

Estimating Prediction Model Performance on A New Target Population: A Transportability Analysis

Destiny Rankins

Abstract

- **Background:** Transporting a prediction model can help determine how we a model performs in a new target population.
- **Methods:** A logistic regression model was trained and validated using the source population Framingham heart study data. The performance of the model was estimated in a new underlying target population from National Health and Nutrition Examination Survey data where the observed outcome from source population was not available. For the transportability analysis, an estimator the inverse-odds weights was used to estimate the Brier risk in the target population. Summary level data was used to simulate individual level data from the target population to assess how the prediction model performs on the simulated data.
- **Results:** The Brier Score obtained after assessing the performance of the logistic regression model on the source population was 0.147 The estimated Brier risk from the underlying NHANES target population was 0.103. The lowest estimated Brier risk was from the simulated target population was 0.004.
- **Conclusion:** The estimator for the Brier risk allows for measuring the performance of a prediction model on a new target population where the observed outcome is not available.

Introduction

Prediction models are proposed and developed for many real world applications in the clinical setting. Before these models are applied in clinical settings, they have to be validated for accuracy. Internal validation is done to assess the performance and help measure the accuracy of the model. This can be implemented by splitting the data sample into a training set

for model development and a test set for model validation to determine how well the model performs on unseen data. External validation is an examination of model performance in different dataset(s), often regarded as a gold-standard of model ‘credibility’ [3]. These models are often developed with data from different cohorts and include different predictor variables such as demographic predictors, health risk predictors, laboratory test results, etc. Furthermore, the outcome of interest may not be available (e.g., when outcome ascertainment requires specialized assessments) or inadequate (e.g., when the number of outcome events is small due to incomplete follow-up) in the external data sample used for validation [1]. Given that those who use these prediction models may want to validate and implement them in another population, a transportability analysis can be conducted to determine how well the prediction model performs in a new target population data sample.

Research in prediction modeling include studies where methodology for transporting models is developed and assessed. Bing et al. proposed three estimators for area under receiving operator curve (AUC) in the target population when no outcome information is available from the sample and show that the estimators are consistent and asymptotically normal and have good finite-sample performance[2]. Steingrimmson et al. demonstrated methodology for assessing transportability to a new target population where the outcome sample is not available by using inverse-odds weights for a weighting estimator for the target population mean squared error [1].

In this analysis, a prediction model’s performance in a new target population will be measured by using the inverse-odds weights to create an estimator for the Brier risk. This method will be applied to a risk score prediction model built from the Framingham heart study data (source population data sample) to estimate performance of that model in a new target population underlying the National Health and Nutrition Examination Survey (NHANES) survey data (target population data sample). The performance of the prediction model will also be measured on new target population from a simulated data sample. The source population data has the observed outcome in the sample however, the observed outcome is not available in the target population data.

Methods

Study Population

The prediction model was developed on the source population Framingham study data and internally validated. The Framingham study data used for the regression model included continuous variables for serum total cholesterol, systolic blood pressure, blood pressure based on whether or not anti-hypertensive medication was taken, high density lipoprotein cholesterol, and body mass index. The continuous variables for sex, age, diabetes, anti-hypertensive medication, and cardiovascular disease are also included. The target population from NHANES data included similar categorical and continuous variables as the Framingham study data excluding the observed outcome variable for cardiovascular disease.

Model Development

A logistic regression model (1) was developed where P was the probability of experiencing the outcome variable CVD for cardiovascular disease. This prediction model was applied after splitting the Framingham source population data into a training set and test set. Eighty percent of the Framingham data was used for the training set to build the model and twenty percent of the data was used for the test set to evaluate the performance of the model. The Brier score was used to measure the accuracy of the predicted probabilities. This measure helps to assess how well the model's predicted outcome aligns with the actual observed values. Given the focus of this project, only the aforementioned performance measure was used to evaluate the model's performance.

$$\begin{aligned} \text{logit}(P) = & \beta_0 + \beta_1 * \log(HDLC) + \beta_2 * \log(TOTCHOL) + \beta_3 * \log(AGE) + \beta_4 * \log(SYSBP_UT + 1) \\ & + \beta_5 * \log(SYSBP_T + 1) + \beta_6 * CURSMOKE + \beta_7 * DIABETES \end{aligned} \quad (1)$$

Transportability Analysis

The NHANES target population data subset had about 32% of missing data. For the analysis, a complete data set is needed. The mice() function was used to impute five complete data sets and the mean was calculated across the data sets and used for the analysis. Furthermore, the NHANES data was filtered to only include subjects that met the eligibility of the Framingham study. This subset only included individuals from age 30 to 62. This data was used as the target population and combined with the Framingham source population data frame. The combined data frame included variables that were in both studies. A population indicator variable "S" was created to denote if an individual was in the Framingham study (S=1) and the NHANES study (S=0) and used to calculate the inverse-odd weights for the Brier risk.

To perform logistic regression, the combined data set was split into a training set to develop the model to obtain the inverse-odds weights and a test set to evaluate the model's performance. The logistic regression model included covariates similar to those used to fit the logistic regression model for the Framingham data, however the "S" variable was used as the outcome to denote being in the source population or the target population. To estimate how well the model performed in the NHANES target population, the predicted probabilities were calculated using the predict function and the obtained predicted probabilities were used to obtain the Brier Score. The obtained Brier Score and inverse-odds weights was then used to calculate the estimator for Brier risk in the target population. The following weighting estimator for the target population MSE:

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

Table 1: Brier Score Measures

Population	Brier Score
Framingham Men	0.1546
Framingham Women	0.1523
Framingham Both	0.1467
NHANES Men	0.1047 *
NHANES Women	0.1046 *
NHANES	0.1035 *
Sim Target Men	0.004 †
Sim Target Women	0.0039 †
Sim Target	0.0039 †

* estimator for Brier risk in target population

† estimator for Brier risk in simulated target population

where $\hat{o}(X)$ is and estimator the inverse-odds weights in the test set, $\frac{\Pr[S=0|X, D_{test,i}=1]}{\Pr[S=1|X, D_{test,i}=1]}[1]$. This probability was used to calculate the odds of being in the source population. Steingrimssson et al. proposed the use of inverse-odds weights estimated using training set for model building and inverse-odds weights estimated using test set from estimating model performance to ensure independence between the data used to train the model and the data used to evaluate the model[1].

Simulation

Summary level data from the individual level data from the target population was simulated by using random normal distribution functions. To simulate the continuous variables from the target population, the `rnormTrunc()` function was used where $X \sim Norm(\mu, \sigma, a, b)$ where a is the minimum value and b is the maximum value. This function was used because it allows for generation of “n” random numbers from a truncated random normal distribution within a specified range or interval with a specified mean and standard deviation. The means and standard deviations for the continuous variables were obtained from the NHANES target population data summary statistics and used in the truncated random normal distribution. The maximum and minimum values for the truncated normal distribution were obtained from the NHANES 2017-2018 Data, Documentation, Codebook, and Frequencies [4]. The binary variables were generated using the `rbinom()` function where proportions from the summary statistics from the target populations were used for the probability parameter.

Results

The observed outcome variable for cardiovascular disease was only available in the source population Framingham heart study. From the Table 1: Brier Score Measures, we observe

the Brier score for the model's prediction on the source population was 0.147. The Brier risk estimate on the NHANES target population was 0.103. The Brier risk estimate on the simulated target population was 0.004. The Brier score values of 0.25 are the same as a chance and lower values of Brier score represent increased calibration of the model[5]. As Brier score decreases from 0.25 to zero, the predicted probability increasingly equals the observed probability and the calibration is improved.

Conclusion

In this analysis, a prediction model was transported to a new target population where the variable for the observed outcome was not available. To assess how the model performed in the target population, an estimate for the Brier risk was measured after combining the data set with the source population. The prediction model performance was also assessed on simulated individual level data from the target population by estimating the Brier risk. The estimate for the Brier risk (0.004) was lowest when evaluating the performance of the prediction model on the simulated target population. There are several limitations in this analysis. The prediction model was only assessed by measuring the Brier score for the source population Framingham study data and by using the estimator for the Brier risk in the target population NHANES data and the simulated data. To further measure the prediction model for accuracy and transportability, further validation methods should be employed. The simulated data frame includes continuous variables generated from the truncated random normal distribution function. This function was used because it allows for the inclusion of specified values for the maximum and minimum interval parameters, where any values outside of the interval is cut off or truncated. The minimum and maximum values for the intervals were available in the NHANES codebook however, this may not always be the case. This analysis can potentially be reproduced and improved for future studies by including additional methods for measuring the model's performance on a new target population where the outcome is not available i.e., using the three estimators for area under receiving operator curve (AUC) proposed by Bing et al. to show they are consistent, asymptotically normal and have good finite sample performance[2].

References

1. Steingrimsdottir, J. A., Constantine Gatsonis, Li, B., & Dahabreh, I. J. (2021). *Transporting a Prediction Model for Use in a New Target Population*. 192(2), 296–304. <https://doi.org/10.1093/aje/kwac128>
2. Li, B., Constantine Gatsonis, Dahabreh, I. J., & Steingrimsdottir, J. A. (2022). Estimating the area under the ROC curve when transporting a prediction model to a target population. *Biometrics*. <https://doi.org/10.1111/biom.13796>
3. Sperrin, M., Riley, R.D., Collins, G.S. *et al.* Targeted validation: validating clinical prediction models in their intended population and setting. *Diagn Progn Res* **6**, 24 (2022). <https://doi.org/10.1186/s41512-022-00136-8>
4. *NHANES Questionnaires, Datasets, and Related Documentation*. (n.d.). [Wwwn.cdc.gov. https://www.cdc.gov/nchs/nhanes/continuousnhanes/default.aspx?BeginYear=2017](https://www.cdc.gov/nchs/nhanes/continuousnhanes/default.aspx?BeginYear=2017)
5. Deppen, S. A., Blume, J. D., Aldrich, M. C., Fletcher, S. A., Massion, P. P., Walker, R. C., Chen, H. C., Speroff, T., Degesys, C. A., Pinkerman, R., Lambright, E. S., Nesbitt, J. C., Putnam, J. B., & Grogan, E. L. (2014). Predicting Lung Cancer Prior to Surgical Resection in Patients with Lung Nodules. *Journal of Thoracic Oncology*, 9(10), 1477–1484. <https://doi.org/10.1097/jto.0000000000000287>
6. <https://github.com/destinyr2/Project3.git>

Code Appendix

```
suppressPackageStartupMessages(library(riskCommunicator))
suppressPackageStartupMessages(library(tidyverse))
suppressPackageStartupMessages(library(tableone))
suppressPackageStartupMessages(library(pROC))
suppressPackageStartupMessages(library(ggplot2))
suppressPackageStartupMessages(library(DescTools))
suppressPackageStartupMessages(library(knitr))
suppressPackageStartupMessages(library(kableExtra))
suppressPackageStartupMessages(library(gridExtra))
suppressPackageStartupMessages(library(rmarkdown))
suppressPackageStartupMessages(library(tinytex))
suppressPackageStartupMessages(library(webshot))
suppressPackageStartupMessages(library(mice))
suppressPackageStartupMessages(library(EnvStats))
suppressPackageStartupMessages(library(wakefield))
suppressPackageStartupMessages(library(reporter))

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 %>% select(c(CVD, TIMECVD, SEX, TOTCHOL, AGE,
                                         SYSBP, DIABP, CURSMOKE, DIABETES, BPMEDS,
                                         HDLC, BMI))

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)) %>%
  select(-c(TIMECVD))
#dim(framingham_df)

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

# Fit models with log transforms for all continuous variables
mod_men <- glm(CVD~log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
  log(SYSBP_T+1)+CURSMOKE+DIABETES,
  data= framingham_df_men, 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, family= "binomial")

# The NHANES data here finds the same covariates among this national survey data
library(nhanesA)

# blood pressure, demographic, bmi, smoking, and hypertension info
bpx_2017 <- nhanes("BPX_J") %>%
  select(SEQN, BPXSY1 ) %>%
  rename(SYSBP = BPXSY1)
demo_2017 <- nhanes("DEMO_J") %>%
  select(SEQN, RIAGENDR, RIDAGEYR) %>%
  rename(SEX = RIAGENDR, AGE = RIDAGEYR)
bmx_2017 <- nhanes("BMX_J") %>%
  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)) %>%
  select(SEQN, CURSMOKE)
bpq_2017 <- nhanes("BPQ_J") %>%
  mutate(BPMEDS = case_when(
    BPQ020 == 2 ~ 0,

```



```

    BPQ040A == 2 ~ 0,
    BPQ050A == 1 ~ 1,
    TRUE ~ NA )) %>%
  select(SEQN, BPMEDS)
tchol_2017 <- nhanes("TCHOL_J") %>%
  select(SEQN, LBXTC) %>%
  rename(TOTCHOL = LBXTC)
hdl_2017 <- nhanes("HDL_J") %>%
  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)) %>%

  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")

#CreateTableOne(data = df_2017)

# 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)
# change categorical variable to factors in Framingham data set
framingham_df[,c(1, 2, 7, 8, 9)] <- lapply(framingham_df[,c(1, 2, 7, 8, 9)], factor)

#make this example reproducible
set.seed(1)

#use 80% of dataset as training set and 20% as test set
sample_framingham <- sample(c(TRUE, FALSE), nrow(framingham_df), replace=TRUE, prob=c(0.8,
train_framingham_df <- framingham_df[sample_framingham, ]
test_framingham_df <- framingham_df[!sample_framingham, ]

```

```

# Filter to each sex
train_framingham_df_men <- train_framingham_df %>% filter(SEX == 1)
train_framingham_df_women <- train_framingham_df %>% filter(SEX == 2)

# Train regression models on Framingham data
train_model_fram <- glm(CVD~log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)+
                        log(SYSBP_T+1)+CURSMOKE+DIABETES,
                        data= train_framingham_df, family= "binomial")

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

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

# Calculate predicted probabilities for the source test set
fram_predict_probs <- predict.glm(train_model_fram, newdata = test_framingham_df, type = "

fram_predict_probs_men <- predict.glm(train_model_fram_men, newdata = test_framingham_df,

fram_predict_probs_women <- predict.glm(train_model_fram_women, newdata = test_framingham

# Calculate Brier score for the source test set
test_brier <- BrierScore(as.numeric(as.character(test_framingham_df$CVD)), fram_predict_probs)

test_brier_men <- BrierScore(as.numeric(as.character(test_framingham_df$CVD)), fram_predict_probs_men)

test_brier_women <- BrierScore(as.numeric(as.character(test_framingham_df$CVD)), fram_predict_probs_women)
# multiple imputation on target population data
df_2017_mice_out <- mice(df_2017, m=5, pri = FALSE, seed=123)
completedf_2017 <- mice::complete(df_2017_mice_out)

# Calculate the average across imputed data sets
impdf_2017 <- completedf_2017 %>%
  group_by(SEQN) %>%
  summarise_all(mean, na.rm = TRUE)

# subset of the NHANES data that meets the eligibility of the # Framingham study
eligible_NHANES <- subset(impdf_2017, AGE >= 30 & AGE <= 62)

```



```

model_framnhanes_women <- glm(S~log(HDLC)+log(TOTCHOL)+log(AGE)+log(BMI)+
                             log(SYSBP_UT+1)+log(SYSBP_T+1)+CURSMOKE+DIABETES,
                             data= train_framnhanes_df_women, family= "binomial")

# Calculate predicted probabilities for the model
model_framnhanes_predict_probs <- predict.glm(model_framnhanes, newdata = framnhanes_test_

model_framnhanes_predict_probs_men <- predict.glm(model_framnhanes_men, newdata = framnhan

model_framnhanes_predict_probs_women <- predict.glm(model_framnhanes_women, newdata = fram

# Calculate Brier score for the model
framnhanes_brier <- BrierScore(as.numeric(as.character(framnhanes_test_df$S)), model_framn

framnhanes_brier_men <- BrierScore(as.numeric(as.character(framnhanes_test_df$S)), model_f

framnhanes_brier_women <- BrierScore(as.numeric(as.character(framnhanes_test_df$S)), model

# estimator for the inverse odds weights
inv_odds_weights <- sum(framnhanes_test_df$S==0)/sum(framnhanes_test_df$S==1)

# estimator for Brier risk in population
framnhanes_brier_estimate <- ((sum(framnhanes_test_df$S==1))*framnhanes_brier*inv_odds_w

framnhanes_brier_estimate_men <- ((sum(framnhanes_test_df$S==1))*framnhanes_brier_men*inv_

framnhanes_brier_estimate_women <- ((sum(framnhanes_test_df$S==1))*framnhanes_brier_women*
# Set seed for reproducibility
set.seed(123)

# number of simulations
simulations <- 100

# total sample size
n <- 9254

# empty data frame for simulated data
simulated_df <- data.frame()

for(i in 1:simulations){
#Age0_70 <- rnormTrunc(n, 95.38, 20.98,0,79)

```

```

#Age80 <- rnormTrunc(n, 4.61, 20.98, 80, 80)
AGE <- rnormTrunc(n, 34.33, 25.50, 0, 80)
#AGE80 <- age(n, x = 80, prob = 0.04618713, name = "Age80")
#AGE0.70 <- age(n, x = 0:70, prob = 0.9547864, name = "Age0-70")
SYSBP <- rnormTrunc(n, 121.33, 20.39, 72, 228)
SEX <- sex(n, x = c("Male", "Female"), prob = c(0.4929151, 0.5080584), name = "SEX")
BMI <- rnormTrunc(n, 26.58, 8.29, 12.3, 86.2)
HDL <- rnormTrunc(n, 53.39, 14.75, 10, 189)
CURSMOKE <- rbinom(n, size = 1, prob = 0.17)
BPMEDS <- rbinom(n, size=1, prob = 0.28)
TOTCHOL <- rnormTrunc(n, 179.89, 40.60, 76, 446)#COUNT/CUM 7638
DIABETES <- rbinom(n, size = 1, prob = 0.10)

# Combine variables into a data frame
simulation_data <- data.frame(AGE = round(AGE), SYSBP = round(SYSBP), SEX, BMI = round(BMI))

# Append the simulation data to the overall data frame
simulated_df <- rbind(simulated_df, simulation_data)
}
# change the variable for sex to binary in the simulated data frame
simulated_df$SEX <- ifelse(simulated_df$SEX == "Male", 1, 2)

# subset of the simulated target population that meets the eligibility of the Framingham study
eligible_sim_NHANES <- subset(simulated_df, AGE >= 30 & AGE <= 62)

# find variables that are common in both studies
similar_vars <- intersect(names(framingham_df), names(eligible_sim_NHANES))

# combine data set from both studies
merge_framesim <- merge(framingham_df, eligible_sim_NHANES, by = similar_vars, all = TRUE)

# S population indicator, 0 for NHANES and 1 for Framingham
merge_framesim$S <- if_else(is.na(merge_framesim$CVD), 0, 1)

# Select similar variables for data frame
same_variables <- c("SEX", "AGE", "TOTCHOL", "SYSBP", "CURSMOKE", "DIABETES", "BPMEDS",
  "HDL", "BMI", "S")

# Data frame with similar variables
combined_framesim <- merge_framesim[same_variables]
# change change categorical variables to factors in combined data frame

```

```

combined_framesim[,c(1, 5, 6, 7, 10)] <- lapply(combined_framesim[,c(1, 5, 6, 7, 10)], facto

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

#make this example reproducible
set.seed(1)

#use 80% of dataset as training set and 20% as test set
samplesim <- sample(c(TRUE, FALSE), nrow(combined_framesim), replace=TRUE, prob=c(0.8,0.2))
sim_train_df <- combined_framesim[samplesim, ]
sim_test_df <- combined_framesim[!samplesim, ]

# Filter to each sex
sim_train_df_men <- sim_train_df %>% filter(SEX == 1)
sim_train_df_women <- sim_train_df %>% filter(SEX == 2)

# model for combined data sets
model_sim <- glm(S~log(HDLC)+log(TOTCHOL)+log(AGE)+log(BMI)+
                log(SYSBP_UT+1)+log(SYSBP_T+1)+CURSMOKE+DIABETES,
                data= sim_train_df, family= "binomial")

model_sim_men <- glm(S~log(HDLC)+log(TOTCHOL)+log(AGE)+log(BMI)+
                    log(SYSBP_UT+1)+log(SYSBP_T+1)+CURSMOKE+DIABETES,
                    data= sim_train_df_men, family= "binomial")

model_sim_women <- glm(S~log(HDLC)+log(TOTCHOL)+log(AGE)+log(BMI)+
                      log(SYSBP_UT+1)+log(SYSBP_T+1)+CURSMOKE+DIABETES,
                      data= sim_train_df_women, family= "binomial")

# Calculate predicted probabilities for the model
model_sim_predict_probs <- predict.glm(model_sim, newdata = sim_test_df, type = "response")

model_sim_predict_probs_men <- predict.glm(model_sim_men, newdata = sim_test_df, type = "r

model_sim_predict_probs_women <- predict.glm(model_sim_women, newdata = sim_test_df, type

# Calculate Brier score for the model
sim_combine_brier <- BrierScore(as.numeric(as.character(sim_test_df$S)), model_sim_predict

```

```

sim_combine_brier_men <- BrierScore(as.numeric(as.character(sim_test_df$S)), model_sim_pre

sim_combine_brier_women <- BrierScore(as.numeric(as.character(sim_test_df$S)), model_sim_p

# estimator for the inverse odds weights
sim_inv_odds_weights <- sum(sim_test_df$S==0)/sum(sim_test_df$S==1)

# estimator for Brier risk in population
sim_brier_estimate <- ((sum(sim_test_df$S==1))*sim_combine_brier*sim_inv_odds_weights)/sum

sim_brier_estimate_men <- ((sum(sim_test_df$S==1))*sim_combine_brier_men*sim_inv_odds_weig

sim_brier_estimate_women <- ((sum(sim_test_df$S==1))*sim_combine_brier_women*sim_inv_odds_

# data frame for Brier Scores
brierscores <- data.frame(Population = c("Framingham Men", "Framingham Women", "Framingham

footnote(knitr::kable(brierscores, "latex", col.names = c("Population", "Brier Score"), cap

```