)) +
  geom_point(color = "darkgray") +
  geom_smooth( method = "loess",color = "#024873", se = FALSE) +
  custom_theme

library(grid)
library(gridExtra)
grid.arrange(scatter_age, scatter_severity, nrow = 1,
             top=textGrob("Missing of Y by age and severity",
                          gp = gpar(fontsize = 14, fontface = "bold")))
```



2. Based on your exploratory analysis, specify a model for Y to be used in a single and multiple (M = 20) predicted mean imputation.

From the exploratory analyses, we observed that the proportion of missing outcomes is 0.124 overall, higher among standard care (0.173) compared to the treatment group (0.073). When plotting missingness of outcomes against age and severity, the distribution of age and servity are different for missing vs. non-missing data overall or in treatment group, indicating that these variables may influence the missingness. This suggests that age, severity, and treatment assignment should be considered as factors in the imputation models.

```
#  Single predicted mean
imp_predmean <- mice(data[,-1], method = "norm",
                     m = 1, seed = 654, maxit = 50, print = F)
# multiple (M = 20) predicted mean imputation.
imp_mi <- mice(data[,-1], method = "norm",
               m = 20, maxit = 50, seed = 654, print = F)
```

3. Estimate the marginal treatment effect, with a 95% confidence interval, using the complete cases and the data from the single and multiple imputation. Compare the estimates and whether the data supports a benefit to quality of life of the novel targeted blood pressure management protocol compared to the standard of care.

```r
# Complete case analysis
fit_compcase <- lm(Y ~ as.factor(A), na.action=na.omit, data = data)
summary(fit_compcase)
```

```
##
## Call:
## lm(formula = Y ~ as.factor(A), data = data, na.action = na.omit)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.7088  -4.3714  -0.5364   4.2946  16.7574
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    74.6910     0.6110 122.244   <2e-16 ***
## as.factor(A)1   2.0055     0.8469   2.368   0.0188 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.261 on 217 degrees of freedom
##   (31 observations deleted due to missingness)
## Multiple R-squared:  0.02519,    Adjusted R-squared:  0.0207
## F-statistic: 5.608 on 1 and 217 DF,  p-value: 0.01876
```

```r
ci_compcase <- confint(fit_compcase)["as.factor(A)1", ]


# Single predicted mean
data_predmean <- data
data_predmean$Y_impute <- mice::complete(imp_predmean)$Y
fit_predmean <- lm(Y_impute ~ as.factor(A), data = data_predmean)
summary(fit_predmean)
```

```
##
## Call:
## lm(formula = Y_impute ~ as.factor(A), data = data_predmean)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.9870  -4.4746  -0.4074   4.1224  16.8089
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)     74.969      0.552 135.808   <2e-16 ***
## as.factor(A)1    1.676      0.787   2.129   0.0342 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 6.221 on 248 degrees of freedom
## Multiple R-squared:  0.01795,    Adjusted R-squared:  0.01399
## F-statistic: 4.534 on 1 and 248 DF,  p-value: 0.03421
```

```
ci_predmean <- confint(fit_predmean)["as.factor(A)1", ]


# Multiple imputation
data_imp_mi <- mice::complete(imp_mi, action = "long", include = TRUE)
# Save as mids object
data_imp_mi <- as.mids(data_imp_mi)
# Run the models using mice function with
fit_imp_mi <- with(data_imp_mi, lm(Y ~ as.factor(A)))
# Pool the estimates
summary_multiple <- summary(pool(fit_imp_mi))


ci_multiple <- c((summary_multiple$estimate[2]-1.96*summary_multiple$std.error[2]
),
(summary_multiple$estimate[2]+1.96*summary_multiple$std.error[2]
))


# Combine altogether in one table
imp_results <-
  data.frame(model = c("Complete case", "Single predicted mean", "Multiple predicted mean"),
             beta = unlist(c(coef(fit_compcase)[2],
                             coef(fit_predmean)[2],
                             summary(pool(fit_imp_mi))[2,2])),
             se = unlist(c(sqrt(diag(vcov(fit_compcase)))[2],
                           sqrt(diag(vcov(fit_predmean)))[2],
                           summary(pool(fit_imp_mi))[2,3])),
             LowerCI = unlist(round(c(ci_compcase[1],
                                      ci_predmean[1],
                                      ci_multiple[1]),3)),
             UpperCI = unlist(round(c(ci_compcase[2],
                                      ci_predmean[2],
                                      ci_multiple[2]),3))
             )

# Combine altogether in one table

imp_results
```

```
##                       model     beta        se LowerCI UpperCI
## 1           Complete case 2.005471 0.8468594   0.336   3.675
## 2   Single predicted mean 1.675798 0.7870019   0.126   3.226
## 3 Multiple predicted mean 2.048971 0.8386283   0.405   3.693
```

```
kable(imp_results, caption = "Marginal Treatment Effect Estimates with 95% CIs")
```

Table 1: Marginal Treatment Effect Estimates with 95% CIs

model	beta	se	LowerCI	UpperCI
Complete case	2.005471	0.8468594	0.336	3.675
Single predicted mean	1.675798	0.7870019	0.126	3.226
Multiple predicted mean	2.048971	0.8386283	0.405	3.693

The marginal treatment effect, with a 95% confidence interval, using the complete cases and the data from the single and multiple imputation are 2.01 (95%CI 0.34, 3.68), 1.68 (95%CI 0.13, 3.23) and 2.05 (95%CI 0.41, 3.69), respectively. The results supported a benefit to quality of life of the novel targeted blood pressure management protocol compared to the standard of care.

4. Write an abstract of no more than 500 words with an objective, study design and data, methods, results and discussion. Be sure to focus on the objective of the trial, i.e. to evaluate the efficacy of the novel targeted blood pressure management protocol. Provide sufficient details and results from both the complete case and multiple imputation results. And be quantitative in your results section

Objective To evaluate the efficacy of novel targeted blood pressure management protocol (intervention) compared to standard care (control) in improving patients' quality of life.

Study design and data This randomized controlled study involved 250 adult stroke patients, randomly assigned to receive either the novel protocol or standard care. Using the complete cases and the data from the single and multiple imputation, we explored the effect of treatment on the improvement of quality of life.

Methods We conducted an exploratory analysis of the missing data in this stroke trial to explore the patterns of missing data. We compared the proportion of missing outcomes and the distribution of age and disease severity for those with and without missing data, considering the different treatment assignment. To interpolate the missing data, we conducted a single predicted mean imputation and multiple (M=20) predicted mean imputation using the observed patients age, condition severity and treatment assignment as predictors. Then we fit a linear regression to estimate the marginal treatment effect, adjusting for age and severity, using the complete cases, single and multiple imputation data.

Results A total of 250 patients participated in the study. The proportion of missing outcomes was 0.124 overall, 0.173 for standard of care and 0.073 for the novel protocol group. The mean age was 64.3 (SD 3.84) years for non-missing participants and 64.6 (SD 4.65) for missing data. Mean severity scores were 10.1 (SD 2.63) for non-missing participants, and 9.97 (SD 2.43) for missing data. The distribution of age was different for those with and without missing data (overall and within treatment group). The marginal treatment effects, with 95% confidence intervals, were 2.01 (95% CI: 0.34, 3.68) for complete cases, 1.68 (95% CI: 0.13, 3.23) for single imputation, and 2.05 (95% CI: 0.41, 3.69) for multiple imputation.

Discussion The results supported a benefit to quality of life of the novel targeted blood pressure management protocol compared to the standard care. The multiple imputation result was consistent with the complete cases analysis. The multiple imputation analysis, designed to account for the uncertainty due to missing data, provided a comparable effect size to the complete case analysis. The positive treatment effect observed in the complete case and imputation analyses suggests that the novel targeted blood pressure management protocol may be more effective than the standard of care in improving the quality of life for patients with hypertension.

5. The estimated marginal treatment effect in the trial had there been no missing data was 0.71 (95% CI: -0.96, 2.34, p-value 0.39) (I know this because I simulated the dataset). Explain what may account for the differences between the estimated marginal treatment effect had there been no missing data, based on the complete cases and based on the imputation approach.

Under the scenario with no missing data, the estimated marginal effect is much smaller than the results above and shows no significance. Several factors could contribute to this difference:

1) **Selection Bias:** The observed complete cases only include subjects with complete data. If the missing mechanism is not Missing Completely At Random (MCAR), analyzing only the complete data can introduce bias. Particularly, if the missingness is related to the unobserved data (Missing Not At Random, MNAR mechanism), the information from the remaining subjects may not accurately represent the characteristics of the overall population. This can lead to statistical bias, potentially overestimating the effect towards a more positive outcome. While multiple imputation may mitigate this bias compared to single imputation, it may still broaden the confidence interval without fully addressing the issue.
2) **Limitations of Imputation Methods:** Both single and multiple imputation methods assume Missing At Random (MAR) or MCAR and utilize linear models for imputation. However, they may not fully account for nonlinear relationships and are not robust when these assumptions are violated. Consequently, if there are nonlinear missing associations or MNAR mechanisms present, the estimated results may be heavily skewed.

6. If you are working in a group of 2 or 3 students:

a. Provide the names of your group members Sunan Gao, Chunyu Liu, Siyu Zou
b. Describe your contributions to the analysis, interpretation and writing. Each group member provides a version of the results, followed by a discussion and subsequent amendments

III. Connection of logistic regression to 2x2 tables; confounding and effect modification

Upon mastery of this problem, a student should be able to: • create one or multiple 2x2 tables from which to estimate log odds ratios that correspond to coefficients from simple or multiple logistic regressions • appreciate the invariance of the odds ratio as one important reason logistic regression is popular in epidemiology • pool log odds ratios across strata using weighted averages as an approximation to logistic regression

Use the National Medical Expenditure Survey (NMES) data set for this problem. The general goal is to describe the association of self-reported smoking with the indicator of major smoking-caused disease (mscd), a group of diseases the U.S. Surgeon General and WHO say are caused by smoking.

Part A: Simple logistic regression

1. Define a variable mscd to represent whether or not a person has a major smoking caused disease (e.g. lc5 or chd5 =1). Make a 2x2 table of mscd against eversmk (1-yes; 0-no). Calculate the log odds ratio, its standard error and 95% CI using 652 methods for 2x2 tables. To simplify the analysis, drop those people who have a missing value of eversmk.

```r
load("nmes.rdata")
d <- nmes
d[d == "."] <- NA
```

```r
data1 <- d |>
  filter(!is.na(eversmk)) |>
  mutate(mscd = ifelse(lc5 + chd5 > 0, 1, 0),
         bigexp = ifelse(totalexp > 1000, 1, 0))

tab <- table(macd = data1$mscd, eversmk = data1$eversmk)
tab
```

```
##     eversmk
## macd    0    1
##    0 4626 5739
##    1  433  886
```

```r
a <- tab[2,2]
b <- tab[1,2]
c <- tab[2,1]
d <- tab[1,1]
# logOR <- log((4626*886) / (5739*433))
logOR <- log((a * d) / (b * c))
# standard error and 95% CI
SE <- sqrt(1/a + 1/b + 1/c + 1/d)
CI_lower <- logOR - 1.96 * SE
CI_upper <- logOR + 1.96 * SE

unadjusted_logOR = data.frame( "Log odds ratio" = logOR, "Standard error" = SE, "CI_lower" = CI_lower,
)
unadjusted_logOR
```

```
##   Log.odds.ratio Standard.error  CI_lower  CI_upper
## 1      0.5003868      0.0618753 0.3791112 0.6216624
```

2. Regress mscd on eversmk using logistic regression. Compare the regression coefficient and its standard error with the log odds ratio and standard error in Part A Question 1 above.

```r
model1 <- glm(mscd ~ eversmk, data = data1, family = "binomial")
summary(model1)
```

```
##
## Call:
## glm(formula = mscd ~ eversmk, family = "binomial", data = data1)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.36871    0.05026 -47.133  < 2e-16 ***
## eversmk1     0.50039    0.06188   8.087 6.11e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 8237.5  on 11683  degrees of freedom
## Residual deviance: 8169.5  on 11682  degrees of freedom
## AIC: 8173.5
```

```
##
## Number of Fisher Scoring iterations: 5
```

```
model_logistic = data.frame( "Regression coefficien" = summary(model1)$coefficients["eversmk1", "Estima
)
model_logistic
```

```
##   Regression.coefficien Standard.error
## 1             0.5003868      0.0618753
```

The regression coefficient and its standard error are same to the log odds ratio and standard error in Part A Question 1 above.

3. Use logistic regression to regress eversmk on mscd. Compare the log odds ratio and standard error from this regression with those from Part A Questions 1 and 2.

```
model2 <- glm(as.factor(eversmk) ~ mscd, data = data1, family = binomial)
summary(model2)
```

```
##
## Call:
## glm(formula = as.factor(eversmk) ~ mscd, family = binomial, data = data1)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.21559    0.01976  10.911  < 2e-16 ***
## mscd         0.50039    0.06188   8.087 6.11e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 15987  on 11683  degrees of freedom
## Residual deviance: 15919  on 11682  degrees of freedom
## AIC: 15923
##
## Number of Fisher Scoring iterations: 4
```

```
model2_reverse = data.frame( "Regression coefficien" = summary(model2)$coefficients["mscd", "Estimate"]
)
model2_reverse
```

```
##   Regression.coefficien Standard.error
## 1             0.5003868     0.06187529
```

The log odds ratio and standard error from this regression are also same to the results in Questions 1 and 2.

4. Write a couple of sentences that can be used to teach a public health professional a) the interpretation of the logistic regression coefficient, b) the invariance property of the odds ratio and c) why the invarince property is important for public health research (HINT: this about different study designs that may generate data within a 2x2 table)

The logistic regression coefficient means the log odds ratio. In this result, the logistic regression coefficient is 0.5003868, means the log odds of major smoking-caused disease (mscd) among those ever smoker are roughly

0.500 (95% CI 0.379, 0.622) times the log odds among those not smoking. We can exponentiate the logistic regression coefficient to get the odds ratio. The odds of having MSCD among those ever smoker are 1.65 times the odds among those ever smoker.

Invariance property of the odds ratio: This means the odds ratio remains consistent regardless of which variable is the “exposure” and which is the “outcome”. This reveals the link between retrospective (case-control) and prospective (cohort) studies for the estimation of risk of disease due to exposure. In prospective study, we took samples of exposed and unexposed individuals at the beginning of study and follow them up. With the retrospective (case-control) approach, we sample according to disease status rather than exposure status. $\frac{P(D|E)/P(\overline{D}|E)}{P(D|\overline{E})/P(\overline{D}|\overline{E})} = \frac{P(E|D)/P(\overline{E}|D)}{P(E|\overline{D})/P(\overline{E}|\overline{D})}$.

Why the invarince property is important for public health research

The OR calculated from a prospective sampling is identical to the OR from a retrospective sampling. The relative risk, the ratio of disease incidences, can be approximated by the case-control OR (under the rare disease assumption), even though the latter provides no information about the absolute magnitude of the incident rates in the exposed and unexposed groups. This is particularly important in public health research because it enables the generalization of results from studies with different designs, settings, or populations. Whether the data come from a retrospective (case-control) and prospective (cohort) study, the odds ratio can provide a consistent measure of association, thus enhancing the comparability of study findings across diverse public health contexts.

Part B. Association of mscd and eversmk, controlling for age.

1. Stratify age by: <50, 51-60, 61-70, >70. Within each stratum, calculate the log odds ratio and standard error for the mscd-eversmk association. Complete the table below. Here, weight is defined to be the inverse of the variance normalized to sum to 1.0 across strata: weightj=(1/sej$^{2)/\mathrm{sumj}(1/\mathrm{sej}}$2)

```
data1$age_stratum <- cut(as.numeric(data1$lastage),
                         breaks = c(-Inf, 50, 60, 70, Inf),
                         labels = c("<50", "51-60", "61-70", ">70"), right = FALSE)


results <- data.frame(
    age_stratum = c("<50", "51-60", "61-70", ">70"),
    log_odds_ratio = numeric(4),
    std_error = numeric(4),
    Inverse_se2 = numeric(4),
    weight = numeric(4)
)

log_odds_ratio <- function(subdata) {
  model <- glm(mscd ~ eversmk, data = subdata, family = binomial)
  coefficient <- coef(summary(model))["eversmk1", "Estimate"]
  se <- coef(summary(model))["eversmk1", "Std. Error"]
  return(c(coefficient, se))
}

# Calculate log odds ratio and standard error for each stratum
for (stratum in levels(data1$age_stratum)) {
      subdata <- subset(data1, age_stratum == stratum)
      results[results$age_stratum == stratum, c("log_odds_ratio", "std_error")] <- log_odds_ratio(subda


# Calculate the weights for each stratum
```

```r
results$Inverse_se2 <- 1 / (results$std_error^2)
sum_inv_variance <- sum(results$Inverse_se2)
results$weight <- results$Inverse_se2 / sum_inv_variance
# View the results
print(results)
```

```
##   age_stratum log_odds_ratio std_error Inverse_se2     weight
## 1         <50      0.7999698 0.2805223    12.70765 0.05378017
## 2       51-60      1.0043238 0.2046128    23.88551 0.10108609
## 3       61-70      0.7470741 0.1186463    71.03811 0.30064105
## 4         >70      0.5766316 0.0881622   128.65752 0.54449269
```

2. Calculate the inverse-variance weighted average log odds ratio from the data above and its standard error (SE = sqrt{1/sumj[1/se2j] }).

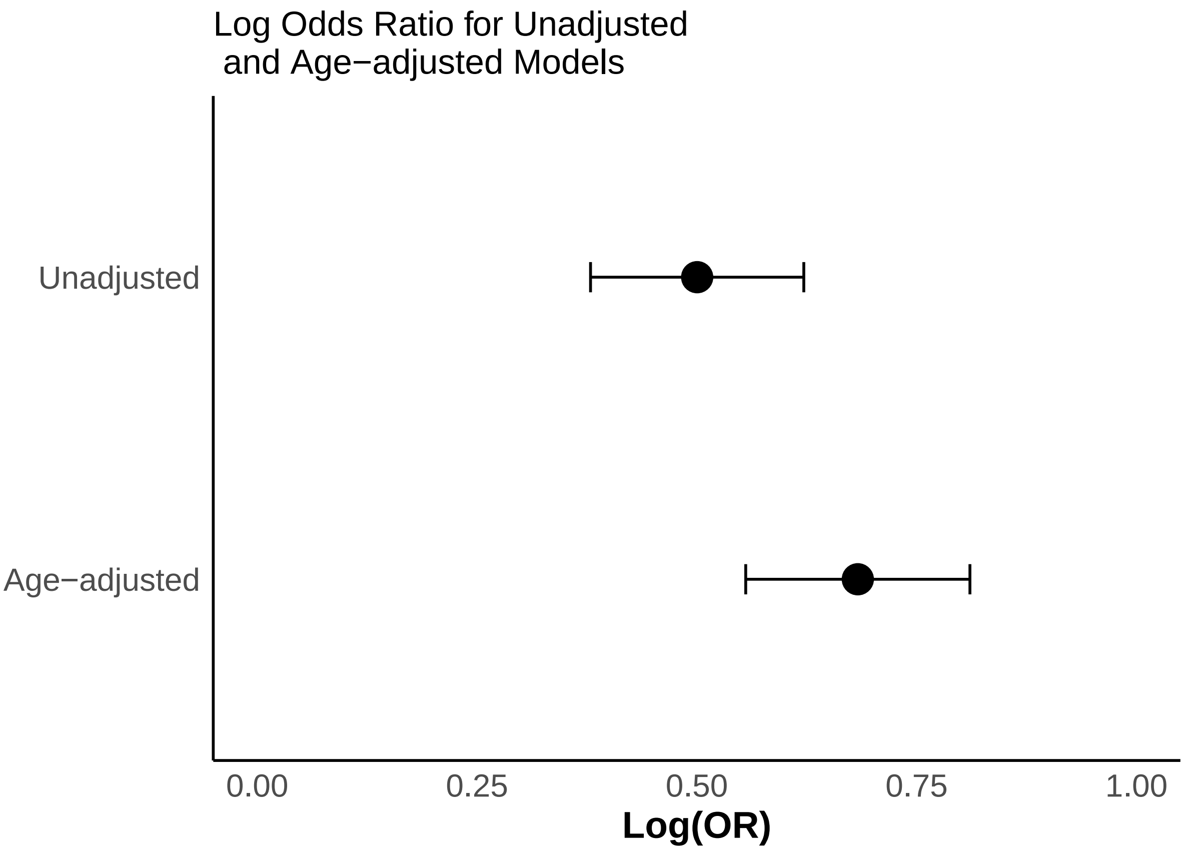
Create a plot to compare the age-adjusted log odds ratio to the unadjusted log-odds ratio from Part A. Question 1, include 95% confidence intervals in the plot. Add any additional information for evaluating whether age is a confounder to your figure. Explain in a sentence or two whether age is a “confounder” of the disease-smoking association and why?

```r
# inverse-variance weighted average log odds ratio
weighted_log_odds_ratio <- sum(results$log_odds_ratio * results$weight)
# standard error for the weighted average log odds ratio
weighted_se <- sqrt(1 / sum_inv_variance )

plot_data <- data.frame(
  Type = c( "Age-adjusted", "Unadjusted"),
  Log_OR = c(weighted_log_odds_ratio, unadjusted_logOR$`Log.odds.ratio`),
  SE = c( weighted_se, unadjusted_logOR$`Standard.error`)
)

plot_data$CI_lower <- plot_data$Log_OR - 1.96 * plot_data$SE
plot_data$CI_upper <- plot_data$Log_OR + 1.96 * plot_data$SE
plot_data
```

```
##           Type    Log_OR         SE  CI_lower  CI_upper
## 1 Age-adjusted 0.6831185 0.06505466 0.5556114 0.8106257
## 2   Unadjusted 0.5003868 0.06187530 0.3791112 0.6216624
```

```r
# Create custom theme
custom_theme <- theme(
  panel.background = element_blank(),
  axis.ticks = element_blank(),
  axis.text = element_text(size = 12),
  axis.title = element_text(size = 14, face = "bold"),
  axis.line = element_line(size = 0.5)
)

# Visualize in a figure
results1_plot <- ggplot(plot_data, aes(y = factor(Type))) +
  geom_point(aes(x = Log_OR), size = 5) +
  geom_errorbar(aes(xmin = CI_lower, xmax = CI_upper), linewidth = 0.5, width = 0.1) +
  scale_x_continuous(limits = c(0, 1)) +
  labs(y = "", x = "Log(OR)", title = "Log Odds Ratio for Unadjusted\n and Age-adjusted Models") +
  custom_theme
```

```
results1_plot
```



```
# Run the bootstrap
library(boot)
```

```
##
## Attaching package: 'boot'

## The following object is masked from 'package:biostat3':
##
##     melanoma

## The following object is masked from 'package:survival':
##
##     aml
```

```
set.seed(653)

# my.boot <- function(dat, x) {
#   d = dat[x,]
#   model1 <- glm(mscd ~ eversmk, data = d, family = "binomial")
#   model3 <- glm(mscd ~ eversmk + as.factor(age_stratum), data = d, family = "binomial")
#   model3$coefficients[2] - model1$coefficients[2]
# }

my.boot <- function(dat, x) {
  d = dat[x,]
  ## (Non)-Weighted-estimate
  d$eversmk = as.numeric(d$eversmk)
  strata_results <- d %>%
    filter(!is.na(eversmk)) %>%
    group_by(age_stratum) %>%
    summarize(
      a = sum(mscd == 1 & eversmk == 1),
      b = sum(mscd == 0 & eversmk == 1),
      c = sum(mscd == 1 & eversmk == 0),
      d = sum(mscd == 0 & eversmk == 0),
      log_or = log((a*d)/(b*c)),
      se = sqrt(1/a + 1/b + 1/c + 1/d),
      inv_se2 = 1/se^2
    ) %>%
    mutate(weight = inv_se2 / sum(inv_se2))
  weighted_log_or <- sum(strata_results$log_or * strata_results$weight)
  model1 <- glm(mscd ~ eversmk, data = d, family = "binomial")
  weighted_log_or - model1$coefficients[2]
}

results1_boot <- boot(data1, my.boot, R = 100)
boot.ci(results1_boot, type = "perc")
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 100 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = results1_boot, type = "perc")
##
## Intervals :
## Level     Percentile
## 95%   ( 0.1422,  0.2173 )
## Calculations and Intervals on Original Scale
## Some percentile intervals may be unstable
```

The estimate of the change $\beta_{adjusted}$ - $\beta_{unadjusted}$ =0.18 and a 95% bootstrap percentile confidence interval (CI): 0.14 to 0.22. The unadjusted and adjusted estimates are statistically different, which suggests age may be a confounder.

3. Use logistic regression to regress mscd on eversmk and 3 indicator variables for the 4 age strata, i.e. make age category a factor. Compare the resulting eversmk coefficient and standard error with the value above in Part B Question 2.

```
model3 <- glm(mscd ~ eversmk + as.factor(age_stratum), data = data1, family = "binomial")
summary_model3 <- summary(model3)

log_or_model3 <- coef(summary_model3)["eversmk1", "Estimate"]
se_model3 <- coef(summary_model3)["eversmk1", "Std. Error"]

compare <- data.frame(
  model = c("Inverse-variance weighting", "Logistic regression estimation"),
  Log_OR = c(weighted_log_odds_ratio, log_or_model3),
  SE = c(weighted_se, se_model3)
)

compare
```

```
##                             model    Log_OR         SE
## 1     Inverse-variance weighting 0.6831185 0.06505466
## 2 Logistic regression estimation 0.6867405 0.06443204
```

The resulting eversmk coefficient and standard error are similar to the value above in Part B Question 2, calculated by inverse-variance weighting.

4. Now repeat the analysis controlling for age with your favorite smooth function of continuous age.

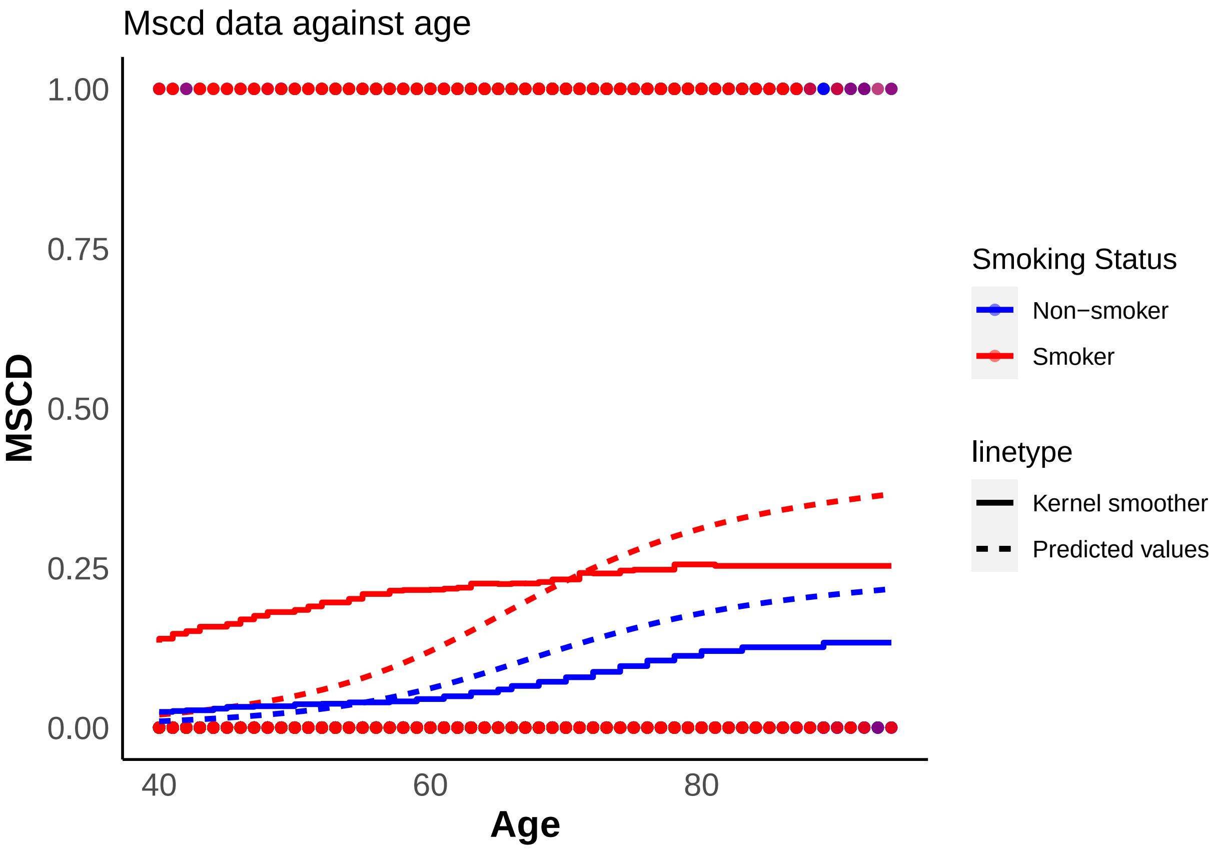
```
library(splines)
model_continuous <- glm(mscd ~ eversmk +  ns(lastage, df=3), data = data1, family = "binomial")
summary_continuous <- summary(model_continuous)
log_or_continuous <- summary_continuous$coefficients["eversmk1", "Estimate"]
se_continuous <- summary_continuous$coefficients["eversmk1", "Std. Error"]
cat("Age continuous:", log_or_continuous, ", SE:", se_continuous, "\n")
```

```
## Age continuous: 0.7318652 , SE: 0.06561812
```

5. Plot the mscd data against age using the eversmk value as the plotting symbol or color.

Add the predicted values from your model and a kernel smoother fit separately to each smoking group for comparison.

```
data1$predicted <-  predict(model_continuous, type = "response")

# Kernel smoother with bandwidth 20 months
data1 <- data1 |>
  arrange(eversmk, lastage, mscd) |>
  mutate(ksmooth_mscd = ksmooth(lastage, mscd, bandwidth = 20)$y)

ggplot(data1, aes(x = lastage, y = mscd, color = factor(eversmk))) +
  geom_point(alpha = 0.5) +
  geom_line(aes(y=predicted, linetype = "Predicted values"), linewidth = 1)  +
  geom_line(aes(y=ksmooth_mscd, linetype = "Kernel smoother"),  linewidth = 1)  +
  labs(x = "Age",
       y = "MSCD",
       title = "Mscd data against age",
       color = "Smoking Status"
       ) +
  scale_color_manual(values = c("blue", "red"), labels = c("Non-smoker", "Smoker")) +
  custom_theme
```



Compare the model predictions with the kernel smoothers to see if there is evidence of effect modification of the mscd-smoking association by age?

Yes, there is effect modification of the mscd-smoking association by age.

6. Propose an extended model to directly address the possibility that age modifies the effect of smoking on disease prevalence.

Fit this model and compare it to the model without effect modification using a likelihood ratio test.

```r
library(splines)
library(lmtest)
model_extend <- glm(mscd ~ eversmk * lastage, data = data1, family = binomial)
# anova(model_continuous, model_extend, test = "LRT")
model_add<- glm(mscd ~ eversmk  + lastage, data = data1, family = binomial)
anova(model_add, model_extend, test = "LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: mscd ~ eversmk + lastage
## Model 2: mscd ~ eversmk * lastage
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1     11681     7380.7
## 2     11680     7380.6  1 0.063812   0.8006
```

```r
model_extend_ns <- glm(mscd ~ eversmk * ns(lastage,df=3), data = data1, family = binomial)
model_add_ns <- glm(mscd ~ eversmk +  ns(lastage,df=3), data = data1, family = "binomial")
anova(model_extend_ns, model_add_ns, test = "LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: mscd ~ eversmk * ns(lastage, df = 3)
## Model 2: mscd ~ eversmk + ns(lastage, df = 3)
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1     11676     7333.4
## 2     11679     7339.5 -3  -6.0825   0.1077
```

The likelihood ratio test compares models with age as a confounder (continuous or natural spline) and age as an effect modifier are not statistically significant different, with a p-value 0.8006 and 0.1077, respectively. This indicates that age not significantly modifies the effect of smoking on disease prevalence.