PHP 2550_Project 3

Himashree Chandru

2023-11-27

Abstract

Cardiovascular disease (CVD) poses a significant global health challenge, underscoring the need for accurate risk prediction models. This collaborative project evaluates the transportability of a cardiovascular risk prediction model from the Framingham Heart Study to diverse populations, using both real and simulated data from the NHANES dataset. Our study employs logistic regression models, inverse odds weights, and Brier scores for evaluation. Results indicate promising transportability, with the risk model performing well in both the NHANES and simulated target populations. However, simulations reveal nuances in model performance, emphasizing the importance of refining data-generating mechanisms for more accurate predictions.

Introduction

Cardiovascular disease (CVD) remains a leading cause of morbidity and mortality worldwide, necessitating accurate risk prediction models for timely intervention and prevention (World Health Organization, 2021). In the realm of predictive modeling, the application of risk prediction models across diverse populations is of paramount importance. This collaborative project, conducted in partnership with Dr. Jon Steingrimsson from the Biostatistics Department, focuses on evaluating the transportability of a cardiovascular risk prediction model originally developed on a source population when applied to a target population both when data from the target population is available, and when it is simulated.

The predictive model under scrutiny originates from the Framingham Heart Study, a seminal investigation comprising participants aged 30-62 (National Heart, Lung, and Blood Institute). Historically, such models, while valuable in their original context, have demonstrated challenges when extrapolated to populations with different demographic compositions, as seen in the example of the Framingham ATP-III model's suboptimal generalization to multi-ethnic populations (Steingrimsson).

The objectives of this project are two-fold:

- Evaluate performance of a cardiovascular risk prediction model (D'Agostino et al., 2008) in a target population underlying NHANES: The evaluation is conducted using the weighting estimator for the Brier score in the target population (Steingrimsson et al., 2022).
- Conduct the same analysis when the target population is simulated: The simulation process involves utilizing summary statistics from the NHANES data to simulate the target population. The same estimator for weighted Brier score in the target population is used to evaluate the performance of the models in the simulated populations.

Data and Preprocessing

The Framingham data is obtained from the `riskCommunicator` package, and it comes from the Framingham Heart Study, a study that began in 1948 by collecting data from individuals between the ages of 30-62 from Framingham, Massachusetts, with the aim of identifying risk factors for cardiovascular disease (National Heart, Lung, and Blood Institute).

NHANES data comes from annual surveys conducted by the NHANES program, whose participants are selected to be representative of the U.S. population belonging to all ages (Centers for Disease Control and Prevention). The NHANES data is obtained from the `nhanesA` package.

The preprocessing involves using data from the Framingham study to obtain a sample of source population data. From the Framingham data, the variables CVD (indicating whether myocardial infarction, fatal coronary heart disease, atherothrombotic infarction, cerebral embolism, intracerebral hemorrhage, or subarachnoid hemorrhage or fatal cerebrovascular disease occured during followup), TIMECVD (number of days from baseline exam to first CVD event during followup or number of days from baseline to censor date), SEX (participant sex), TOTCHOL (serum total cholesterol in mg/dL), AGE (age at exam), SYSBP (systolic blood pressure in mmHg; mean of last two of three measurements), DIABP (diastolic blood pressure in mmHg; mean of last two of three measurements), CURSMOKE (indicating whether participants are current smokers), DIABETES (indicating whether the participant is diabetic), BPMEDS (whether anti-hypertensive medication was being used at the time of exam), HDLC (high density lipoprotein cholesterol in mg/dL) and BMI (body mass index) are extracted, and any NA's are removed.

Different variables (`SYSBP_T` and `SYSBP_UT`) are created to store the blood pressure values depending on whether or not the individuals were on anti-hypertensive medication. In addition, any observations corresponding to individuals without cardiovascular events within 15 years were removed (removal of censored data). Two different datasets corresponding to males and females was then created.

Then, from the NHANES data for the year 2017-18, variables corresponding to those selected from the Framingham study (above) are similarly selected to create a separate dataset corresponding to a sample of the target population data. The observations are filtered to only include data corresponding to participants belonging to ages 30-62 so that the eligibility criteria for the source population (Framingham study participants) is met.

Summary statistics for the two datasets are calculated stratified by sex, and presented in the tables below. It can be seen that the distributions of the different variables are different for each sex in both datasets. In addition, the distributions for each sex is also different between the two datasets, suggesting varying demographics and risk levels.

```
##                       Stratified by SEX
##                        1                 2                 p      test
##   n                    1110              1468
##   TIMECVD (mean (SD)) 7226.18 (2402.62) 7952.63 (1830.88) <0.001
##   SEX (mean (SD))        1.00 (0.00)       2.00 (0.00)    <0.001
##   TOTCHOL (mean (SD))  226.34 (41.49)    246.22 (45.91)   <0.001
##   AGE (mean (SD))       60.08 (8.23)      60.62 (8.41)     0.102
##   SYSBP (mean (SD))    138.90 (21.05)    140.02 (23.74)    0.215
##   DIABP (mean (SD))     81.88 (11.41)     80.33 (11.08)    0.001
##   HDLC (mean (SD))      43.58 (13.36)     53.03 (15.69)   <0.001
##   BMI (mean (SD))       26.21 (3.49)      25.55 (4.25)    <0.001
```

```
##                       Stratified by SEX
##                        1              2              p      test
##   n                    1417           1583
##   SYSBP (mean (SD))   126.07 (16.63) 122.46 (18.76) <0.001
##   SEX (mean (SD))       1.00 (0.00)    2.00 (0.00)  <0.001
##   AGE (mean (SD))      47.16 (9.97)   46.75 (9.85)   0.251
##   BMI (mean (SD))      30.19 (6.79)   30.77 (8.37)   0.048
##   HDLC (mean (SD))     47.45 (14.54)  57.59 (16.25) <0.001
##   TOTCHOL (mean (SD)) 192.86 (40.71) 195.40 (38.95)  0.098
```

Table 1: Summary Statistics for Framingham Data

	1	2	p	test
n	1110	1468		
TIMECVD (mean (SD))	7226.18 (2402.62)	7952.63 (1830.88)	<0.001	
SEX (mean (SD))	1.00 (0.00)	2.00 (0.00)	<0.001	
TOTCHOL (mean (SD))	226.34 (41.49)	246.22 (45.91)	<0.001	
AGE (mean (SD))	60.08 (8.23)	60.62 (8.41)	0.102	
SYSBP (mean (SD))	138.90 (21.05)	140.02 (23.74)	0.215	
DIABP (mean (SD))	81.88 (11.41)	80.33 (11.08)	0.001	
HDLC (mean (SD))	43.58 (13.36)	53.03 (15.69)	<0.001	
BMI (mean (SD))	26.21 (3.49)	25.55 (4.25)	<0.001	

Table 2: Summary Statistics for NHANES Data

	1	2	p	test
n	1417	1583		
SYSBP (mean (SD))	126.07 (16.63)	122.46 (18.76)	<0.001	
SEX (mean (SD))	1.00 (0.00)	2.00 (0.00)	<0.001	
AGE (mean (SD))	47.16 (9.97)	46.75 (9.85)	0.251	
BMI (mean (SD))	30.19 (6.79)	30.77 (8.37)	0.048	
HDLC (mean (SD))	47.45 (14.54)	57.59 (16.25)	<0.001	
TOTCHOL (mean (SD))	192.86 (40.71)	195.40 (38.95)	0.098	

Methods

NHANES Target Population

First, the data from the target population (NHANES data) has missing values, which we explore.

Table 3: Summary of Missing Data in NHANES

variable	n_miss	pct_miss
SYSBP_UT	524	17.4666667
SYSBP	468	15.6000000
HDLC	316	10.5333333
TOTCHOL	316	10.5333333
SYSBP_T	300	10.0000000
BPMEDS	196	6.5333333
BMI	170	5.6666667
DIABETES	1	0.0333333
SEQN	0	0.0000000
SEX	0	0.0000000
AGE	0	0.0000000
CURSMOKE	0	0.0000000

On looking at the missing value summary, we see that the percent of missing variables is not too high (highest percent missing is <18%), and hence complete case analysis is employed.

Risk Score Model

The model to be fit is a logistic regression model where CVD is the response variable, and the predictors include the logarithmically transformed variables log(HDLC), log(TOTCHOL), log(AGE), log(SYSBP_UT+1), and log(SYSBP_T+1), as well as the binary variables CURSMOKE and DIABETES (D'Agostino et al., 2008).

In order to obtain the fitted risk score model, a 70:30 train-test split was employed for the source population, and the model was fit on the training set. This process was done for men and women separately as their covariate distributions were found to be different in both source and target data during preprocessing.

The Brier scores from fitting the risk score model on the test data from the source population are given in Table 3.

Table 4: Brier Score Estimates in the Test Set of the Source Population

Model	Brier Score
Men	0.1897063
Women	0.1098168

Next, to evaluate model performance on the target population, we calculate the Brier score for the target population using the weighted estimator for Brier score in the target population (Steingrimsson et al., 2022):

$$\hat{\psi}_{\hat{\beta}} = \frac{\sum_{i=1}^{n} I(S_i=1, D_{test,i}=1)\hat{o}(X_i)(Y_i - g_{\hat{\beta}}(X_i))^2}{\sum_{i=1}^{n} I(S_i=1, D_{test,i}=1)}$$

where $S_i = 1$ implies the observation is in the source population,

$D_{test,i}$ implies the observation is in the test set,

$\hat{o}(X_i)$ gives estimates of the inverse-odds weights in the test set,

Y_i is the observed outcome, and

$g_{\hat{\beta}}(X_i)$ is the predicted probability of CVD.

$\hat{o}(X_i)$ is obtained by fitting a logistic regression model where the response variable is `S`, which indicates if the observation is in the source population, conditional on the same covariates as in the model to obtain weights to develop the risk score model above.

The Brier score results are given in Table 4.

Table 5: Weighted Brier Score Estimates in the Target Population

Model	Brier Score
Men	0.0656376
Women	0.0231503

Simulations

The ADEMP framework:

- Aim: To evaluate the impact of simulating data with similar distributions of covariates to the NHANES data on the estimated Brier scores of cardiovascular risk score models.
- Data generating mechanisms: Two different DGMs are considered. In the first DGM, each of the covariates in the risk score models, `SYSBP`, `AGE`, `HDLC`, `CURSMOKE`, `BPMEDS`, `TOTCHOL` and `DIABETES`, are simulated according to whether they are continuous or categorical, and their summary statistics from

the NHANES data. The continuous variables are simulated are normal random variables with the specified mean and standard distribution equal to that in the NHANES data, and the categorical random variables are simulated as binomial distributions with a trial size of 1 and a success probability equal to the proportion of 1's in those variables in the NHANES data. In the second DGM, the continuous variables were simulated in a correlated manner as a multivariate normal distribution, using the correlations between the variables in the NHANES data. A sample size of 1500 was chosen to mimic the number of observations in the test set of the target population for men and women (1372 and 1575, respectively). The simulations are run separately for men and women due to the difference in covariate distributions, and the evaluation of two separate models for men and women.

- Estimand: Brier score for the cardiovascular risk score models in the simulated target populations.
- Methods: The simulation size was determined by deciding that the Monte Carlo Standard Error of Bias be less than 0.005. By the formula $MonteCarloSE(Bias) = sqrt(Var(BrierScore)/n_{sim})$ and assuming that $SD(BrierScore) <= 0.2$ (verified), the number of simulations is determined to be 1600. In order to calculate the Brier score in the simulated population, a logistic regression model is first fit on all the test data (test data from the source population and the simulated data) to obtain the odds of membership in the source population. Inverse odds weights are then calculated and used to obtain the weighted Brier score estimate for the cardiovascular risk score model in the simulated target population. This is done separately for the male and female data. A seed value of "1234" was used for the simulation of both male and female data to ensure reproducibility.
- Performance Measures: Bias, Empirical Standard Error, and Mean Squared Error are the performance measures collected.

First, we simulate data for men according to the first DGM, and the performance measures are displayed in Table 5.

Table 6: Performance Measures for the Simulations Estimating Brier Scores for Men using DGM 1

Measure	Estimate	MCSE
thetamean	0.0512	NA
thetamedian	0.0511	NA
bias	-0.0145	1e-04
empse	0.0025	0e+00
mse	0.0002	0e+00

Next, we simulate data for women for the first DGM, and the performance measures are displayed in Table 6.

Table 7: Performance Measures for the Simulations Estimating Brier Scores for Women using DGM 1

Measure	Estimate	MCSE
thetamean	0.0138	NA
thetamedian	0.0137	NA
bias	-0.0093	0
empse	0.0018	0
mse	0.0001	0

Now, we simulate the data for men using the second DGM, and the performance measures are displayed in Table 7.

Table 8: Performance Measures for the Simulations Estimating Brier Scores for Men using DGM 2

Measure	Estimate	MCSE
thetamean	0.0120	NA
thetamedian	0.0120	NA
bias	-0.0536	0
empse	0.0009	0
mse	0.0029	0

Finally, we simulate the data for women using the second DGM, and the performance measures are displayed in Table 8.

Table 9: Performance Measures for the Simulations Estimating Brier Scores for Women using DGM 2

Measure	Estimate	MCSE
thetamean	0.0011	NA
thetamedian	0.0011	NA
bias	-0.0220	0
empse	0.0002	0
mse	0.0005	0

Results

From the Brier Scores in Table 4, it can be seen that the models transport rather well to the NHANES data, possibly due to the inverse odds weights used in the estimation of the Brier score.

From Tables 5 and 6 (simulations for men and women using DGM 1, respectively), as well as Tables 7 and 8 (simulations for men and women using DGM 2, respectively), it can be seen that the mean Brier Scores are lower than those in Table 4. This indicates that the model performs slightly better in the simulated target populations than in the NHANES target population, which in turn implies that the covariate distributions of the simulated data might not be completely similar to the actual covariate distributions in the NHANES data. The estimator seems to have a higher bias in the case of the model for men compared to that for women. The empirical standard error is lower in the case of DGM 2, indicating lower variability in estimates across simulations as compared to DGM 1. The mean squared error is also low except in the case of DGM 2 in men, indicating that the estimates have high accuracy and low variability.

Comparing the two DGMs, using correlations to generate the continuous variables as a multivariate normal distribution in the case of DGM 2 seems to lead to greater divergence from the weighted Brier scores in the target population. This indicates that one or more of the variables might not be normally distributed, necessitating further investigation to arrive at a more accurate DGM.

Limitations

In the simulation process, we acknowledge the oversimplification inherent in assuming specific distributions for covariates. The chosen data-generating mechanisms may not fully capture the complexity of real-world data, necessitating further refinement. Furthermore, the reliance on the Brier score as the primary evaluation metric, while informative, may benefit from complementarity with additional metrics to offer a more nuanced assessment of model performance.

Conclusions

Our findings contribute valuable insights into the transportability of cardiovascular risk prediction models, specifically from the Framingham Heart Study to the NHANES dataset. The successful adaptation of the model to NHANES underscores its potential utility in diverse populations. The incorporation of inverse odds weights plays a pivotal role in this achievement, facilitating effective model adjustment and enhancing its applicability across different demographic groups.

Simulations further illuminate the model's behavior in target populations, revealing nuanced dynamics. While the model performs slightly better in simulated populations compared to NHANES, this discrepancy suggests potential differences in covariate distributions. These insights underscore the necessity of refining data-generating mechanisms to better align with the intricacies of real-world scenarios.

A notable observation emerges when comparing simulation mechanisms. The use of correlations in generating continuous variables results in greater divergence from weighted Brier scores, signaling the importance of accurate data-generating mechanisms. This highlights an avenue for future research to refine and enhance simulation processes, ensuring a more faithful representation of the observed data.

Code Appendix

```
knitr::opts_chunk$set(echo = TRUE)
knitr::opts_knit$set(root.dir = "~/Downloads")

#Packages
install.packages("riskCommunicator", repos = "http://cran.us.r-project.org")
install.packages("nhanesA", repos = "http://cran.us.r-project.org")

library(riskCommunicator)
library(tidyverse)
library(tableone)
library(nhanesA)
library(naniar)
library(mice)
library(survey)
library(corrplot)
library(rsimsum)
library(knitr)
library(MASS)
data("framingham")

# The Framingham data has been used to create models for cardiovascular risk.
# The variable selection and model below are designed to mimic the models used
# in the paper General Cardiovascular Risk Profile for Use in Primary Care
# This paper is available (cvd_risk_profile.pdf) on Canvas.

framingham_df <- framingham %>% dplyr::select(c(CVD, TIMECVD, SEX, TOTCHOL, AGE,
                                      SYSBP, DIABP, CURSMOKE, DIABETES, BPMEDS,
                                      HDLC, BMI))
framingham_df <- na.omit(framingham_df)

framingham_df$CURSMOKE <- as.factor(framingham_df$CURSMOKE)
framingham_df$DIABETES <- as.factor(framingham_df$DIABETES)
framingham_df$BPMEDS <- as.factor(framingham_df$BPMEDS)
framingham_df$CVD <- as.factor(framingham_df$CVD)
```

```r
framingham_dem <- print(CreateTableOne(data=framingham_df, strata = c("SEX"))$ContTable)

# Get blood pressure based on whether or not on BPMEDS
framingham_df$SYSBP_UT <- ifelse(framingham_df$BPMEDS == 0,
                                 framingham_df$SYSBP, 0)
framingham_df$SYSBP_T <- ifelse(framingham_df$BPMEDS == 1,
                                framingham_df$SYSBP, 0)

# Looking at risk within 15 years - remove censored data
# dim(framingham_df)
framingham_df <- framingham_df %>%
  filter(!(CVD == 0 & TIMECVD <= 365*15)) %>%
  dplyr::select(-c(TIMECVD)) %>%
  mutate(S = 1)
# dim(framingham_df)

# Filter to each sex
framingham_df_men <- framingham_df %>% filter(SEX == 1)
framingham_df_women <- framingham_df %>% filter(SEX == 2)

# The NHANES data here finds the same covariates among this national survey data

# blood pressure, demographic, bmi, smoking, and hypertension info
bpx_2017 <- nhanes("BPX_J") %>%
  dplyr::select(SEQN, BPXSY1 ) %>%
  rename(SYSBP = BPXSY1)
demo_2017 <- nhanes("DEMO_J") %>%
  dplyr::select(SEQN, RIAGENDR, RIDAGEYR) %>%
  rename(SEX = RIAGENDR, AGE = RIDAGEYR)
bmx_2017 <- nhanes("BMX_J") %>%
  dplyr::select(SEQN, BMXBMI) %>%
  rename(BMI = BMXBMI)
smq_2017 <- nhanes("SMQ_J") %>%
  mutate(CURSMOKE = case_when(SMQ040 %in% c(1,2) ~ 1,
                              SMQ040 == 3 ~ 0,
                              SMQ020 == 2 ~ 0)) %>%
  dplyr::select(SEQN, CURSMOKE)
bpq_2017 <- nhanes("BPQ_J") %>%
  mutate(BPMEDS = case_when(
    BPQ020 == 2 ~ 0,
    BPQ040A == 2 ~ 0,
    BPQ050A == 1 ~ 1,
    TRUE ~ NA )) %>%
  dplyr::select(SEQN, BPMEDS)
tchol_2017 <- nhanes("TCHOL_J") %>%
  dplyr::select(SEQN, LBXTC) %>%
  rename(TOTCHOL = LBXTC)
hdl_2017 <- nhanes("HDL_J") %>%
  dplyr::select(SEQN, LBDHDD) %>%
  rename(HDLC = LBDHDD)
diq_2017 <- nhanes("DIQ_J") %>%
  mutate(DIABETES = case_when(DIQ010 == 1 ~ 1,
                              DIQ010 %in% c(2,3) ~ 0,
```

```
                                TRUE ~ NA)) %>%
  dplyr::select(SEQN, DIABETES)

# Join data from different tables
df_2017 <- bpx_2017 %>%
  full_join(demo_2017, by = "SEQN") %>%
  full_join(bmx_2017, by = "SEQN") %>%
  full_join(hdl_2017, by = "SEQN") %>%
  full_join(smq_2017, by = "SEQN") %>%
  full_join(bpq_2017, by = "SEQN") %>%
  full_join(tchol_2017, by = "SEQN") %>%
  full_join(diq_2017, by = "SEQN") %>%
  filter(AGE < 63 & AGE > 29) #subset of data that meets the eligibility criteria for the Framingham st

df_2017$CURSMOKE <- as.factor(df_2017$CURSMOKE)
df_2017$DIABETES <- as.factor(df_2017$DIABETES)
df_2017$BPMEDS <- as.factor(df_2017$BPMEDS)

nhanes_dem <- print(CreateTableOne(data = df_2017[-1], strata = c("SEX"))$ContTable)

df_2017$SYSBP_UT <- ifelse(df_2017$BPMEDS == 0,
                           df_2017$SYSBP, 0)
df_2017$SYSBP_T <- ifelse(df_2017$BPMEDS == 1,
                          df_2017$SYSBP, 0)
framingham_dem %>%
  kable(caption = "Summary Statistics for Framingham Data")

nhanes_dem %>%
  kable(caption = "Summary Statistics for NHANES Data")
miss_var_summary(df_2017) %>%
  kable(caption = "Summary of Missing Data in NHANES")
set.seed(1234)

#data sampled from the source population
#to create test and train data sets
ignore_source_men <- sample(c(TRUE, FALSE), nrow(framingham_df_men), replace = TRUE, prob = c(0.3,0.7))
ignore_source_women <- sample(c(TRUE, FALSE), nrow(framingham_df_women), replace = TRUE, prob = c(0.3,0

framingham_df_men_train <- framingham_df_men[!ignore_source_men,]
framingham_df_men_test <- framingham_df_men[ignore_source_men,]

framingham_df_women_train <- framingham_df_women[!ignore_source_women,]
framingham_df_women_test <- framingham_df_women[ignore_source_women,]

# Fit risk score models with log transforms for all continuous variables on training set from source po
mod_men <- glm(CVD~log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                 log(SYSBP_T+1)+CURSMOKE+DIABETES,
               data= framingham_df_men_train, family= "binomial")

mod_women <- glm(CVD~log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                   log(SYSBP_T+1)+CURSMOKE+DIABETES,
                 data= framingham_df_women_train, family= "binomial")
```

```r
#Brier score on test set from source population data
#men
framingham_df_men_test$pred_probs <- predict(mod_men, newdata = framingham_df_men_test, type = "response
brier_men_source_test <- (sum((framingham_df_men_test$pred_probs - (as.numeric(framingham_df_men_test$CV

#women
framingham_df_women_test$pred_probs <- predict(mod_women, newdata = framingham_df_women_test, type = "re
brier_women_source_test <- (sum((framingham_df_women_test$pred_probs - (as.numeric(framingham_df_women_t

#table of brier scores
brier_df_source <- data.frame(
  Model = c("Men", "Women"),
  Brier_Score = c(brier_men_source_test, brier_women_source_test)
)

kable(brier_df_source, col.names = c("Model", "Brier Score"), caption = "Brier Score Estimates in the Te
set.seed(1234)

#data sampled from the target population
complete_df_2017 <- df_2017[complete.cases(df_2017),] %>%
  mutate(S = 0)

#filtering by sex
complete_df_2017_men <- complete_df_2017 %>%
  filter(SEX == 1)
complete_df_2017_women <- complete_df_2017 %>%
  filter(SEX == 2)

#combining the testing data from both the source and target populations to get the inverse odds weights
men_test_full_df <- rbind(framingham_df_men_test[-c(1,6,15)], complete_df_2017_men[-1])
women_test_full_df <- rbind(framingham_df_women_test[-c(1,6,15)], complete_df_2017_women[-1])

#models for odds of belonging to the source population

#men
men_source_lo_test <- glm(S ~ log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                  log(SYSBP_T+1)+CURSMOKE+DIABETES,
                     data = men_test_full_df, family = "binomial")
men_test_full_df$odds <- predict(men_source_lo_test, newdata = men_test_full_df, type = "response")/(1-
men_test_full_df$weights <- 1/men_test_full_df$odds #inverse odds weights
men_test_source <- men_test_full_df %>% filter(S == 1) #only source population data to fit model
men_test_source <- cbind(framingham_df_men_test$CVD, men_test_source) %>%
  rename("CVD" = "framingham_df_men_test$CVD")
men_test_source$CVD <- as.numeric(men_test_source$CVD)-1

#brier score
men_test_source$pred_probs <- predict(mod_men, newdata = men_test_source, type = "response")
men_test_source$brier_num <- (men_test_source$weights)*((men_test_source$CVD - men_test_source$pred_prob

men_brier_score <- sum(men_test_source$brier_num)/nrow(complete_df_2017_men)

#women
women_source_lo_test <- glm(S ~ log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
```

```
                  log(SYSBP_T+1)+CURSMOKE+DIABETES,
                     data = women_test_full_df, family = "binomial")
women_test_full_df$odds <- (predict(women_source_lo_test, newdata = women_test_full_df, type = "respons
women_test_full_df$weights <- 1/women_test_full_df$odds
women_test_source <- women_test_full_df %>% filter(S == 1)
women_test_source <- cbind(framingham_df_women_test$CVD, women_test_source) %>%
  rename("CVD" = "framingham_df_women_test$CVD")
women_test_source$CVD <- as.numeric(women_test_source$CVD)-1

#brier score
women_test_source$pred_probs <- predict(mod_women, newdata = women_test_source, type = "response")
women_test_source$brier_num <- (women_test_source$weights)*((women_test_source$CVD - women_test_source$

women_brier_score <- sum(women_test_source$brier_num)/nrow(complete_df_2017_women)

#table of brier scores
brier_df <- data.frame(
  Model = c("Men", "Women"),
  Brier_Score = c(men_brier_score, women_brier_score)
)

kable(brier_df, col.names = c("Model", "Brier Score"), caption = "Weighted Brier Score Estimates in the
set.seed(1234)
sim_brier_function_m <- function(sample_size){
  sim_target_male_df <- data.frame(
  SYSBP = rnorm(n = sample_size, mean = 126.07, sd = 16.63),
  AGE = rnorm(n = sample_size, mean = 47.16, sd = 9.97),
  HDLC = rnorm(n = sample_size, mean = 47.45, sd = 14.54),
  CURSMOKE = rbinom(n = sample_size, size = 1, prob = 0.259),
  BPMEDS = rbinom(n = sample_size, size = 1, prob = 0.755),
  TOTCHOL = rnorm(n = sample_size, mean = 192.86, sd = 40.71),
  DIABETES = rbinom(n = sample_size, size = 1, prob = 0.131),
  S = 0
)

sim_target_male_df$CURSMOKE <- as.factor(sim_target_male_df$CURSMOKE)
sim_target_male_df$DIABETES <- as.factor(sim_target_male_df$DIABETES)
sim_target_male_df$BPMEDS <- as.factor(sim_target_male_df$BPMEDS)

# Get blood pressure based on whether or not on BPMEDS
sim_target_male_df$SYSBP_UT <- ifelse(sim_target_male_df$BPMEDS == 0,
                                      sim_target_male_df$SYSBP, 0)
sim_target_male_df$SYSBP_T <- ifelse(sim_target_male_df$BPMEDS == 1,
                                     sim_target_male_df$SYSBP, 0)

#combining the testing data from both the source and target populations to get the inverse odds weights
men_test_full_df_sim <- rbind(framingham_df_men_test[,-c(1,2,6,11,15)], sim_target_male_df)

#models for odds of belonging to the source population

#men
men_source_lo_test_sim <- glm(S ~ log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                  log(SYSBP_T+1)+CURSMOKE+DIABETES,
```

```
                    data = men_test_full_df_sim, family = "binomial")
men_test_full_df_sim$odds <- (predict(men_source_lo_test_sim, newdata = men_test_full_df_sim, type = "r
men_test_full_df_sim$weights <- 1/men_test_full_df_sim$odds #inverse odds weights
men_test_source_sim <- men_test_full_df_sim %>% filter(S == 1) #only source population data to fit mode
men_test_source_sim <- cbind(framingham_df_men_test$CVD, men_test_source_sim) %>%
  rename("CVD" = "framingham_df_men_test$CVD")
men_test_source_sim$CVD <- as.numeric(men_test_source_sim$CVD)-1

#brier score
men_test_source_sim$pred_probs <- predict(mod_men, newdata = men_test_source_sim, type = "response")
men_test_source_sim$brier_num <- (men_test_source_sim$weights)*((men_test_source_sim$CVD - men_test_sou

men_brier_score_sim <- sum(men_test_source_sim$brier_num)/nrow(sim_target_male_df)

return(men_brier_score_sim)
}

men_brier_sim_res <- replicate(1600, sim_brier_function_m(1500))
output_men_df <- data.frame(dataset = 1:1600, bs = men_brier_sim_res)
simsum_men <- simsum(data = output_men_df, estvarname = "bs", true = men_brier_score)
rsimsum::kable(simsum_men, stats = c("thetamean", "thetamedian", "bias", "empse", "mse"), caption = "Pe
set.seed(1234)
sim_brier_function_f <- function(sample_size){
  sim_target_female_df <- data.frame(
  SYSBP = rnorm(n = sample_size, mean = 122.46, sd = 18.76),
  AGE = rnorm(n = sample_size, mean = 46.75, sd = 9.85),
  HDLC = rnorm(n = sample_size, mean = 57.59, sd = 16.25),
  CURSMOKE = rbinom(n = sample_size, size = 1, prob = 0.171),
  BPMEDS = rbinom(n = sample_size, size = 1, prob = 0.8),
  TOTCHOL = rnorm(n = sample_size, mean = 195.40, sd = 38.95),
  DIABETES = rbinom(n = sample_size, size = 1, prob = 0.11),
  S = 0
)

sim_target_female_df$CURSMOKE <- as.factor(sim_target_female_df$CURSMOKE)
sim_target_female_df$DIABETES <- as.factor(sim_target_female_df$DIABETES)
sim_target_female_df$BPMEDS <- as.factor(sim_target_female_df$BPMEDS)

# Get blood pressure based on whether or not on BPMEDS
sim_target_female_df$SYSBP_UT <- ifelse(sim_target_female_df$BPMEDS == 0,
                                 sim_target_female_df$SYSBP, 0)
sim_target_female_df$SYSBP_T <- ifelse(sim_target_female_df$BPMEDS == 1,
                                sim_target_female_df$SYSBP, 0)

#combining the testing data from both the source and target populations to get the inverse odds weights
women_test_full_df_sim <- rbind(framingham_df_women_test[,-c(1,2,6,11,15)], sim_target_female_df)

#models for odds of belonging to the source population

#women
women_source_lo_test_sim <- glm(S ~ log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                  log(SYSBP_T+1)+CURSMOKE+DIABETES,
                     data = women_test_full_df_sim, family = "binomial")
```

```r
women_test_full_df_sim$odds <- (predict(women_source_lo_test_sim, newdata = women_test_full_df_sim, type
women_test_full_df_sim$weights <- 1/women_test_full_df_sim$odds #inverse odds weights
women_test_source_sim <- women_test_full_df_sim %>% filter(S == 1) #only source population data to fit
women_test_source_sim <- cbind(framingham_df_women_test$CVD, women_test_source_sim) %>%
  rename("CVD" = "framingham_df_women_test$CVD")
women_test_source_sim$CVD <- as.numeric(women_test_source_sim$CVD)-1

#brier score
women_test_source_sim$pred_probs <- predict(mod_women, newdata = women_test_source_sim, type = "respons
women_test_source_sim$brier_num <- (women_test_source_sim$weights)*((women_test_source_sim$CVD - women_

women_brier_score_sim <- sum(women_test_source_sim$brier_num)/nrow(sim_target_female_df)

return(women_brier_score_sim)
}

women_brier_sim_res <- replicate(1600, sim_brier_function_f(1500))
output_women_df <- data.frame(dataset = 1:1600, bs = women_brier_sim_res)
simsum_women <- simsum(data = output_women_df, estvarname = "bs", true = women_brier_score)
rsimsum::kable(simsum_women, stats = c("thetamean", "thetamedian", "bias", "empse", "mse"), caption = "
#sigma matrices
cor_mat_men <- cor(complete_df_2017_men[c("SYSBP", "AGE", "HDLC", "TOTCHOL")])
cor_mat_women <- cor(complete_df_2017_women[c("SYSBP", "AGE", "HDLC", "TOTCHOL")])

set.seed(1234)
sim_brier_function_m_2 <- function(sample_size){
  mv_res <- mvrnorm(n = sample_size, mu = c(126.07, 47.16, 47.45, 192.86),
                    Sigma = cor_mat_men)
  sim_target_male_df <- data.frame(
  SYSBP = mv_res[,1],
  AGE = mv_res[,2],
  HDLC = mv_res[,3],
  TOTCHOL = mv_res[,4],
  CURSMOKE = rbinom(n = sample_size, size = 1, prob = 0.259),
  BPMEDS = rbinom(n = sample_size, size = 1, prob = 0.755),
  DIABETES = rbinom(n = sample_size, size = 1, prob = 0.131),
  S = 0
)

sim_target_male_df$CURSMOKE <- as.factor(sim_target_male_df$CURSMOKE)
sim_target_male_df$DIABETES <- as.factor(sim_target_male_df$DIABETES)
sim_target_male_df$BPMEDS <- as.factor(sim_target_male_df$BPMEDS)

# Get blood pressure based on whether or not on BPMEDS
sim_target_male_df$SYSBP_UT <- ifelse(sim_target_male_df$BPMEDS == 0,
                                      sim_target_male_df$SYSBP, 0)
sim_target_male_df$SYSBP_T <- ifelse(sim_target_male_df$BPMEDS == 1,
                                     sim_target_male_df$SYSBP, 0)

#combining the testing data from both the source and target populations to get the inverse odds weights
men_test_full_df_sim <- rbind(framingham_df_men_test[,-c(1,2,6,11,15)], sim_target_male_df)

#models for odds of belonging to the source population
```

```r
#men
men_source_lo_test_sim <- glm(S ~ HDLC + TOTCHOL + AGE + SYSBP_UT + SYSBP_T + CURSMOKE + DIABETES,
                      data = men_test_full_df_sim, family = "binomial")
men_test_full_df_sim$odds <- (predict(men_source_lo_test_sim, newdata = men_test_full_df_sim, type = "r
men_test_full_df_sim$weights <- 1/men_test_full_df_sim$odds #inverse odds weights
men_test_source_sim <- men_test_full_df_sim %>% filter(S == 1) #only source population data to fit mode
men_test_source_sim <- cbind(framingham_df_men_test$CVD, men_test_source_sim) %>%
  rename("CVD" = "framingham_df_men_test$CVD")
men_test_source_sim$CVD <- as.numeric(men_test_source_sim$CVD)-1

#brier score
men_test_source_sim$pred_probs <- predict(mod_men, newdata = men_test_source_sim, type = "response")
men_test_source_sim$brier_num <- (men_test_source_sim$weights)*((men_test_source_sim$CVD - men_test_sou

men_brier_score_sim <- sum(men_test_source_sim$brier_num)/nrow(sim_target_male_df)

return(men_brier_score_sim)
}

men_brier_sim_res_2 <- replicate(1600, sim_brier_function_m_2(1500))
output_men_df_2 <- data.frame(dataset = 1:1600, bs = men_brier_sim_res_2)
simsum_men_2 <- simsum(data = output_men_df_2, estvarname = "bs", true = men_brier_score)
rsimsum::kable(simsum_men_2, stats = c("thetamean", "thetamedian", "bias", "empse", "mse"), caption = "
set.seed(1234)
sim_brier_function_f_2 <- function(sample_size){
  mv_res <- mvrnorm(n = sample_size, mu = c(122.46, 46.75, 57.59, 195.40),
                    Sigma = cor_mat_women)
  sim_target_female_df <- data.frame(
  SYSBP = mv_res[,1],
  AGE = mv_res[,2],
  HDLC = mv_res[,3],
  TOTCHOL = mv_res[,4],
  CURSMOKE = rbinom(n = sample_size, size = 1, prob = 0.171),
  BPMEDS = rbinom(n = sample_size, size = 1, prob = 0.8),
  DIABETES = rbinom(n = sample_size, size = 1, prob = 0.11),
  S = 0
)

sim_target_female_df$CURSMOKE <- as.factor(sim_target_female_df$CURSMOKE)
sim_target_female_df$DIABETES <- as.factor(sim_target_female_df$DIABETES)
sim_target_female_df$BPMEDS <- as.factor(sim_target_female_df$BPMEDS)

# Get blood pressure based on whether or not on BPMEDS
sim_target_female_df$SYSBP_UT <- ifelse(sim_target_female_df$BPMEDS == 0,
                                 sim_target_female_df$SYSBP, 0)
sim_target_female_df$SYSBP_T <- ifelse(sim_target_female_df$BPMEDS == 1,
                                sim_target_female_df$SYSBP, 0)

#combining the testing data from both the source and target populations to get the inverse odds weights
women_test_full_df_sim <- rbind(framingham_df_women_test[,-c(1,2,6,11,15)], sim_target_female_df)

#models for odds of belonging to the source population
```

```r
#women
women_source_lo_test_sim <- glm(S ~ HDLC + TOTCHOL + AGE + SYSBP_UT + SYSBP_T + CURSMOKE + DIABETES,
                    data = women_test_full_df_sim, family = "binomial")
women_test_full_df_sim$odds <- (predict(women_source_lo_test_sim, newdata = women_test_full_df_sim, type
women_test_full_df_sim$weights <- 1/women_test_full_df_sim$odds #inverse odds weights
women_test_source_sim <- women_test_full_df_sim %>% filter(S == 1) #only source population data to fit
women_test_source_sim <- cbind(framingham_df_women_test$CVD, women_test_source_sim) %>%
  rename("CVD" = "framingham_df_women_test$CVD")
women_test_source_sim$CVD <- as.numeric(women_test_source_sim$CVD)-1

#brier score
women_test_source_sim$pred_probs <- predict(mod_women, newdata = women_test_source_sim, type = "respons
women_test_source_sim$brier_num <- (women_test_source_sim$weights)*((women_test_source_sim$CVD - women_

women_brier_score_sim <- sum(women_test_source_sim$brier_num)/nrow(sim_target_female_df)

return(women_brier_score_sim)
}

women_brier_sim_res_2 <- replicate(1600, sim_brier_function_f_2(1500))
output_women_df_2 <- data.frame(dataset = 1:1600, bs = women_brier_sim_res_2)
simsum_women_2 <- simsum(data = output_women_df_2, estvarname = "bs", true = women_brier_score)
rsimsum::kable(simsum_women_2, stats = c("thetamean", "thetamedian", "bias", "empse", "mse"), caption =
```

References

"About the National Health and Nutrition Examination Survey." Centers for Disease Control and Prevention, https://www.cdc.gov/nchs/nhanes/about_nhanes.htm.

"Cardiovascular Diseases (Cvds)." World Health Organization, www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds).

D'Agostino, Ralph B Sr et al. "General cardiovascular risk profile for use in primary care: the Framingham Heart Study." Circulation vol. 117,6 (2008): 743-53. doi:10.1161/CIRCULATIONAHA.107.699579

"Framingham Heart Study (FHS)." National Heart, Lung, and Blood Institute, https://www.nhlbi.nih.gov/science/framingham-heart-study-fhs.

Steingrimsson, Jon A et al. "Transporting a Prediction Model for Use in a New Target Population." American journal of epidemiology vol. 192,2 (2023): 296-304. doi:10.1093/aje/kwac128