Examine Transportability of Cardiovascular Risk Prediction Models through A Simulation Study

Wanyi Chen

Abstract

Background: In recent years, there have been many methods developed to help transporting measures of model performance from the source population to the target population. Given these novel methods, it is important to apply and use them to investigate the model strengths.

Method: In this study, an ADEMP simulation study is conducted to evaluate the performance of a Cardiovascular Risk prediction model in a target population (NHANES) that differs from the population (Framingham) originally used for model development and/or evaluation. More specifically, the Brier score estimator for transportability analysis is used to evaluate the model performance. The average of the Brier scores for each simulated sample are obtained after the simulation and performance measures (Bias and Mean Squared Error) of the estimator are also calculated to further interpret the simulation result.

Result: The bias for the simulated data sets ranges from -0.005 to -0.05 and the MSE ranges from 0 to 0.0025. The low overall bias and MSE give strong evidence that using simulated data to conduct transportability analysis is valid way to estimate Brier scores.

Conclusion: The simulation result gives some ideas of how transportability can be affected by the varying factors such as the standard deviations and correlations of covariates in the target population. Furthermore, this simulation study shows that simulated target data can be used to conduct transportability analysis when individual-level data is not available.

Introduction

Risk prediction models are commonly used in the healthcare system. These developed models would then be deployed to help identify individuals at high risk for a specific event. The data used for development of the risk prediction models are referred to as the source study data and they are usually data collected from randomized trials or large observational studies (Steingrimsson). However, these data are often not a random sample of the target population or the population of interest. As a result, the target population and the population underlying the source data differ, and it would be difficult to figure the performance of these models in their target population. To approach this problem, there are recent methods developed to transport measures of model performance from the source population to the target population. A simulation study would be helpful in this case, since it allows the understanding of the behavior of statistical methods from the process of generating data. Therefore, the goal of this study is to conduct a transportability analysis using simulated data set, and compare it to the non-simulated transportanility analysis.

Background and Data Collection

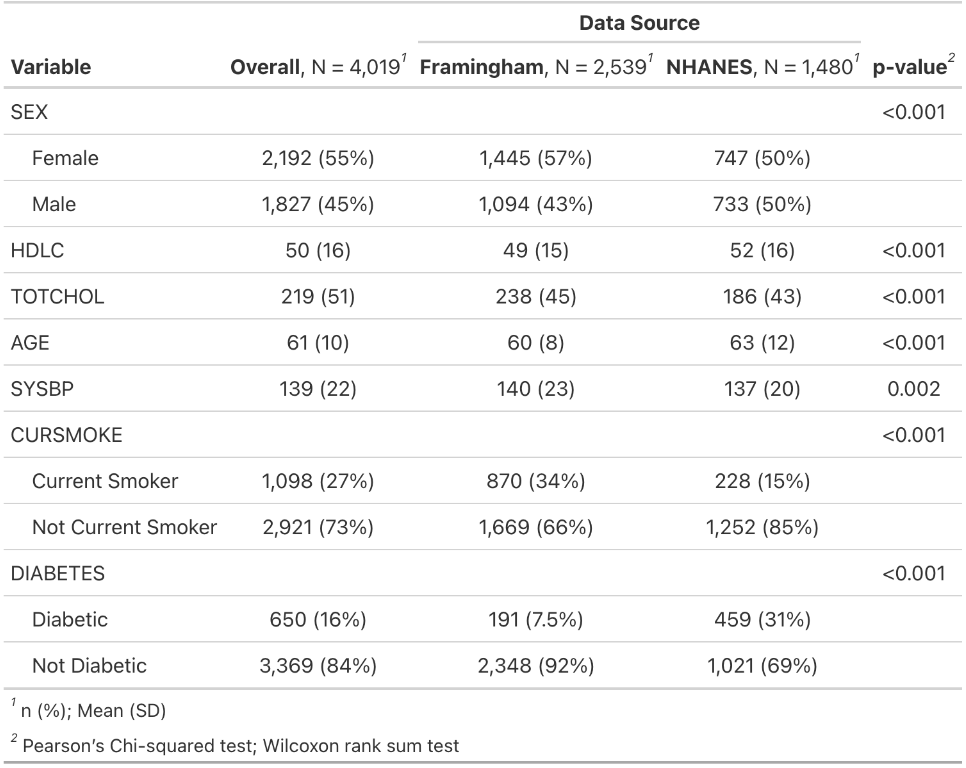
For this simulation study, the source data is obtained from the Framingham Heart Study, a long term prospective study of the etiology of cardiovascular disease among a population in Framingham, Massachusetts. The framingham data set used in this study is a subset of the data collected as part of the Framingham study and includes laboratory, clinic, questionnaire, and adjudicated event data on 4,434 participants. Data was collected during three examination periods, approximately 6 years apart, from roughly 1956 to 1968. Each participant was followed for a total of 24 years for the outcome of events: Angina Pectoris, Myocardial Infarction, Atherothrombotic Infarction, Stroke or death. The target population data is obtained from the NHANES study, and it was conducted by the National Center for Health Statistics (NCHS). In order to conduct the transportability analysis, a combined data set from Framingham study data and NHANES data is created. For the NHANES data or the target data, the outcome of interest `CVD` (cardiovascular status) is not available, only observations of complete cases are considered, and the data is subsetted so that it meets the eligibility of the Framingham study. NHANES is filtered so that all the participants are 40 years of age or above.

The Framingham data is also split into train and test sets for model derivation. The final combined set is obtained by concatenating the same variables in Framingham test set and the NHANES data. Note that NHANES data contains missing data and only complete cases are combined. NHANES data is still sufficient (observations > 1000) after removing the observations with missingness. Within the combined data set, both data include information of: `SEX` (Female = 2, Male = 1), `HDLC` (High Density Lipoprotein Cholesterol (mg/dL)), `TOTCHOL` (Serum Total Cholesterol (mg/dL)), `AGE` (Age at exam (years)), `SYSBP` (Systolic Blood Pressure (mean of last two of three measurements) (mmHg)), `CURSMOKE` (Current Smoker = 1, Not current smoker = 0), `BPMEDS` (0 = Not currently used anti-hypertension medication, 1 = Current used), and `DIABETES` (Diabetic = 1, Not diabetic = 0). The risk prediction model used in this study includes information of all these factors, and covariates `SYSBP_UT` (SYSBP if BPMEDS = 0, otherwise = 0) and `SYSBP_T` (SYSBP if BPMEDS = 1, otherwise = 0) are derived based on the status BPMEDS and SYSBP. In addition, all continuous variables are transformed using the natural log. There is also a separate model for females and males but using the same covariates and outcome of interest.

Data Summary Statistics

From Table 1, it can be seen that the sex group is fairly balanced though the p-value is statistically significant (with an alpha level of 0.05). There are similar proportions (an overall proportion of 55% female and 45% male) of female and male in combined data set, Framingham data, and NHANES data. The continuous variables seem to be similar in terms of the mean and standard deviation but the p-values are again statistically significant (with an alpha level of 0.05). The proportions of categorical variables CURSMOKE and DIABETES are not very similar and are both significantly different with p-values less than 0.05. These differences in the two data sets might imply the strength of the risk prediction model. For instance, if the distribution of an influential covariate from the target data is significantly different from the source data, the model performance can be affected. This can be examined more closely in the later simulation section.

Table 1. Summary statistics of Framingham and NHANES data sets.



		Data Source		
Variable	**Overall**, N = 4,019[1]	**Framingham**, N = 2,539[1]	**NHANES**, N = 1,480[1]	**p-value**[2]
SEX				<0.001
Female	2,192 (55%)	1,445 (57%)	747 (50%)	
Male	1,827 (45%)	1,094 (43%)	733 (50%)	
HDLC	50 (16)	49 (15)	52 (16)	<0.001
TOTCHOL	219 (51)	238 (45)	186 (43)	<0.001
AGE	61 (10)	60 (8)	63 (12)	<0.001
SYSBP	139 (22)	140 (23)	137 (20)	0.002
CURSMOKE				<0.001
Current Smoker	1,098 (27%)	870 (34%)	228 (15%)	
Not Current Smoker	2,921 (73%)	1,669 (66%)	1,252 (85%)	
DIABETES				<0.001
Diabetic	650 (16%)	191 (7.5%)	459 (31%)	
Not Diabetic	3,369 (84%)	2,348 (92%)	1,021 (69%)	

[1] n (%); Mean (SD)

[2] Pearson's Chi-squared test; Wilcoxon rank sum test

Figure 1 visualizes the 6 covariates (from Framingham data) used in the risk prediction model by gender. The bar plots of CURSMOKE and DIABETES again show the balanced number of indivdiuals in both gender groups. The distribution of HDLC is heavily right skewed due to some very high outliers but besides that, the distribution is roughly normal for both genders and centered around 50. For TOTCHOL, it is simlar to HDLC in which it also has very high outliers, but can still be roughly normal and centered around 200 without the outliers. AGE is roughly uniform and centered around 60 for both genders. SYSBP is slightly right-skewed but overall roughly normal and density centered around 140 for both genders.

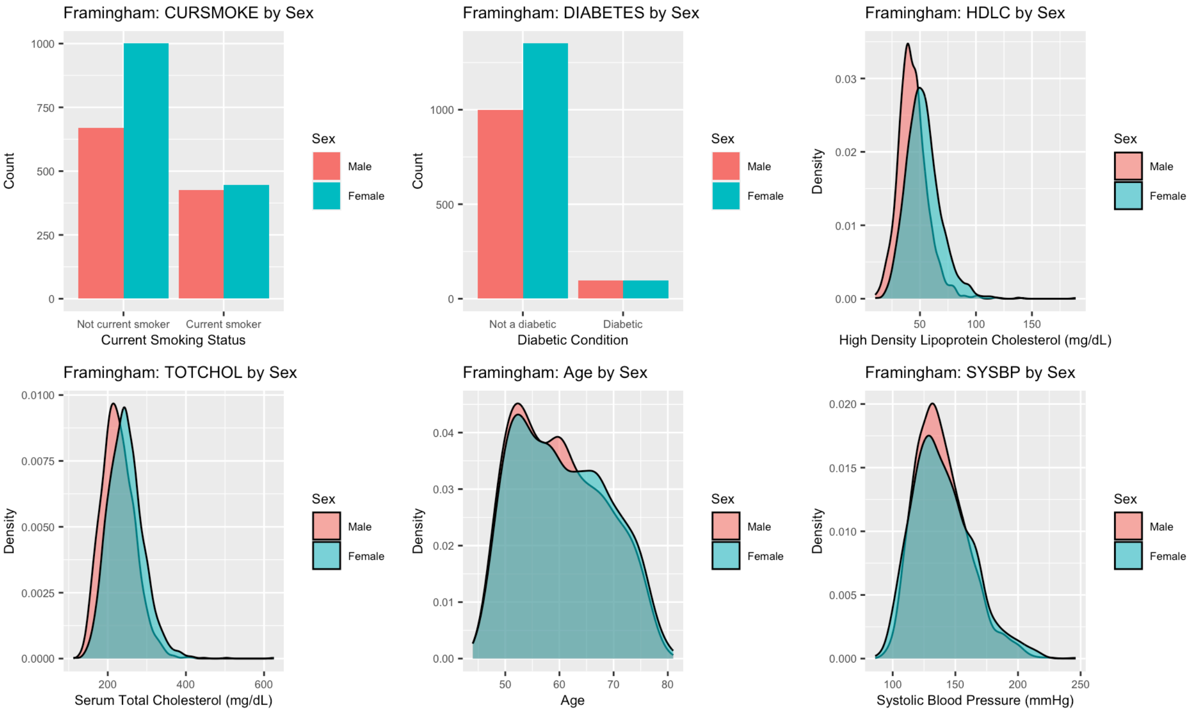


Figure 1: Visualizations of some important categorical and continuous variables in the Framingham data set.

Figure 2 also visualizes the 6 covariates used in the risk prediction model by gender but from the NHANES data. Similar to Figure 1, the bar plots of CURSMOKE and DIABETES again show the balanced number of individuals in both gender groups. The distribution of HDLC is heavily right skewed, and distributions between males and females are different which is different from HDLC in Figure 1. The male group is more skewed with more density centered around 40 while the female group is centered around 50 with a less skewed distribution. For TOTCHOL, it is similar to the TOTCHOL in Figure 1, in which it also has very high outliers, but it is roughly normal and centered around 190 without the outliers. The Age distribution here is very different from the AGE in Framingham data, that it is left skewed with most of the density centered around 60 for both sex groups. SYSBP is roughly normal and density centered around 135 for both genders. Based on the plots from Figure 1 and Figure 2, it seems like the distribution of Age is the one covariate that deviates from the source population the most. Due to this observed difference, it would be helpful to vary the distribution of AGE in the simulation study, and investigate its effect on the performance of the risk prediction model.

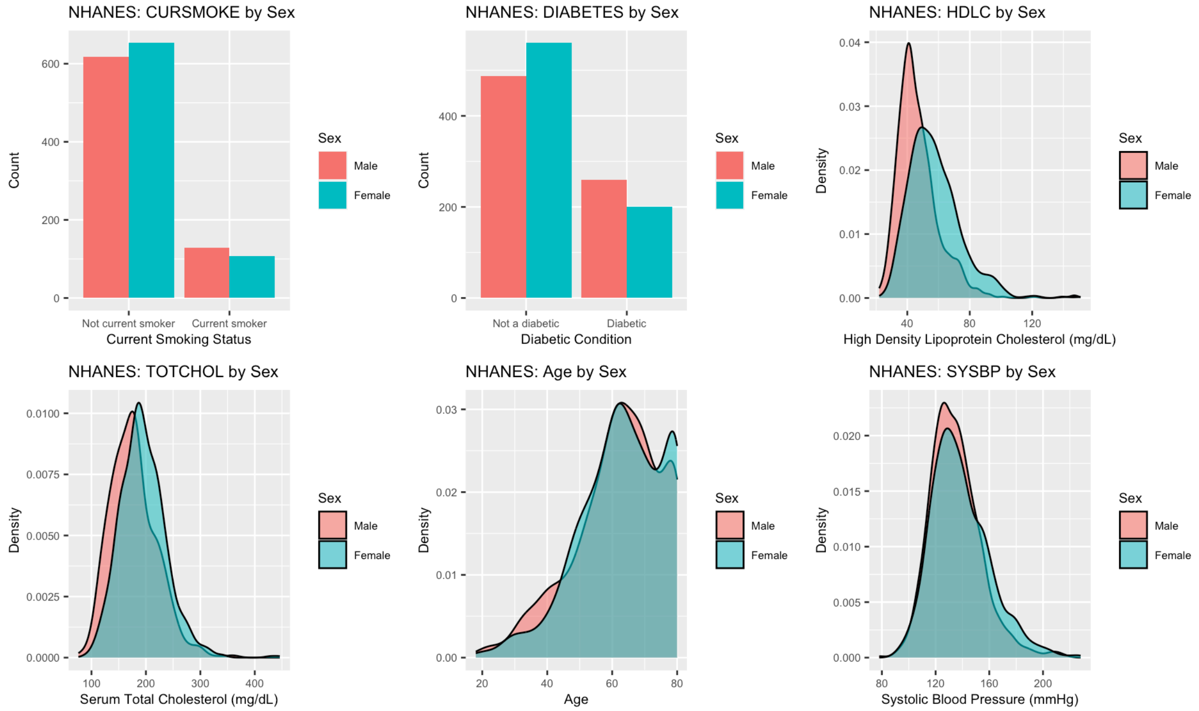


Figure 2: Visualizations of some important categorical and continuous variables in the NHANES data set.

Before diving into the simulation part, it is also important to get a sense of the actual CVD and predicted CVD obtained from the risk prediction model. From Figure 3, it can be observed that actual CVD and predicted CVD from the Framingham data are very close and similar proportions of female and males in each level (levels of CVD occurrence, did not occur = 0 and did occur = 1) are closely maintained. The NHANES predictions are not as accurate as the Framingham predictions but the overall proportion of CVD occurrence is still maintained and similar to the Framingham CVD occurrence proportions. Since we do not have CVD information from NHANES, it difficult to conclude anything regarding the model's transportability.

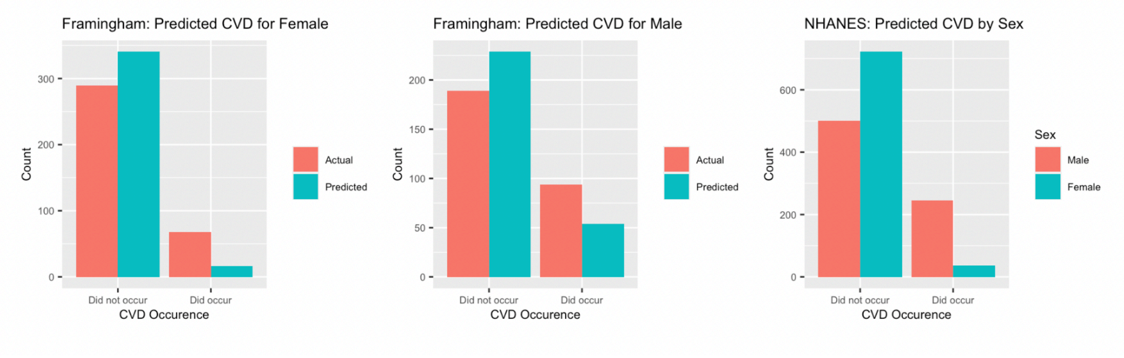
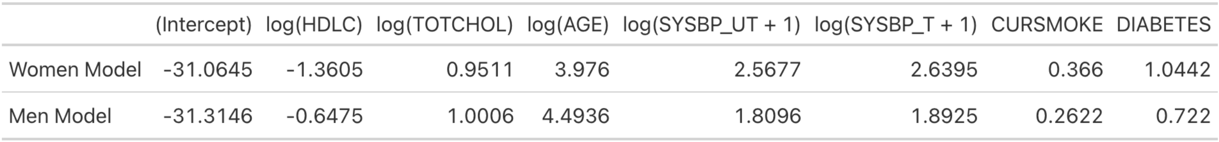


Figure 3: Bar plots for comparison between predicted CVD using the risk prediction models (men and women models) and actual CVD from Framingham data set.

The coefficient estimates for the two cardiovascular prediction models are provided in Table 2. The coefficients are very similar and the same set of covariates are used for the two models. Also note that the continuous variables are log transformed to get more appropriate outcome. Stating the model specifically help the simulation process in the later part since the simulation framework relies on this as methods.

Table 2. Coefficient estimates for Framingham ATP-III models for males and females.



	(Intercept)	log(HDLC)	log(TOTCHOL)	log(AGE)	log(SYSBP_UT + 1)	log(SYSBP_T + 1)	CURSMOKE	DIABETES
Women Model	-31.0645	-1.3605	0.9511	3.976	2.5677	2.6395	0.366	1.0442
Men Model	-31.3146	-0.6475	1.0006	4.4936	1.8096	1.8925	0.2622	0.722

Simulation

The ADEMP framework is used for this simulation study. An ADEMP framework helps researchers to design, analyze, and report simulation study accurately and effectively. ADEMP involves defining aims, data-generating mechanisms, estimands, methods, and performance measures (Morris and White et al.).

1. **Aim**

   - The aim of this simulation study is to investigate how the performance/transportability of the cardiovascular risk prediction models derived using Framingham train set can be affected by the simulated NHANES target data.

2. **Data Generation**

- Throughout the entire simulation study, seed is set to 2550. The two varying factors are: standard deviation of AGE (3, 8, 13, 18, 23) and correlation of AGE and SYSBP (0.01, 0.25, 0.5, 0.75, 0.99). The reason behind varying these factors is because the large deviation between the distributions of AGE in Framingham and NHANES data. When conducting transportability analysis, it is important to understand how these differences in the covariates can affect model performance. The overall sample size for each repetition is fixed and set to the same as the non-simulated combined data set. Since it is assumed that there is no individual-level information from the target data, the continuous variables are all randomly sampled from the normal distributions with corresponding means and standard deviations using the target data summary statistics. For the categorical variables, they are all randomly sampled by maintaining a similar proportions to that of the target data. The number of simulations is set to 1000 to keep the Monte Carlo Standard Error below 0.1.

3. **Estimand**

   - The estimand of this simulation study is the Brier Score estimator for transportability analysis.

4. **Method**

   - The method used are two cardiovascular risk prediction models for males and females. The models are generalized linear models, which assume binomial distributions for CVD. See Table 2 for more details regarding the two models.

5. **Performance Measure**

   - Bias and MSE are calculated for the Brier Scores obtained from simulation. Bias is calculated for each combination ($Bias = \frac{1}{n_{sim}} \sum_{i=1}^{n_{sim}} \widehat{Brier} - Brier$), and it indicates the amount by which $\widehat{Brier}$ exceeds the true value $Brier$ on average.

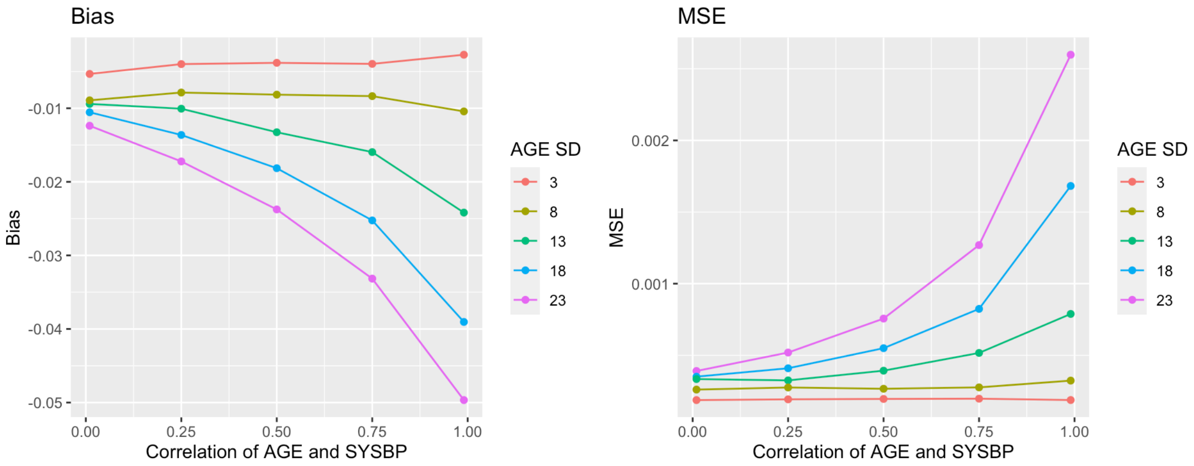


Figure 4: Performance measures (Bias and Mean Squared Error) are calculated for brier scores obtained from the simulation study.

From Figure 4, there are some obvious trends observed for the two performances measures. In both cases, the absoulte bias and MSE increases as correlation between AGE and SYSBP increases. Despite the increasing trend, bias is still relatively low and ranges from -0.005 to -0.05. MSE is also small and ranges from 0 to 0.0025. This suggests that, when conducting transportability analysis, simulated data is a valid alternative when individual-level data is not available. Furthermore, correlation does not need to be carefully considered when simulating data for transportability analysis since bias and MSE are very low in all simulated scenarios. Given the observed patterns in Figure 4, however, it would be the best to simulate data in the "lower" setting where SD and correlations of covariates are low. Table 3 gives three sets of Brier scores calculated based on the Framingham test set, Framingham test set and individual NHANES data, and Framingham test set and simulated NHANES data. Note that the simulated data here used AGE standard deviation equal 13 and low correlation of 0.25. This is to keep the similarities between the original NHANES data and the simulated data so that the Brier scores can be more comparable with the other two sets. Brier scores for the non-simulated and simulated combined data are very close which again shows that simulated data is valid alternative. The Brier scores for the women model are all smaller compared to the Brier scores for the men model. The Brier scores from Framingham test set are also larger which could be because of the smaller sample size in test set.

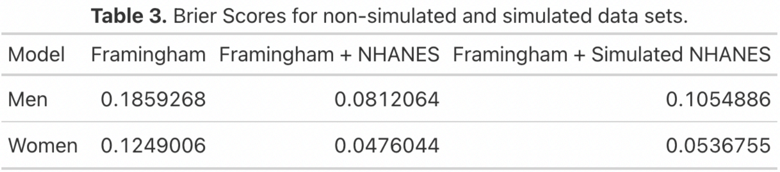


Table 3. Brier Scores for non-simulated and simulated data sets.

Model	Framingham	Framingham + NHANES	Framingham + Simulated NHANES
Men	0.1859268	0.0812064	0.1054886
Women	0.1249006	0.0476044	0.0536755

In conclusion, this simulation study gives an overview of how to apply Brier score estimator in transportability analysis and how these results can be influenced by the target study data. There are some limitations in this study that can be addressed through further research and investigation in the methods for transportability analysis. One limitation is the avaliability of data in the target population. Only about 15% of the observations in the target data are complete cases and can be used in this study. Future research with larger pool of data in both source population and target population can give a more comphresive simulation study for transportability analysis. In addition, only a few factors are being varied on this study, there might be other interesting patterns or relationships in the data that can affects the model performance. Also, there were one type of model used to test the transportability, a larger research can focus on comparing several prediction models through simulation study. Understanding the statistical power of different models can really help in advancing and deriving more robust risk prediction models.

Reference

Li B, Gatsonis C, Dahabreh IJ, Steingrimsson JA. Estimating the area under the ROC curve when transporting a prediction model to a target population. *Biometrics.* 2023;79(3):2382-2393. doi:10.1111/biom.13796

Morris, TP, White, IR, Crowther, MJ. Using simulation studies to evaluate statistical methods. *Statistics in Medicine.* 2019; 38: 2074–2102. https://doi.org/10.1002/sim.8086

Steingrimsson JA, Gatsonis C, Li B, Dahabreh IJ. Transporting a Prediction Model for Use in a New Target Population. *Am J Epidemiol.* 2023;192(2):296-304. doi:10.1093/aje/kwac128

Code Appendix

```
# read libraries
library(tidyverse)
library(dplyr)
library(ggplot2)
library(patchwork)
library(gridExtra)
library(gtsummary)
library(DescTools)
library(faux)
library(fGarch)
library(tableone)
library(gt)
```

```r
# set seed
set.seed(2550)

nhanes_df <- read.csv("df_2017.csv")[,-1]
fram_df <- read.csv("framingham_df.csv")[,-1]

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

head(nhanes_df)
dim(nhanes_df)

head(fram_df)
dim(fram_df)

nhanes_df <- nhanes_df %>%
  select(SEX, HDLC, TOTCHOL, AGE, SYSBP, SYSBP_UT, SYSBP_T, CURSMOKE, DIABETES)
nhanes_df <- nhanes_df[complete.cases(nhanes_df) == TRUE,]

combined_df <- rbind(nhanes_df, fram_df %>%
                        select(SEX, HDLC, TOTCHOL, AGE, SYSBP,
                 SYSBP_UT, SYSBP_T, CURSMOKE, DIABETES))

# Myocardial infarction (Hospitalized and silent or unrecognized), Fatal Coronary Heart Di
# Atherothrombotic infarction, Cerebral Embolism, Intracerebral Hemorrhage, or Subarachnoi
# Hemorrhage or Fatal Cerebrovascular Disease. 0 = Did not occur during followup, 1 = Did
# during followup.

p1 <- ggplot(fram_df, aes(fill = as.factor(SEX), x = as.factor(CVD))) +
  geom_bar(position = "dodge") +
  labs(title = "Framingham: CVD by Sex", x = "CVD Occurence", y = "Count",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  scale_x_discrete(labels = c("Did not occur", "Did occur")) +
  theme(text = element_text(size=8))

# High Density Lipoprotein Cholesterol (mg/dL). Available for Period 3 only.
# Values range from 10-189.
```

```
p2 <- ggplot(fram_df, aes(fill = as.factor(SEX), x = HDLC)) +
  geom_density(alpha = 0.6) +
  labs(title = "Framingham: HDLC by Sex",
       x = "High Density Lipoprotein Cholesterol (mg/dL)",
       y = "Density",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  theme(text = element_text(size=8))

# Serum Total Cholesterol (mg/dL). Values range from 107-696.
p3 <- ggplot(fram_df, aes(fill = as.factor(SEX), x = TOTCHOL)) +
  geom_density(alpha = 0.6) +
  labs(title = "Framingham: TOTCHOL by Sex",
       x = "Serum Total Cholesterol (mg/dL)",
       y = "Density",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  theme(text = element_text(size=8))

# Age at exam (years). Values range from 32-81.
p4 <- ggplot(fram_df, aes(fill = as.factor(SEX), x = AGE)) +
  geom_density(alpha = 0.6) +
  labs(title = "Framingham: Age by Sex",
       x = "Age",
       y = "Density",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  theme(text = element_text(size=8))

# Systolic Blood Pressure (mean of last two of three measurements) (mmHg).
# Values range from 83.5-295.
p5 <- ggplot(fram_df, aes(fill = as.factor(SEX), x = SYSBP)) +
  geom_density(alpha = 0.6) +
  labs(title = "Framingham: SYSBP by Sex",
       x = "Systolic Blood Pressure (mmHg)",
       y = "Density",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  theme(text = element_text(size=8))

# Current cigarette smoking at exam. 0 = Not current smoker (n = 6598),
# 1 = Current smoker (n = 5029).
p6 <- ggplot(fram_df, aes(fill = as.factor(SEX), x = as.factor(CURSMOKE))) +
  geom_bar(position = "dodge") +
  labs(title = "Framingham: CURSMOKE by Sex", x = "Current Smoking Status",
       y = "Count",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  scale_x_discrete(labels = c("Not current smoker", "Current smoker")) +
  theme(text = element_text(size=8))

# Diabetic according to criteria of first exam treated or first exam with
# casual glucose of 200 mg/dL or more. 0 = Not a diabetic (n = 11097),
# 1 = Diabetic (n = 530)
p7 <- ggplot(fram_df, aes(fill = as.factor(SEX), x = as.factor(DIABETES))) +
  geom_bar(position = "dodge") +
  labs(title = "Framingham: DIABETES by Sex", x = "Diabetic Condition",
       y = "Count",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  scale_x_discrete(labels = c("Not a diabetic", "Diabetic")) +
  theme(text = element_text(size=8))

p6 + p7 + p2 + p3 + p4 + p5


# High Density Lipoprotein Cholesterol (mg/dL). Available for Period 3 only.
# Values range from 10-189.
p8 <- ggplot(nhanes_df, aes(fill = as.factor(SEX), x = HDLC)) +
  geom_density(alpha = 0.6) +
  labs(title = "NHANES: HDLC by Sex",
       x = "High Density Lipoprotein Cholesterol (mg/dL)",
       y = "Density",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  theme(text = element_text(size=8))

# Serum Total Cholesterol (mg/dL). Values range from 107-696.
p9 <- ggplot(nhanes_df, aes(fill = as.factor(SEX), x = TOTCHOL)) +
  geom_density(alpha = 0.6) +
  labs(title = "NHANES: TOTCHOL by Sex",
       x = "Serum Total Cholesterol (mg/dL)",
       y = "Density",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  theme(text = element_text(size=8))

# Age at exam (years). Values range from 32-81.
p10 <- ggplot(nhanes_df, aes(fill = as.factor(SEX), x = AGE)) +
  geom_density(alpha = 0.6) +
  labs(title = "NHANES: Age by Sex",
       x = "Age",
       y = "Density",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  theme(text = element_text(size=8))

# Systolic Blood Pressure (mean of last two of three measurements) (mmHg).
# Values range from 83.5-295.
p11 <- ggplot(nhanes_df, aes(fill = as.factor(SEX), x = SYSBP)) +
  geom_density(alpha = 0.6) +
  labs(title = "NHANES: SYSBP by Sex",
       x = "Systolic Blood Pressure (mmHg)",
       y = "Density",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  theme(text = element_text(size=8))

# Current cigarette smoking at exam. 0 = Not current smoker (n = 6598),
# 1 = Current smoker (n = 5029).
p12 <- ggplot(nhanes_df, aes(fill = as.factor(SEX), x = as.factor(CURSMOKE))) +
  geom_bar(position = "dodge") +
  labs(title = "NHANES: CURSMOKE by Sex", x = "Current Smoking Status",
       y = "Count",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  scale_x_discrete(labels = c("Not current smoker", "Current smoker")) +
  theme(text = element_text(size=8))

# Diabetic according to criteria of first exam treated or first exam with
# casual glucose of 200 mg/dL or more. 0 = Not a diabetic (n = 11097),
# 1 = Diabetic (n = 530)
p13 <- ggplot(nhanes_df, aes(fill = as.factor(SEX), x = as.factor(DIABETES))) +
  geom_bar(position = "dodge") +
  labs(title = "NHANES: DIABETES by Sex", x = "Diabetic Condition",
       y = "Count",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  scale_x_discrete(labels = c("Not a diabetic", "Diabetic")) +
  theme(text = element_text(size=8))

p12 + p13 + p8 + p9 + p10 + p11

p14 <-  ggplot(fram_preds_df, aes(fill = as.factor(SEX), x = as.factor(CVD))) +
  geom_bar(position = "dodge") +
  labs(title = "Framingham: Predicted CVD by Sex", x = "CVD Occurence", y = "Count",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  scale_x_discrete(labels = c("Did not occur", "Did occur")) +
  theme(text = element_text(size=8))

p15 <-  ggplot(nhanes_preds_df, aes(fill = as.factor(SEX), x = as.factor(CVD))) +
  geom_bar(position = "dodge") +
  labs(title = "NHANES: Predicted CVD by Sex", x = "CVD Occurence", y = "Count",
       fill = "Sex") +
  scale_fill_discrete(labels = c("Male", "Female")) +
  scale_x_discrete(labels = c("Did not occur", "Did occur")) +
  theme(text = element_text(size=8))

p1 + p14 + p15


# combined_df$SEX <- as.factor(combined_df$SEX)
# combined_df$CURSMOKE <- as.factor(combined_df$CURSMOKE)
# combined_df$DIABETES <- as.factor(combined_df$DIABETES)
table1 <- combined_df %>%
  select(SOURCE, SEX, HDLC, TOTCHOL, AGE, SYSBP, CURSMOKE, DIABETES) %>%
  tbl_summary(by = SOURCE,
              statistic = list(
              all_continuous() ~ "{mean} ({sd})",
                all_categorical() ~ "{n} ({p}%)"
              ),
              missing_text = "(Missing)") %>%
  add_p(pvalue_fun = ~ style_pvalue(.x, digits = 2)) %>%
```

```r
  modify_header(label ~ "**Variable**") %>%
  add_overall() %>%
  modify_spanning_header(c("stat_1", "stat_2") ~ "**Data Source**") %>%
  modify_caption("**Table 1. Summary Statistics of Framingham and NHANES Data**")

gt::gtsave(as_gt(table1), file = "project3_table1.png")


set.seed(2550)
test_ind <- sample(c(0, 1), nrow(combined_df), replace=TRUE, prob=c(0.7,0.3))

combined_df$SOURCE <- c(rep(0, 1506), rep(1, 2539))
combined_df$D <- test_ind
combined_df$CVD <- c(rep(NA, 1506), fram_df$CVD)

head(combined_df)
brier_est(combined_df)

nhanes_df <- nhanes_df %>%
  mutate(ID = rep(1:1506))
nhanes_women <- nhanes_df %>%
  filter(SEX == 2)
nhanes_men <- nhanes_df %>%
  filter(SEX == 1)

fram_df <- fram_df %>%
  mutate(ID = rep(1507:4045))
fram_women <- fram_df %>%
  filter(SEX == 2)
fram_men <- fram_df %>%
  filter(SEX == 1)

nhanes_women$pred <- predict(mod_women, nhanes_women, type = "response")
nhanes_men$pred <- predict(mod_men, nhanes_men, type = "response")
fram_women$pred <- predict(mod_women,fram_women, type = "response")
fram_men$pred <- predict(mod_men, fram_men, type = "response")

nhanes_preds <- rbind(nhanes_women, nhanes_men)
nhanes_df <- left_join(nhanes_df, nhanes_preds %>% select(ID, pred), by = "ID")
fram_preds <- rbind(fram_women, fram_men)
fram_df <- left_join(fram_df, fram_preds %>% select(ID, pred), by = "ID")
```

```r
preds_df <- rbind(nhanes_df %>% select(ID, pred), fram_df %>% select(ID, pred))
combined_df$pred <- preds_df[,2]



brier_est <- function(df){
  #' @description brier score estimator for transportability analysis when target
  #' population does not include outcome of interest
  #' @param df a combined dataframe of target and source data
  #' @return brier score for the combined data

  # prediction model
  mod <- glm(SOURCE ~ D + log(abs(HDLC)) + log(abs(TOTCHOL)) + log(abs(AGE)) +
               log(abs(SYSBP_UT+1)) + log(abs(SYSBP_T+1)) + CURSMOKE + DIABETES,
      data= df, family= "binomial")

  df$o_hat <- 1/predict(mod, type = "response") # inverse odds weight
  df_temp <- df[df$SOURCE == 1 & df$D == 1, ]
  # brier risk estimate
  score <- sum(df_temp$o_hat*(df_temp$CVD - df_temp$pred)^2) /
    sum(df$SOURCE == 0 & df$D == 1)

  return(score)

}


brier_est(combined_df)
#  0.3183978
# 0.3912489



# simulation
set.seed(2550)

n_target <- c(150, 250, 500, 750, 900)
age_mean <- c(32, 42, 52, 62, 72)
n_source <- 1000 - n_target

# create function for simulation
```

```r
sim_fun <- function(){
  #' @description a simulation function that simulate a new combined data set
  #' with sample size of 1000 based on varying n_target and age_mean

  # create df for results
  res_df = data.frame(n_target = numeric(0),
                      mean_age = numeric(0),
                      b_score = numeric(0))

  # loop through the varying factors and obtain brier for each df
  for (i in 1:length(n_target)){
    for (j in 1:length(age_mean)){
      SEX <- sample(c(1, 2), n_target[i], replace=TRUE, prob=c(0.5,0.5)) # SEX
      HDLC <- rsnorm(n_target[i], 52, 16, xi = 3) # HDLC
      TOTCHOL <- rnorm(n_target[i], 186, 43) # TOTCHOL
      AGE <- abs(rnorm(n_target[i], age_mean[j], 13)) # AGE
      BPMEDS <- sample(c(0, 1), n_target[i], replace=TRUE, prob=c(0.14, 0.86)) # BPMEDS
      SYSBP <- rnorm(n_target[i], 137, 20) # SYSBP
      CURSMOKE <- sample(c(0, 1), n_target[i], replace=TRUE, prob=c(0.84,0.16)) # CURSMOKE
      DIABETES <- sample(c(0, 1), n_target[i], replace=TRUE, prob=c(0.7,0.3)) # DIABETES
      CVD <- rep(NA, n_target[i])
      SOURCE <- rep(0, n_target[i])
      target_df <- data.frame(cbind(SEX, HDLC, TOTCHOL, AGE, SYSBP, BPMEDS, CURSMOKE,
                                    DIABETES, CVD, SOURCE))
      target_df$SYSBP_UT <- ifelse(target_df$BPMEDS == 0,
                                       target_df$SYSBP, 0)
      target_df$SYSBP_T <- ifelse(target_df$BPMEDS == 1,
                                      target_df$SYSBP, 0)
      target_df <- target_df %>% select(-c(SYSBP, BPMEDS)) %>%
        relocate(SYSBP_UT, .after = AGE) %>%
        relocate(SYSBP_T, .after = SYSBP_UT)

      fram_ind <- sample(1:2359, n_source[i], replace = FALSE)
      source_df <- fram_df[fram_ind,] %>%
        select(SEX, HDLC, TOTCHOL, AGE, SYSBP_UT, SYSBP_T, CURSMOKE, DIABETES, CVD)
      source_df$SOURCE <- rep(1, n_source[i])

      sim_df <- rbind(target_df, source_df)
      sim_df$ID <- rep(1:1000)

      sim_df <- rbind(target_df, source_df)
      sim_df$ID <- rep(1:1000)

      sim_women <- sim_df %>% filter(SEX == 2)
      sim_men <- sim_df %>% filter(SEX == 1)
      sim_women$pred <- predict(mod_women, sim_women, type = "response")
      sim_men$pred <- predict(mod_men, sim_men, type = "response")
      sim_preds <- rbind(sim_men, sim_women)
      sim_df <- left_join(sim_df, sim_preds %>% select(ID, pred), by = "ID")

      sim_df$D <- sample(c(0, 1), nrow(sim_df), replace=TRUE, prob=c(0.7, 0.3))

      # bind result
      res_df[dim(res_df)[1]+1,] <- c(n_target[i], age_mean[j],
                                     brier_est(sim_df))

    }
  }
  return(res_df)
}



# repeat 1000 times
n_sim <- 2000

sim_res <- replicate(n_sim, sim_fun())
sim_res_df <- data.frame(n_target = numeric(0),
                      mean_age = numeric(0),
                      b_score = numeric(0))
# bind result into one dataframe
for (i in 1:n_sim){
  sim_res_df <- rbind(as.data.frame(sim_res[, i]), sim_res_df)
}

sim_res_df <- sim_res_df %>%
  arrange(n_target, mean_age)

performance <- function(df){
  #' @param df a dataframe
  #' @return a list of calculated performance measures
```

```
    # mean beta
    mean.est.brier <- mean(df$b_score)
    # real beta and variance
    real_b <- 0.3171372
    n_target <- df$n_target[1]
    mean_age <- df$mean_age[1]

    # bias
    bias <- sum(df$b_score -  real_b)/n_sim
    mc.bias.se <- sqrt(sum((df$b_score -  mean.est.brier)^2)/(n_sim*(n_sim-1)))

    # empse
    empse <- sqrt(sum((df$b_score -  mean.est.brier)^2)/(n_sim-1))
    mc.empse.se <- empse/sqrt(2*(n_sim-1))

    # MSE
    MSE <- sum((df$b_score - real_b)^2)/n_sim
    mc.MSE.se <- sqrt(sum((df$b_score - real_b)^2 -
                            MSE)^2 / (n_sim*(n_sim-1)))



  res <- cbind(n_target, mean_age, bias, mc.bias.se, empse, mc.empse.se, MSE,
               mc.MSE.se)
  return(res)
}

 sim_res_df <- sim_res_df %>%
     group_split(grp = as.integer(gl(25, n_sim, 2500)), .keep = FALSE)

# bind performance measures results
perf_df <- data.frame(bias = numeric(0),
                      bias.se = numeric(0),
                      empse = numeric(0),
                      empse.se = numeric(0),
                      mse = numeric(0),
                      mse.se = numeric(0))
for (i in 1:25){
  perf_df <- rbind(performance(sim_res_df[[i]]), perf_df)
}
```

```
# plot performance measures
p1 <- ggplot(perf_df) +
  geom_line(aes(x = n_target, y = bias,
                color = as.factor(mean_age))) +
  geom_point(aes(x = n_target, y = bias,
                color = as.factor(mean_age))) +
  labs(x = "Sample Size of Target Data", y = "Bias Alpha", title = "Bias", color = "Mean A

p2 <- ggplot(perf_df) +
  geom_line(aes(x = n_target, y = empse,
                color = as.factor(mean_age))) +
  geom_point(aes(x = n_target, y = empse,
                color = as.factor(mean_age))) +
  labs(x = "Sample Size of Target Data", y = "EmpSE Alpha", title = "EmpSE", color = "Mean

p3 <- ggplot(perf_df) +
  geom_line(aes(x = n_target, y = MSE,
                color = as.factor(mean_age))) +
  geom_point(aes(x = n_target, y = MSE,
                color = as.factor(mean_age))) +
  labs(x = "Sample Size of Target Data", y = "MSE Alpha", title = "MSE", color = "Mean Age

p1 + p2 / p3
```