Transportability Analysis of the Framingham ATP-III Prediction Model

Alitzel Serrano Laguna

2023-12-15

Abstract

This study assesses the transportability of the Framingham ATP-III model when transported from the source population on which it was developed to a new target population. Examining the performance of a model when being applied to a new target population, often with different characteristics than the source population on which the model was originally developed on, is important in understanding how well the model performs in a different setting. The transportability analysis for this model was conducted through a Monte Carlo simulation with 500 repetitions where Framingham study data was used as the source and NHANES data as the new target population. Model performance was assessed with brier score estimates using an inverse odds-weighted estimand for being in the source population. The mean brier simulation estimate for men (0.1679) is higher than for women (0.0926), with both having relatively low bias with reference to the true brier estimate for men (0.0988) and for women (0.0324). This indicates the model performs relatively well when using simulated data for a new target population.

Introduction

This project explores the transportation of a prediction model trained and evaluated in a source population to a new target population. In this project, the source population comes from the Framingham Heart Study (FHS) and the new target population comes from the National Health and Nutrition Examination Survey (NHANES) wave 2017. The FHS and NHANES represent two different populations. The FHS only recruited participants from Framingham, Massachusetts. To be eligible for this FHS at the first examination individuals had to be between the ages of 30 and 62 with no previous history of heart disease [3]. In contrast, the NHANES aims to be a representative sample of the U.S. population of all ages [2].

The prediction model for cardiovascular risk used in this project follows the models used in the General Cardiovascular Risk Profile for Use in Primary Care [1]. The model includes log transformations for continuous predictor variables of HDLC, total cholesterol, age, and systolic blood pressure measure depending on whether the participant was taking blood pressure medication. The model also includes binary predictor variables as indicators of smoking status and diabetes.

Data Overview

Table 1 below shows the overall characteristics stratified by sex for the Framingham study data. In the Framingham data, we observe differences in both outcome and predictor variables between men and women. Overall, men had a higher proportion for experiencing a CVD outcome (0.33) compared to women (0.17). Women tended to have higher total cholesterol and HDLC.Both men and women had similar age distributions. Table 2 below shows the overall characteristics stratified by sex for the NHANES data. Men tended to have

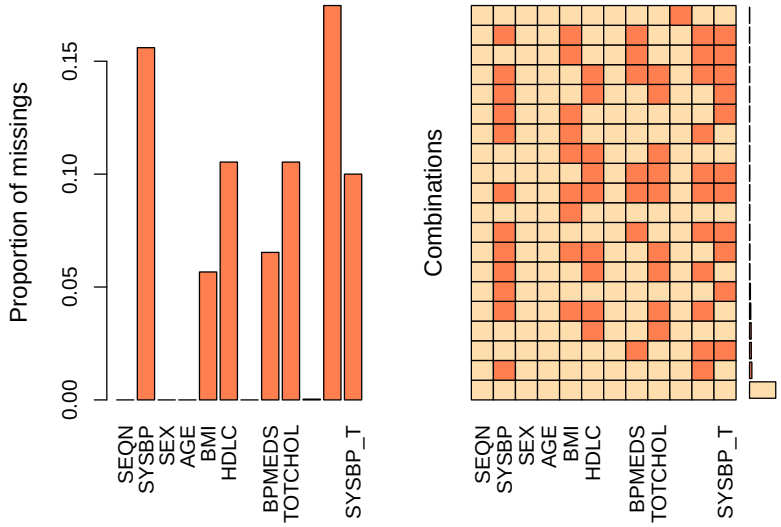
Table 1: Framingham Summary Statistics

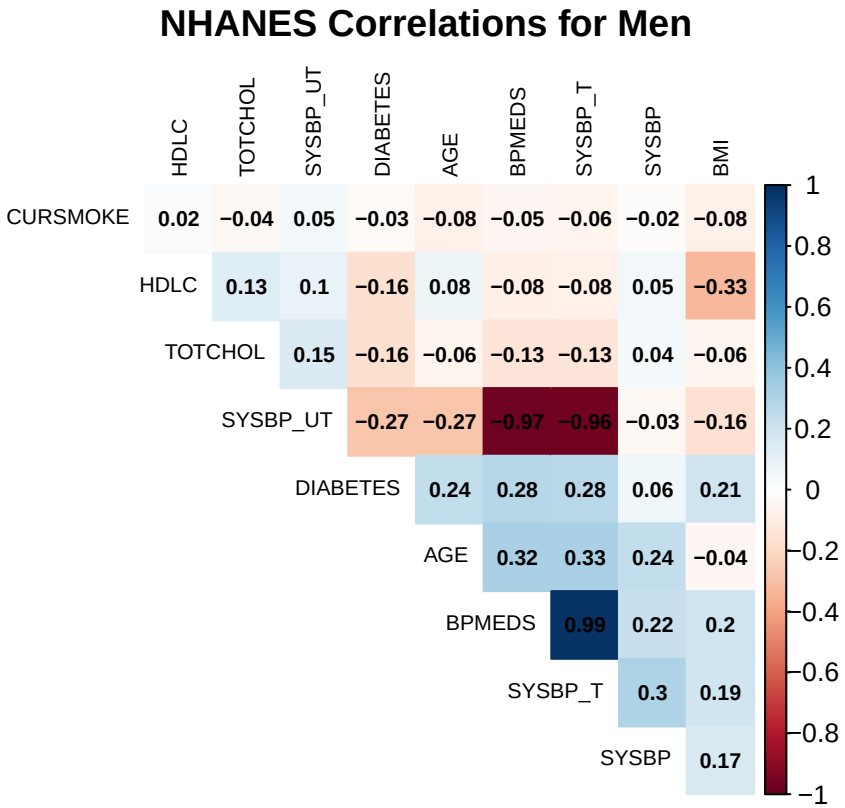
	Male	Female	p	test
n	1094	1445		
CVD (mean (SD))	0.33 (0.47)	0.17 (0.37)	<0.001	
SEX = Female (%)	0 (0.0)	1445 (100.0)	<0.001	
TOTCHOL (mean (SD))	226.44 (41.49)	246.32 (45.51)	<0.001	
AGE (mean (SD))	60.01 (8.18)	60.55 (8.40)	0.106	
SYSBP (mean (SD))	138.94 (20.89)	139.94 (23.71)	0.272	
CURSMOKE (mean (SD))	0.39 (0.49)	0.31 (0.46)	<0.001	
DIABETES (mean (SD))	0.09 (0.28)	0.07 (0.25)	0.037	
BPMEDS (mean (SD))	0.11 (0.32)	0.18 (0.38)	<0.001	
HDLC (mean (SD))	43.63 (13.37)	53.07 (15.67)	<0.001	
BMI (mean (SD))	26.25 (3.47)	25.55 (4.22)	<0.001	
SYSBP_UT (mean (SD))	121.04 (46.69)	111.49 (55.89)	<0.001	
SYSBP_T (mean (SD))	17.90 (50.93)	28.45 (61.53)	<0.001	

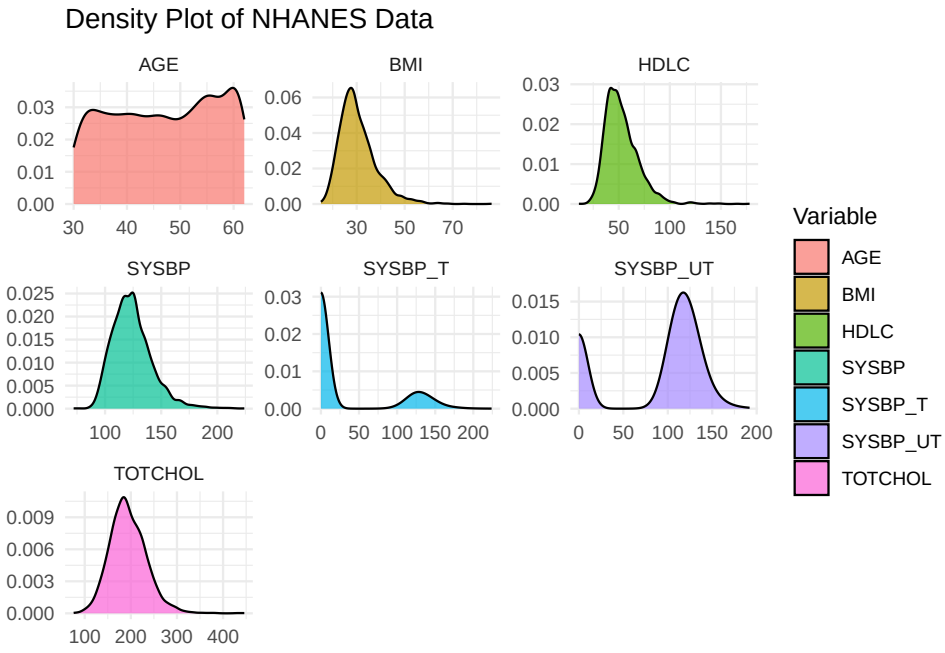
Table 2: NHANES Summary Statistics

	Male	Female	p	test
n	1417	1583		
SYSBP (mean (SD))	126.07 (16.63)	122.46 (18.76)	<0.001	
SEX = Female (%)	0 (0.0)	1583 (100.0)	<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	
CURSMOKE (mean (SD))	0.26 (0.44)	0.17 (0.38)	<0.001	
BPMEDS (mean (SD))	0.25 (0.43)	0.23 (0.42)	0.362	
TOTCHOL (mean (SD))	192.86 (40.71)	195.40 (38.95)	0.098	
DIABETES (mean (SD))	0.13 (0.34)	0.11 (0.31)	0.074	
SYSBP_UT (mean (SD))	89.01 (56.61)	86.51 (53.80)	0.260	
SYSBP_T (mean (SD))	28.95 (55.01)	26.91 (54.29)	0.333	

higher systolic blood pressure in general. The NHANES data did not include CVD outcome variables. In general, participants in the Framingham study tended to have higher mean levels of total cholesterol and systolic blood pressure. From the distribution plots of continuous variables of the NHANES data, we observe that with the exception of age, BMI, HDLC, total cholesterol, and systolic blood pressure, have a right-skewed distribution. Additionally, missingness in the NHANES dataset comes mainly from the SYSBP_UT and SYSBP variables having 17.46% and 15.6% missing data. This may be due to the procedure in which the data was collected, particularly for systolic blood pressure and blood pressure medication usage.







Model Overview

In fitting the model specified below separately for men and women in the Framingham data, we observe differences in the estimated odds of experiencing a CVD event, see Table 3 and Table 4 below. The odds of experiencing a CVD event for each increase in the log of age is slightly higher for women compared to men. Among women taking and not taking blood pressure medication, there is also an increased odds of experiencing a CVD event for each increase in the log of systolic blood pressure compared to men taking and not taking blood pressure medication. Women also have slightly higher odds of experiencing a CVD event if they are a current smoker and have diabetes compared to men. In general, all the variables included in the model are statistically significant for both men and women.

The prediction model for cardiovascular risk provided, where the outcome of interest, CVD, is an indicator of experiencing a CVD event, follows the form below:

$$CVD \sim \log(HDLC) + \log(TOTCHOL) + \log(AGE) + \log(SYSBP\_UT + 1) + \log(SYSBP\_T + 1) + \text{CURSMOKE} + \text{DIABETES}$$

From fitting the model on the Framingham data, we observe model performance from the ROC plot below. The model has an AUC of 0.759 indicating the model has good discrimination. With a threshold of 0.201, the model has a sensitivity of 0.792 meaning the model classifies less false negatives, but a lower specificity of 0.609 indicating the model results in more false positives.

Methods

For transportability analysis of the specified prediction model above, we follow the performance measures according to Steingrimsson et al[4]. Our source population is the Framingham study population, and our target population comes from the NHANES study. The outcome of interest is only observed in the Framingham study. This is referred to as a "non-nested" sampling design, where samples are obtained separately

Table 3: Model for Men

	Coefficient	Odds_Ratio	PValue	CI_Lower	CI_Upper
(Intercept)	-29.913	0.000	0.000	-38.956	-21.139
log(HDLC)	-0.602	0.547	0.017	-1.102	-0.108
log(TOTCHOL)	0.655	1.925	0.144	-0.221	1.537
log(AGE)	3.655	38.676	0.000	2.407	4.931
log(SYSBP_UT + 1)	2.589	13.316	0.000	1.417	3.784
log(SYSBP_T + 1)	2.626	13.815	0.000	1.482	3.793
CURSMOKE	0.192	1.211	0.258	-0.141	0.525
DIABETES	0.703	2.020	0.007	0.192	1.218

Table 4: Model for Women

	Coefficient	Odds_Ratio	PValue	CI_Lower	CI_Upper
(Intercept)	-32.044	0.000	0.000	-41.875	-22.609
log(HDLC)	-1.665	0.189	0.000	-2.311	-1.036
log(TOTCHOL)	0.671	1.956	0.210	-0.380	1.719
log(AGE)	4.544	94.099	0.000	2.959	6.180
log(SYSBP_UT + 1)	2.847	17.229	0.000	1.570	4.140
log(SYSBP_T + 1)	2.845	17.200	0.000	1.595	4.112
CURSMOKE	0.525	1.690	0.019	0.083	0.962
DIABETES	0.864	2.373	0.004	0.262	1.450

Table 5: Model Performance on Framingham Test Data

Metric	All	Men	Women
Sensitivity	0.792	0.786	0.650
Specificity	0.609	0.670	0.772
AUC	0.759	0.761	0.772
Best Threshold	0.201	0.327	0.186

from the source and target populations. Using these samples a composite dataset is generated, and split randomly into a test and train data. The model is built on the training data to obtain, $E[Y|X, S = 1]$, where Y is the CVD outcome, conditional on the predictor variables given that the data is in the source population indicated by S = 1. Additionally, inverse-odds weights are estimated separately for both the training and testing data to ensure independence when estimating model performance.

Model performance can be evaluated through the target population MSE given by:

$$\hat{\psi}_{\hat{\beta}} = \frac{\sum_{i=1}^{n} I(S_i = 1, D_{\text{test},i} = 1)\hat{o}(X_i)(Y_i - g_{\hat{\beta}}(X_i))^2}{\sum_{i=1}^{n} I(S_i = 0, D_{\text{test},i} = 1)} [4]$$

In the formula, $\hat{o}(X)$ is an estimator the inverse-odds weights in the test set, $\frac{\Pr[S=0|X,D_{test,i}=1]}{\Pr[S=1|X,D_{test,i}=1]}$.

The difference between the actual observed outcome and model-derived predicitions are denoted by $Y_i - g_{\hat{\beta}}(X_i)$.

Two assumptions made, (1) independence of the CVD outcome and the population S conditional on the covariates and (2) a positivity assumption for target data given the covariates based on covariate patterns from the source data.

In order to use this data for a transportability analysis, the NHANES data was limited to include participants ages 30 to 62 to match the eligibility criteria of the Framingham data. Also, due to the amount of missing data, multiple imputation was performed. Estimates were calculated and averaged across all five imputed datasets to obtain the true target population brier score.

Simulation Design

In addition to estimating model performance in the observed source Framingham data and its transportability to the new target NHANES data, we also estimate model performance in simulated target data. In this case, with having an already established model, all simulated data in addition to the framingham test data is used to evaluate model performance. We consider a simulation study following an ADEMP framework as follows:

The **AIM**, of this study is to evaluate the performance of a prediction model developed from the Framingham Heart Study source data in a new target population underlying the NHANES data. As for the **Data-Generating Mechanism**, a target population was simulated by first generating the continuous variables using the mvrnorm function from the mvtnorm package in R and transformed to follow a gamma distribution which more accurately mimics the distributions observed in the actual NHANES data. These variables include total cholesterol, systolic blood pressure, HDLC, and BMI.The parameters used to generate these variables are the mean values from the summary statististics of the NHANES dataset, and a specified covariance matrix following the covariance pattern of these variables from the Framingham data. Additionally, the variable for age had a unique distribution and was generated by sampling from a uniform distribution. Categorical variables were generated sampling from a Bernoulli distribution to achieve the proportions from the summary statistics of the NHANES data. These categorical binary variables include diabetes, blood pressure medication, and smoking status. In addition, the blood pressure medication and diabetes variables were allowed to have correlations with other continuous variables. Based on earlier observation, current smoking status was not correlated with other variables in the NHANES data, so it was excluded from this. Using this simulated data, variables for systolic bloop pressure based on blood pressure medication usage were added. Data was generated separately for men and women using the stratified mean summary statistics and Framingham data sets for each strata to get a combined dataset of n = 2,000 with equal proportions of men and women.

The evaluated model predicts the proportion of individuals who were predicted to have CVD given the X covariates in the combined dataset using the model developed from the FHS. Our **estimand** of interest is the target population brier score. **Performance measures** for the simulated data are assessed through the resulting bias from the "true" MSE estimate calculated from the non-simulated data, in addition its confidence intervals.

Table 6: True Target Brier Estimates

Data	Estimate
Men	0.0988
Women	0.0324
Combined	0.0469

Table 7: Simulation Estimates for Men (True Value = 0.0988)

Measure	Estimate	Upper	Lower
N sim	500	NA	NA
Mean	0.1679	NA	NA
Bias	0.0691	0.0656	0.0727

Simulated Data Overview

The plot above shows the correlation of the simulated data. This data follows the mean values of the actual NHANES data, and as observed in the plot, is follows the correlations observed in the NHANES data.

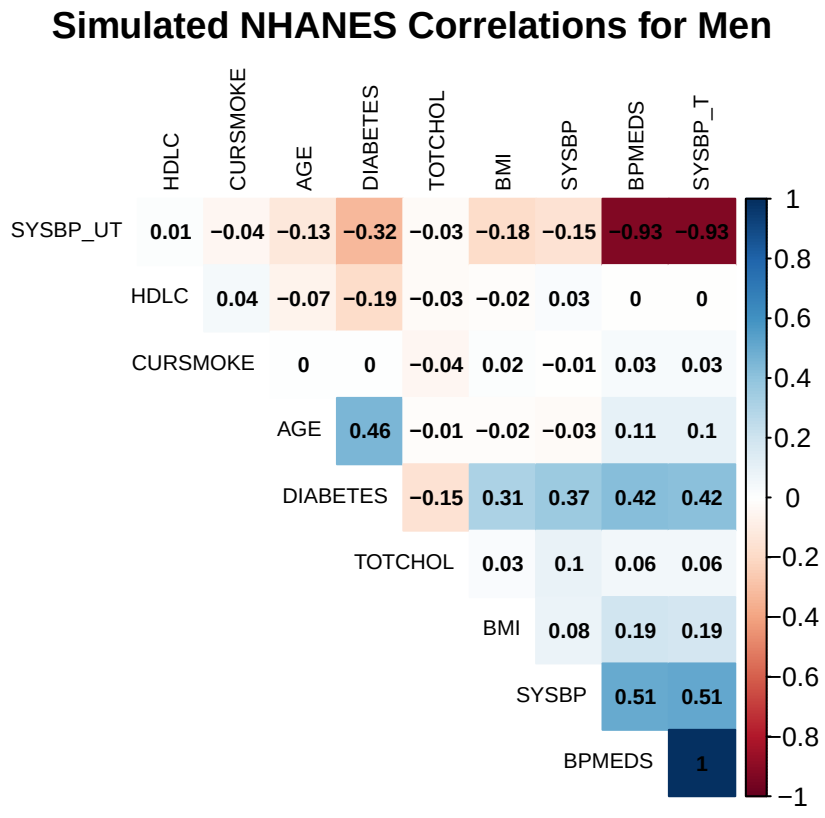


Table 8: Simulation Estimates for Women (True Value = 0.0324)

Measure	Estimate	Upper	Lower
N sim	500	NA	NA
Mean	0.0926	NA	NA
Bias	0.0602	0.0584	0.0619

Table 9: Overall Simulation Estimates (True Value = 0.0469)

Measure	Estimate	Upper	Lower
N sim	500	NA	NA
Mean	0.1084	NA	NA
Bias	0.0615	0.0598	0.0632

Results and Discussion

The brier estimates were calculated using the actual combined data from the framingham and NHANES. Table 5 below shows the results. The brier estimate for the overall combined data was 0.0404. In stratifiying by sex, men had a higher brier estimate of 0.0988 and women had a lower estimate of 0.0324. This indicates the model predictions performed slightly better for women. These values will be referred to as the "true" target brier estimates for comparison purposes in the simulation study. Tables 7 to 9 show the results for men, women, and combined simulated data with n = 500 repetitions. The simulation mean estimate was 0.2775 for men and 0.0649 for women. In reference to the "true" brier estimate, the bias for men,0.1679, was comparatively larger than the bias for women, 0.0926, across simulations. The overall mean estimate across simulations for both men and women is 0.1084.To add, in assuming only summary level statistics are available from the new target population, our data-generating mechanism was limited. In using a gamma distribution, it was difficult to achieve high correlation. This is a setting of interest that should be further looked into.

Conclusion

In this study, the performance of a predictive model was evaluated through a MC simulation using data generated separately for men and women to create a target population. The simulation involved 500 repetitions, and for each repetition the brier score as defined previously using an inverse-odds weighting estimator was calculated. Our results showed the mean estimate for men was considerably higher than the estimate for women. The bias in the simulation mean estimate was relatively low for both men and women in reference to the "true" estimated from the non-simulated data. The overall mean brier estimate across simulations for the combined data of men and women was 0.1084.

When applying the model to the simulated NHANES data, potential differences between men and women were observed. This may be due to what was previously observed when fitting the model separately for men and women on Framingham data yielded higher AIC and residual deviance values for men. Moreover, a limitation is due to our data-generating mechanism. Simulated NHANES data was generated by random sampling from a gamma distribution to match the NHANES data. A different data-generating mechanism should be considered for simulated data with higher correlations than observed. In this scenario, the model performed relatively well in the combined and separate simulated data for men and women, with better performance in women than men. In conclusion, this project highlights the possible transportability of a model developed on a source population to a new target population through simulated data, when the outcome of interest is not observed and only summary level data for the target population of interest are available.

References

Centers for Disease Control and Prevention. (2023, May 31). About NHANES. National Center for Health Statistics. https://www.cdc.gov/nchs/nhanes/about_nhanes.htm

D'Agostino, R. B. Sr, Vasan, R. S., Pencina, M. J., Wolf, P. A., Cobain, M., Massaro, J. M., & Kannel, W. B. (2008). General Cardiovascular Risk Profile for Use in Primary Care. The Framingham Heart Study. Circulation, 117, 743–753. https://doi.org/10.1161/CIRCULATIONAHA.107.699579

National Heart, Lung, and Blood Institute. (n.d.). Framingham Heart Study. National Institutes of Health. https://www.nhlbi.nih.gov/science/framingham-heart-study-fhs

Steingrimsson, J. A., Gatsonis, C., Li, B., & Dahabreh, I. J. (2023). Transporting a Prediction Model for Use in a New Target Population. American Journal of Epidemiology, 192(2), 296–304. https://doi.org/10.1093/aje/kwac128

Appendix: All code for this report

```
knitr::opts_chunk$set(echo = FALSE,
                      message = FALSE,
                      warning = FALSE)

#Load Libraries
library(riskCommunicator)
library(tidyverse)
library(tableone)
library("DescTools")
library(mice)
library(knitr)
library(kableExtra)
library(naniar)
library(nhanesA)
library(pROC)
library(VIM)
library(mice)
library(yardstick)
library(mvtnorm)
library(rsimsum)
library(corrplot)
####################################
# Code provided for framingham data
####################################

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))
```

```r
framingham_df <- na.omit(framingham_df)

#CreateTableOne(data=framingham_df, strata = c("SEX"))

# 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))
#dim(framingham_df)

framingham_df <- framingham_df %>%
  dplyr::select(-c(DIABP))

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


framingham_df$SEX <- factor(framingham_df$SEX, levels = c(1, 2), labels = c("Male", "Female"))
CreateTableOne(data = framingham_df, strata = c("SEX"))  %>% kableone(caption = "Framingham Summary Sta
                                            bootstrap_options = "striped" ,full_width = F)
framingham_df$SEX <- as.numeric(framingham_df$SEX)
#####################################
# Code provided for NHANES 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")

# Get blood pressure based on whether or not on BPMEDS
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)
#eligibility criteria
df_2017 <- df_2017 %>% filter(AGE >= 30 & AGE <= 62)

df_2017$SEX <- factor(df_2017$SEX, levels = c(1, 2), labels = c("Male", "Female"))
CreateTableOne(data = df_2017[,-c(1)], strata = c("SEX"))  %>% kableone(caption = "NHANES Summary Statis


df_2017$SEX <- as.numeric(df_2017$SEX)
#####################################
# Missing Data visualization
#####################################
#suppressMessages(library(VIM))
# Visualizing missing data
a <- (aggr(df_2017, plot = T, col=c('navajowhite','coral')))
#(aggr(df_2017, numbers=T, sortVars=TRUE, col=c('navajowhite','coral')))

#########################
# Create Correlation/ Distribution Plots
#########################

#NHANES Correlations
correlation_matrix <- cor(na.omit(df_2017 %>% filter(SEX == 1) %>% dplyr::select(!c(SEQN, SEX))))
par(mar = c(1, 1, 3, 1))
```

```r
corrplot(
  correlation_matrix,
  method = "color",
  addrect = 2,
  order = "hclust",
  type = "upper",
  number.cex = 0.7,
  tl.cex = 0.7, tl.col = "black",
  addCoef.col = "black",
  diag = FALSE,
  title = "NHANES Correlations for Men",  mar=c(1,1,2,1)
)


#Distributions for NHANES variables
data_long <- tidyr::gather(df_2017 %>% dplyr::select(!c(SEQN, BPMEDS, DIABETES, CURSMOKE, SEX)), key =

# Create a combined distribution plot using ggplot2
ggplot(data_long, aes(x = Value, fill = Variable)) +
  geom_density(alpha = 0.7) +
  labs(title = "Density Plot of NHANES Data", x = "", y = "") +
  theme_minimal() +
  facet_wrap(~ Variable, scales = "free")
#train-test split
set.seed(1)
framingham_df$ind <- 1
sample <- sample(c(TRUE, FALSE), nrow(framingham_df), replace=TRUE, prob=c(0.7,0.3))
train_f <- framingham_df[sample, ]
test_f <- framingham_df[!sample, ]


#train model
mod <- glm(CVD~log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                 log(SYSBP_T+1)+CURSMOKE+DIABETES,
      data= train_f, family= "binomial")

mod_men <- glm(CVD~log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                 log(SYSBP_T+1)+CURSMOKE+DIABETES,
      data= train_f %>% filter(SEX == 1), family= "binomial")

mod_women <- glm(CVD~log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                 log(SYSBP_T+1)+CURSMOKE+DIABETES,
      data= train_f %>% filter(SEX == 2), family= "binomial")

#model predictions on the framingham test data

brier <- function(data, model)
{
#' Calculate Brier Scores
probs_test <- predict(model, newdata = data, type = "response")
brier_df <- data.frame(CVD = as.factor(data$CVD), pred_probs = probs_test )
brier_t <- brier_class(brier_df, CVD, pred_probs)
roc_mod_t <- roc(predictor = probs_test, response = data$CVD)
```

```r
coords_output <- pROC::coords(roc_mod_t, "best")
df <- data.frame(
  Metric = c("Sensitivity", "Specificity", "AUC", "Best Threshold"),
  Value = c(
    coords_output$sensitivity,
    coords_output$specificity,
    auc(roc_mod_t),
    coords_output$threshold
  )
)

return(df)
}

plot1 <- brier(test_f, mod)
plot2 <- brier(test_f %>% filter(SEX == 1), mod_men)
plot3 <- brier(test_f %>% filter(SEX == 2), mod_women)

df <- cbind(plot1,plot2[,2],plot3[,2])
colnames(df) <- c("Metric", "All", "Men", "Women")


get_summary <- function(mod)
{
  summary_mod <- summary(mod)
# Extract coefficients, p-values, and confidence intervals
coefficients <- coef(summary_mod)[,1]
p_values <- summary_mod$coefficients[, "Pr(>|z|)"]
conf_intervals <- confint(mod)

# Calculate odds ratios
odds_ratios <- exp(coefficients)

# Create a data frame for the results
results_df <- data.frame(
  Coefficient = coefficients,
  Odds_Ratio = odds_ratios,
  PValue = p_values,
  CI_Lower = conf_intervals[, 1],
  CI_Upper = conf_intervals[, 2]
)

# Display the results
return(results_df)
}

tab_men <- get_summary(mod_men)
tab_women <- get_summary(mod_women)

tab_men <- tab_men %>% knitr::kable(full_width = F, digits = 3, caption = "Model for Men")
tab_women <- tab_women %>% knitr::kable(full_width = F, digits = 3, caption = "Model for Women")

tab_men
```

```
tab_women

knitr::kable(df, full_width = F, digits = 3, caption = "Model Performance on Framingham Test Data") %>%
####################
#impute nhanes data
####################
df_2017$BPMEDS <- as.factor(df_2017$BPMEDS)
df_2017$DIABETES <- as.factor(df_2017$DIABETES)
imp <- mice(df_2017, m = 5, maxit = 5, print = FALSE, seed = 123)
#imp$method
#indicator for source population
framingham_df$ind <- 1

imp_dat <- list()
combined_dat <- list()
combined_dat_men <- list()
combined_dat_women <- list()

for(i in 1:5)
{

imp_dat[[i]] <- complete(imp, i)
imp_dat[[i]] <- imp_dat[[i]][,c(1:10)] #impute for select variables
imp_dat[[i]] <- cbind(imp_dat[[i]], df_2017[,c("SYSBP_UT","SYSBP_T")])

imp_dat[[i]]$SYSBP_UT <- ifelse(imp_dat[[i]]$BPMEDS == 0,
                                imp_dat[[i]]$SYSBP, 0)
imp_dat[[i]]$SYSBP_T <- ifelse(imp_dat[[i]]$BPMEDS == 1,
                               imp_dat[[i]]$SYSBP, 0)
imp_dat[[i]]$ind <- 0 #indicator of being in nhanes
imp_dat[[i]]$CVD <- NA
imp_dat[[i]] <- imp_dat[[i]][,-1]   #remove seqn

  #Create a combined dataset
  combined_dat[[i]] <- merge(test_f, imp_dat[[i]], all = T)
  combined_dat[[i]]$ind <- as.factor(combined_dat[[i]]$ind)
  combined_dat[[i]]$CURSMOKE <- as.numeric(combined_dat[[i]]$CURSMOKE)
  combined_dat[[i]]$DIABETES <- as.numeric(combined_dat[[i]]$DIABETES)
  combined_dat_men[[i]] <- combined_dat[[i]] %>% filter(SEX == 1)
  combined_dat_women[[i]] <- combined_dat[[i]] %>% filter(SEX == 2)

}

#Fit models for ind with all NHANES data and the Framingham test set

brier_risk <- function(test_c, mod_com)
{

inv_w <- glm(ind ~ log(HDLC) + log(TOTCHOL) + log(AGE)+log(SYSBP_UT+1)+ log(SYSBP_T+1)+
               CURSMOKE+DIABETES,
                 family = binomial(), data = test_c)

probs <- predict(inv_w, type = "response")
```

```r
inv_odds_w <- (1-probs)/probs


y_pred <- predict(mod_com, newdata=test_c, type = "response")
sq_err <- (as.numeric(test_c$CVD) - y_pred)^2
est <- inv_odds_w[test_c$ind == 1] * sq_err[test_c$ind == 1]
est <- sum(est)/sum(test_c$ind == 0)

return(est)
}

estimates_men <- list()
estimates_women <- list()
combined_estimates <- list()
for(i in 1:5)
{
  estimates_men[[i]] <- brier_risk(combined_dat_men[[i]], mod_men)
  estimates_women[[i]] <- brier_risk(combined_dat_women[[i]], mod_women)
  combined_estimates[[i]] <- brier_risk(combined_dat[[i]], mod)
}
df <- data.frame(Data = c("Men", "Women", "Combined"), Estimate = c(mean(as.numeric(estimates_men)), mea
             mean(as.numeric(combined_estimates))))

library(MASS)

set.seed(123)  # Set seed for reproducibility

##################################################
# Function to Generate Data with Gamma/ Uniform dist
##################################################

# Values for men
vals_m <- c(
  SYSBP = 126.07,
  AGE = 47.16,
  BMI = 30.19,
  HDLC = 47.45,
  TOTCHOL = 192.86)

prop_val_m <- c(CURSMOKE = 0.26, DIABETES = 0.13, BPMEDS = 0.25)

# Standard deviations for the mean variables
sds_m <- c(
  SYSBP = 16.63,
  AGE = 9.97,
  BMI = 6.79,
  HDLC = 14.54,
  TOTCHOL = 40.71
)


#values for women
vals_w <- c(SYSBP = (122.46), AGE = (46.75), BMI = (30.77),HDLC = (57.59), TOTCHOL = (195.40))
```

```r
prop_val_w <- c(CURSMOKE = 0.17, DIABETES = 0.11, BPMEDS = 0.23)
sds_w <- c(
  SYSBP = 18.76,
  AGE = 9.85,
  BMI = 8.37,
  HDLC = 16.25,
  TOTCHOL = 38.95
)

# Generate a covariance matrix with low correlation
cor_matrix_low <- matrix(c(1, 0.2, 0.2, 0.2, 0.2,
                       0.2, 1, 0.2, 0.2, 0.2,
                       0.2, 0.2, 1, 0.1, 0.2,
                       0.2, 0.2,0.2, 1, 0.2,
                       0.2, 0.2, 0.2,0.2, 1), ncol = 5)

cor_matrix_high <- matrix(c(1, 0.8, 0.8, 0.8, 0.8,
                       0.8, 1, 0.8, 0.8, 0.8,
                       0.8, 0.8, 1, 0.8, 0.8,
                       0.8, 0.8,0.8, 1, 0.8,
                       0.8, 0.8, 0.8,0.8, 1), ncol = 5)

corr_actual_m <- cor(framingham_df %>% filter(SEX == 1) %>% dplyr::select(SYSBP,AGE,BMI,HDLC,TOTCHOL))
corr_actual_w <- cor(framingham_df %>%  filter(SEX == 2) %>% dplyr::select(SYSBP,AGE,BMI,HDLC,TOTCHOL))

#Models to inform correlation between binary variables
mod1 <- glm(DIABETES ~ log(AGE) + log(SYSBP_T + 1) + log(SYSBP_UT+1)+ log(HDLC) + log(TOTCHOL) + log(BM
#summary(mod1)

mod2 <- glm(BPMEDS ~ log(SYSBP) + log(BMI)+log(HDLC) +log(AGE) , data = framingham_df, family = "binomia
#summary(mod2)

generate_g_distr_data <- function(means, sds, corr_matrix, prop_vals)
{

min_age <- 30
max_age <- 80
n <- 1000

# covariance matrix
cov_matrix <- diag(sds) %*% chol(corr_matrix) %*% diag(sds)

# Generate continuous variables
sim_data_continuous <- mvrnorm(n, mu = means, Sigma = cov_matrix)

# gamma-dist param
shape_parameter <- 5
scale_parameter <- 1

sim_data_continuous[, "SYSBP"] <- rgamma(n, shape = shape_parameter, scale = scale_parameter) + sim_data
sim_data_continuous[, "BMI"] <- rgamma(n, shape = shape_parameter, scale = scale_parameter) + sim_data_
sim_data_continuous[, "HDLC"] <- rgamma(n, shape = shape_parameter, scale = scale_parameter) + sim_data_
sim_data_continuous[, "TOTCHOL"] <- rgamma(n, shape = shape_parameter, scale = scale_parameter) + sim_da
```

```r
# eligibility criteria for AGE
sim_data_continuous[, "AGE"] <- pmax(sim_data_continuous[, "AGE"], min_age)

# binary variables using specified proportions
sim_data_binary <- mvrnorm(n, mu = prop_vals, Sigma = diag(prop_vals * (1 - prop_vals)))

# combine data
sim_data <- cbind(sim_data_continuous, sim_data_binary)
sim_data <- as.data.frame(sim_data)

# Simulate a uniformly distributed AGE variable
sim_data$AGE <- runif(1000, min = min_age, max = max_age)

sim_data$BPMEDS <- ifelse(sim_data$BPMEDS > 0.5, 1, 0)
predicted_probs <- predict(mod2, newdata = sim_data, type = "response")

# Adjust pred probs to get the specified target proportion
adj_probs <- ifelse(predicted_probs > prop_vals[3],
                        predicted_probs * (prop_vals[3] / mean(predicted_probs)),
                        predicted_probs)
sim_data$BPMEDS <- ifelse(adj_probs > prop_vals[3], 1, 0)

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

# Ensure DIABETES remains binary
sim_data$DIABETES <- ifelse(sim_data$DIABETES > 0.5, 1, 0)
predicted_probs <- predict(mod1, newdata = sim_data, type = "response")

# Adjust pred probs to get the specified target proportion
adj_probs <- ifelse(predicted_probs > prop_vals[2],
                        predicted_probs * (prop_vals[2] / mean(predicted_probs)),
                        predicted_probs)

predicted_probs <- predict(mod2, newdata = sim_data, type = "response")

sim_data$DIABETES <- ifelse(adj_probs > prop_vals[2], 1, 0)

sim_data$CURSMOKE <- ifelse(sim_data$CURSMOKE > 0.5, 1, 0)
sim_data$ind <- rep(0,1000)
sim_data$CVD <- rep(NA,1000)

return(sim_data)

}

example <- generate_g_distr_data(vals_m, sds_m, corr_actual_m, prop_val_m)

correlation_matrix <- cor(example[,-c(11,12)])
```

```r
par(mar = c(1, 1, 3, 1))
corrplot(
  correlation_matrix,
  method = "color",
  addrect = 2,
  order = "hclust",
  type = "upper",
  number.cex = 0.7,
  tl.cex = 0.7, tl.col = "black",
  addCoef.col = "black",
  diag = FALSE,
  title = "Simulated NHANES Correlations for Men",  mar=c(1,1,2,1)
)

mc_simu <- function()
{

test_f_m <- test_f %>% filter(SEX == 1)
test_f_w <- test_f %>% filter(SEX == 2)

#data for men
dat_m_cor <- generate_g_distr_data(vals_m, sds_m, corr_actual_m, prop_val_m)
dat_m_cor$SEX <- 1
#combine with test data
dat_m_cor <- merge(test_f_m, dat_m_cor, all = T)

#data for women
dat_w_cor <- generate_g_distr_data(vals_w, sds_w, corr_actual_w, prop_val_w)
dat_w_cor$SEX <- 1

#combine with test data
dat_w_cor <- merge(test_f_w, dat_w_cor, all = T)

#both men and women
dat_m_cor2 <- generate_g_distr_data(vals_m, sds_m, corr_actual_m, prop_val_m)
dat_m_cor2$SEX <- 1
dat_w_cor2 <- generate_g_distr_data(vals_w, sds_w, corr_actual_w, prop_val_w)
dat_w_cor2$SEX <- 1
dat_all_cor <- merge(dat_m_cor2, dat_w_cor2, all = T)
#combine with test data
dat_all_cor <- merge(test_f, dat_all_cor, all = T)


#get estimates
est1 <- brier_risk(dat_m_cor, mod_men)
est2 <- brier_risk(dat_w_cor, mod_women)
est3 <- brier_risk(dat_all_cor, mod)

estimates_sim <- c(est1,est2, est3)
return(estimates_sim)

}
```

```r
# Set seed for reproducibility
set.seed(123)
sim_estimates <- replicate(500, mc_simu())

 sim_estimates <- t(sim_estimates)
 sim_estimates <- as.data.frame(sim_estimates)
 #sim_estimates
 ss1 <- data.frame(dataset = 1:500, ALL = sim_estimates[,1])
 ss2 <- data.frame(dataset = 1:500, ALL = sim_estimates[,2])
 ss3 <- data.frame(dataset = 1:500, ALL = sim_estimates[,3])
 simsum_1 <- simsum(data = ss1, estvarname = "ALL", true = 0.0988, x = F)
 simsum_2 <- simsum(data = ss2, estvarname = "ALL", true = 0.0324, x = F)
 simsum_3 <- simsum(data = ss3, estvarname = "ALL", true = 0.0469, x = F)

s1 <- summary(simsum_1)
s2 <- summary(simsum_2)
s3 <- summary(simsum_3)

tab_men <- data.frame(Measure = c("N sim", "Mean", "Bias"),
                        Estimate = c("500", round(tidy(s1)[2,2],4),
                                     round(tidy(s1)[4,2],4)),
                        Upper = c("NA","NA",round(tidy(s1)[4,4],4)),
                        Lower = c("NA","NA",round(tidy(s1)[4,5],4)))
tab_women <- data.frame(Measure = c("N sim", "Mean", "Bias"),
                        Estimate = c("500", round(tidy(s2)[2,2],4),
                                     round(tidy(s2)[4,2],4)),
                        Upper = c("NA","NA",round(tidy(s2)[4,4],4)),
                        Lower = c("NA","NA",round(tidy(s2)[4,5],4)))
tab_both <- data.frame(Measure = c("N sim", "Mean", "Bias"),
                        Estimate = c("500", round(tidy(s3)[2,2],4),
                                     round(tidy(s3)[4,2],4)),
                        Upper = c("NA","NA",round(tidy(s3)[4,4],4)),
                        Lower = c("NA","NA",round(tidy(s3)[4,5],4)))
library(kableExtra)
 df %>% knitr::kable(full_width = F, digits = 4, caption = "True Target Brier Estimates")

tab_men %>% knitr::kable(caption = "Simulation Estimates for Men (True Value = 0.0988)")
tab_women %>% knitr::kable(caption = "Simulation Estimates for Women (True Value = 0.0324)")
tab_both %>% knitr::kable(caption = "Overall Simulation Estimates (True Value = 0.0469)")
```