Assessing the Transportability of Cardiovascular Risk Prediction Models from the Framingham Heart Study to Target Population

Abstract

Objective : This study evaluates the transportability of cardiovascular disease prediction models using simulated target populations and compares the result with the result from the transportability using actual dataset. Utilizing data from the Framingham Heart Study and NHANES (National Health and Nutrition Examination Survey), we explore how variations in data distribution and correlation affect model performance, specifically in terms of MSE/Brier Score, relative bias, and empirical standard error (SE).

Method : Our approach includes a full-factorial simulation study assessing three design factors: the number of correlated columns, correlation scores, and the underlying data distribution. We explore two levels of correlation among continuous data columns and two data distribution shapes – the Framingham distribution and a normal distribution. The study also considers the number of observations and statistical summary from NHANES to simulate datasets, alongside the cardiovascular disease model developed from the Framingham data.

Result Key findings indicate that datasets with minimal correlation and lognormal distributions yield the best outcomes, closely aligning with source population characteristics. This is evidenced by the smallest MSE, low relative bias, and empirical SE, suggesting the model's ability to produce consistent and reliable estimates. The study also reveals that models perform better when adhering to the source population's original distribution, and overestimating correlations tends to worsen performance metrics.

1. Introduction

Predictive models are typically designed to generate predictions for a specific demographic. For instance, a healthcare system may develop a risk prediction model to identify patients at high risk for cardiovascular events among its care recipients. The data informing the development of such models, known as source study data, often come from controlled experiments, extensive observational databases, or longitudinal studies. However, these datasets usually do not represent a random sampling from the intended demographic, resulting in a discrepancy between the target population and the one represented in the source study data. Take, for example, the widely used Framingham ATP-III model from Wilson(1998), which predicts the ten-year risk of cardiovascular events; it was developed using data primarily from white subjects, which may lead to suboptimal predictions for ethnically diverse populations.

Recently, a variety of techniques have been developed to assess the effectiveness of predictive models within a specified target population (or to adapt model performance metrics from the source to the target population) such as transportability analysis in Steingrimsson (2023). Our aim is to apply these methods to a risk score model created with data from the Framingham Heart Study, and then to evaluate the model's performance using a simulated study population drawn from the NHANES (National Health and Nutrition Examination Survey) data. Our objective is to compare the transportability result based on the existng data with result from simulation based approach.

2. Data

n this study, we reference two datasets: 1) The Framingham Heart Study, and 2) The National Health and Nutrition Examination Survey Data (NHANES).

The Framingham Heart Study (Wilson, 1998) is a long-term, prospective investigation into the causes of cardiovascular disease within a cohort of free-living individuals in Framingham, Massachusetts. This dataset includes clinical examination data such as cardiovascular disease risk factors and indicators like blood pressure. Additionally, it documents whether the participants have experienced cardiovascular events, including myocardial infarction (hospitalized and silent or unrecognized), fatal coronary heart disease, atherothrombotic infarction, cerebral embolism, intracerebral hemorrhage, subarachnoid hemorrhage, or fatal cerebrovascular disease. We regard this dataset as the source population, which we use to construct a model to identify individuals at high risk for cardiovascular events.

Conversely, NHANES data (NHANES, 1999-2004) represents a nationally representative sample of adults and children in the United States. Data collection encompasses detailed, face-to-face interviews, physical and physiological examinations, and laboratory tests, some of which overlap with data from the Framingham Heart Study. However, NHANES does not include long-term outcome information such as cardiovascular disease events. We consider this dataset

Table 1: Framingham Statistics Summary Stratified by Sex

	Sex=1(Male)	Sex=2(Female)	P-Value
n	1094	1445	
SEX (mean (SD))	1.00 (0.00)	2.00 (0.00)	<0.001
HDLC (mean (SD))	43.63 (13.37)	53.07 (15.67)	<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

the target population and aim to determine how well a model built from the source population performs with this data.

The common variables extracted from both datasets include: 1) TOTCHOL, serum total cholesterol (mg/dL); 2) SYSBP, systolic blood pressure; 3) AGE, age at examination; 4) HDLC, high-density lipoprotein cholesterol (mg/dL); 5) SEX, participant sex; 6) CURSMOKE, current cigarette smoking at examination; 7) DIABETES, diabetic status; 8) BPMEDS, use of anti-hypertensive medication at examination.

In terms of preprocessing, we generate two new variables derived from SYSBP and BPMEDS. The new variables are SYSBP_T, representing the systolic blood pressure for participants using anti-hypertensive medication. For participants on such medication, we retain the SYSBP value; otherwise, we assign a 0 score. The second variable, SYSBP_UT, represents the systolic blood pressure for participants not using anti-hypertensive medication. Here, if the participant does not take the medication, we use the SYSBP value; if they do, we assign a 0 score.

There are some missing data in the NHANES dataset and we decided to use complete dataset going forward because in the transportability analysis we prefer to keep the data as original as possible without introducing some error in the dataset.

Below is the statistics summary of the Framingham and NHANES data (complete case).

3. Method

3.1. Analysis Plan

We begin by constructing a logistic regression model using the Framingham data to predict the occurrence of cardiovascular disease, with the model stratified by sex. We transform the variables TOTCHOL, SYSBP_T, SYSBP_UT, AGE, and HDLC by taking their logarithmic

Table 2: NHANES Statistics Summary Stratified by Sex

	Sex=1(Male)	Sex=2(Female)	P-Value
n	746	760	
SEX (mean (SD))	1.00 (0.00)	2.00 (0.00)	<0.001
HDLC (mean (SD))	47.08 (14.15)	57.43 (16.36)	<0.001
TOTCHOL (mean (SD))	177.14 (42.36)	194.63 (42.21)	<0.001
AGE (mean (SD))	61.88 (13.45)	62.98 (12.87)	0.106
SYSBP (mean (SD))	134.79 (18.90)	138.58 (21.52)	<0.001
CURSMOKE (mean (SD))	0.17 (0.38)	0.14 (0.35)	0.086
DIABETES (mean (SD))	0.35 (0.48)	0.26 (0.44)	<0.001
BPMEDS (mean (SD))	0.85 (0.35)	0.86 (0.35)	0.768

values. Additionally, we incorporate binary predictors, such as CURSMOKE and DIABETES, into the models.

To assess the model's transportability to the target population, we utilize the equation outlined in Steingrimsson (2023) to calculate the mean squared error (MSE) for the target population. This calculation also requires the computation of a weighting estimator, which utilizes inverse odds weights derived from a model that predicts the probability of belonging to the source population based on covariates. We combine data from both the source and target populations to establish these weights. We assign a membership indicator (S), where 1 indicates data from the source population and 0 denotes data from the target population.

The formula for transportability analysis that we use in this study is based on Steingrimsson (2023)

$$\hat{\psi_{\hat{\beta}}} = \frac{\Sigma_{i=1}^{n} I(S_i = 1)\hat{o}(X_i)(Y_i - g_{\hat{\beta}}(X_i))^2}{\Sigma_{i=1}^{n} I(S_i = 0)}$$

The inverse weights we use are

$$\hat{o}(X) = \frac{Pr(S = 0|X)}{Pr(S = 1|X)}$$

Given that we have access to the NHANES data, we conduct two types of evaluations. The first uses the actual, non-simulated NHANES dataset, and the second employs a simulated dataset. For the latter scenario, we simulate a situation where we lack access to the individual records of the NHANES data and have only the dataset's statistical summary. This approach reflects a common real-world constraint where full datasets like NHANES are unavailable, and only summary statistics of the target population are accessible.

3.2 GLM model

Two types of generalized linear models (GLMs) are utilized in this study, and each is constructed separately for men and women.

The first model is designed to predict the occurrence of cardiovascular disease, with separate models based on sex.

$$logit(E[Y]) = \beta_o + \beta_1 log(HDLC) + \beta_2 log(TOTCHOL) + \beta_3 log(AGE) + \beta_4 log(SYSBP_{UT} + 1) + \beta_5 log(SYSBP_T + 1) + \beta_6 CURSMOKE + \beta_7 DIABETES$$

The second model aims to predict the probability of membership, again with separate models based on sex. This model is structured similarly to the first but differs in terms of the outcome variable.

$$logit(E[S]) = \beta_o + \beta_1 log(HDLC) + \beta_2 log(TOTCHOL) + \beta_3 log(AGE) + \beta_4 log(SYSBP_{UT} + 1) + \beta_5 log(SYSBP_T + 1) + \beta_6 CURSMOKE + \beta_7 DIABETES$$

3.1 Transportability Analysis

3.3. Non Simulated Target Population

As previously mentioned, to calculate the mean squared error (MSE) for the target population, we first need to merge our datasets. This is applicable to both the non-simulated and simulated target populations.

In the case of the non-simulated dataset, the process is more straightforward. We need to merge the Framingham and NHANES datasets, filtering to include only complete records. We then assign the membership values accordingly. Once the datasets are combined, we apply the first GLM model to estimate Y (in this case, Y represents CVD). Subsequently, we need to apply the second GLM model to determine $Pr(S=1|X)$ and subsequently calculate the inverse weight for each record. We apply both models separately for men and women and then amalgamate the results. Finally, we can estimate the MSE for the target population.

3.4. Simulated Target Population

Factors such as simulation size, random seeds, collected measures, data generation, and simulation parameters are described and justified.

Using the ADEMP framework, below is the design for evaluating the model using simulated target population.

Table 3: NHANES Statistics Summary Stratified by Sex

	Sex=1(Male)	Sex=2(Female)	P-Value
n	746	760	
SEX (mean (SD))	1.00 (0.00)	2.00 (0.00)	<0.001
LOG.HDLC (mean (SD))	3.81 (0.27)	4.01 (0.27)	<0.001
LOG.TOTCHOL (mean (SD))	5.15 (0.24)	5.25 (0.21)	<0.001
LOG.AGE (mean (SD))	4.10 (0.25)	4.12 (0.24)	0.090
LOG.SYSBP (mean (SD))	4.89 (0.14)	4.92 (0.15)	0.001
CURSMOKE (mean (SD))	0.17 (0.38)	0.14 (0.35)	0.086
DIABETES (mean (SD))	0.35 (0.48)	0.26 (0.44)	<0.001
BPMEDS (mean (SD))	0.85 (0.35)	0.86 (0.35)	0.768

3.4.1. Data Generation Method

When considering which parameters to vary, we look to the target population MSE/Brier Score formula. For instance, inaccuracies in specifying the distribution underlying the target population data or introducing correlations in the target population dataset could lead to less accurate predictions of CVD_hat, estimated by a generalized linear model. This, in turn, could result in a larger target population MSE/Brier Score. In light of this, we conduct a full-factorial simulation study examining three design factors: the number of correlated columns, correlation scores for these columns, and the underlying distribution of the data. The values for each factor are as follows:
1) The number of correlated columns (corr_n) is set at two levels: 2 (continuous data partially correlated with each other) and 4 (continuous data full correlated with each other).
2) Correlation scores (corr) for the correlated columns vary across four levels: 0 (no correlation), 0.3 (weak correlation), 0.7 (moderate correlation), and 1 (true correlation).
3) The underlying distribution of the data (shape) is considered at two levels: following the Framingham distribution and using a normal distribution.
Additionally, the shared parameters and models for data generation include:
1) The number of observations (num_obs), set at 2539, matching the source population observations. 2) The statistical summary from the NHANES data, used to simulate all datasets.
3) The model fitted for cardiovascular disease events from the source population.

Below, we present the underlying statistical summary for the NHANES data with log transformation in the Table 3 to generate data with lognormal distribution. We will also refer to Table 2 for the statistical summary for normal distribution.

To determine the underlying distribution of the Framingham continuous data, we refer to the histogram plots in the Figure 1. These plots suggest that the Framingham continuous data closely resemble lognormal distributions and the distribution of the log transformation of the continuous variables resemble normal distribution. For the categorical data, we assume Bernoulli distributions.

Histogram of Framingham Continuous Variables

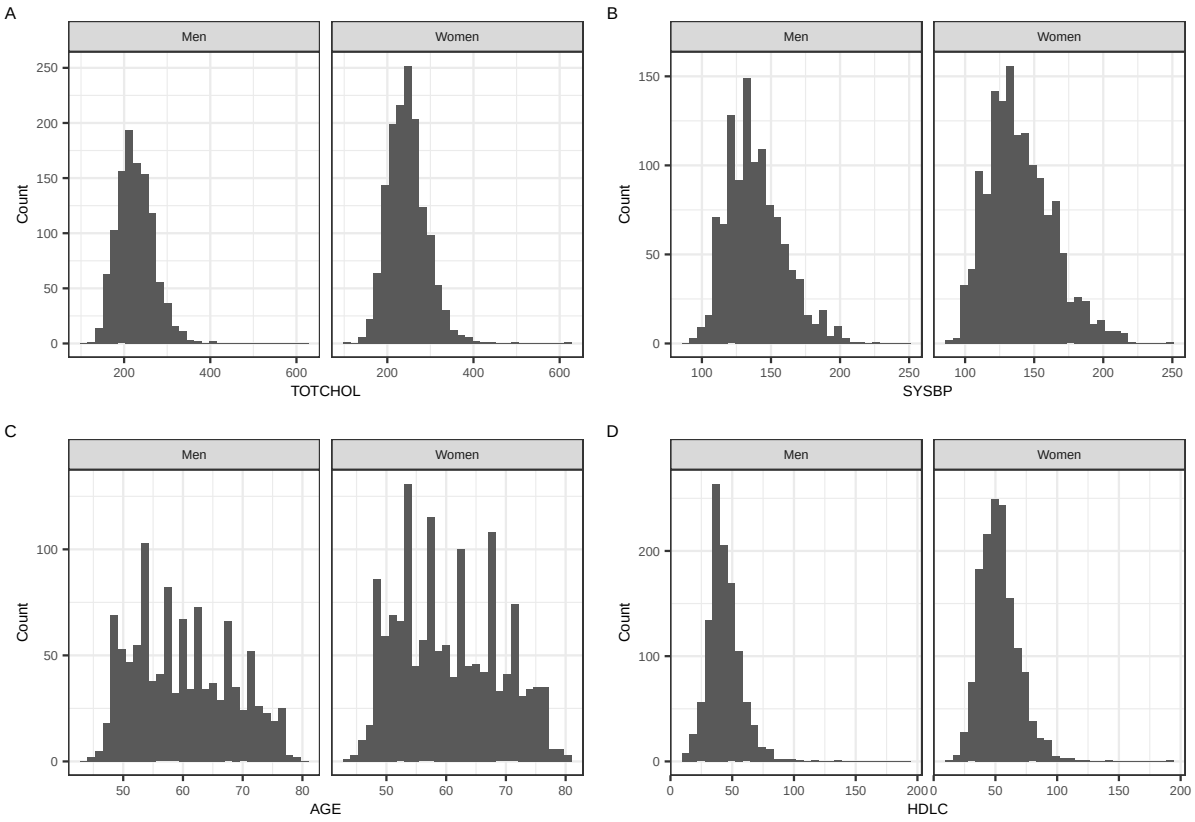


Figure 1. Histogram of Framingham Continuous Variables

Histogram of Framingham Log Continuous Variables

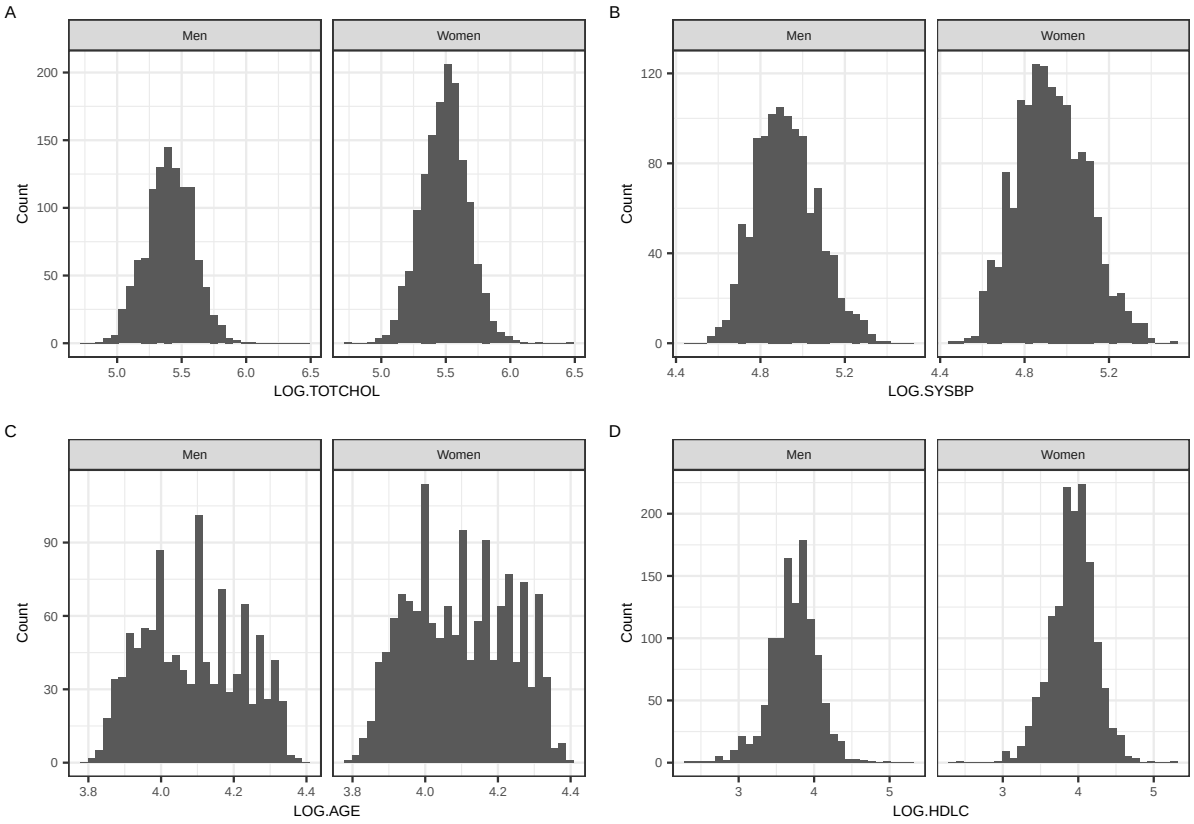


Figure 2. Histogram of Framingham Log Continuous Variables

We could also see from the Table 2 that the continuous data from source population has moderate correlation among each other.

Table 4: Correlation of Source Population Continuous Data

	TOTCHOL	HDLC	AGE	SYSBP
TOTCHOL	1.0000000	0.1816020	0.0890339	0.1025918
HDLC	0.1816020	1.0000000	-0.0072341	-0.0122821
AGE	0.0890339	-0.0072341	1.0000000	0.3341253
SYSBP	0.1025918	-0.0122821	0.3341253	1.0000000

3.4.2. Estimand

The estimand in this simulation study is the target population MSE/Brier Score.

3.4.3. Method

In each simulated target population dataset, we generate data using combinations of correlation values, the number of correlated columns, and different shapes, as previously specified. We then merge the source population data with the simulated target population dataset to perform the transportability analysis and obtain the estimand.

3.4.4. Performance Measure

In this simulation study, we do not have access to the true parameter, so we focus on the relative bias and empirical standard error (SE), with relative bias being the key performance measure of interest. However, as we have the target population data from the non-simulated dataset, we can treat it as the true estimand for comparison purposes. This allows us to compare results from both non-simulated and simulated data. Therefore, mean squared error (MSE) as additional performance measures, assuming that the transportability results from the non-simulated data represent the true estimand.

Regarding the number of repetitions for each simulated dataset, we based our assumptions on the variance of the estimand (determined from an initial small simulation run) is 0.002 (men) and 0.079 (women). We aimed for the required Monte Carlo SE of bias to be lower than 0.005 since the estimates are quite small. Using this information, we calculated that $n_{iter} = 3154$ is necessary.

4. Result and Discussion

The result from the non simulated dataset (NHANES) 0.1267 for men model and 0.0957 for women model. It suggests that the model's predictions in the target population have a mean squared error of 0.1267 for men model 0.0957 for women model. The women model performs better compared to the men model probably because the women in the NHANES data has fewer

comorbidities (e.g. lower diabetes proportion in population). But both indicates a reasonable level of accuracy and the original models could be reliably used in the target population.

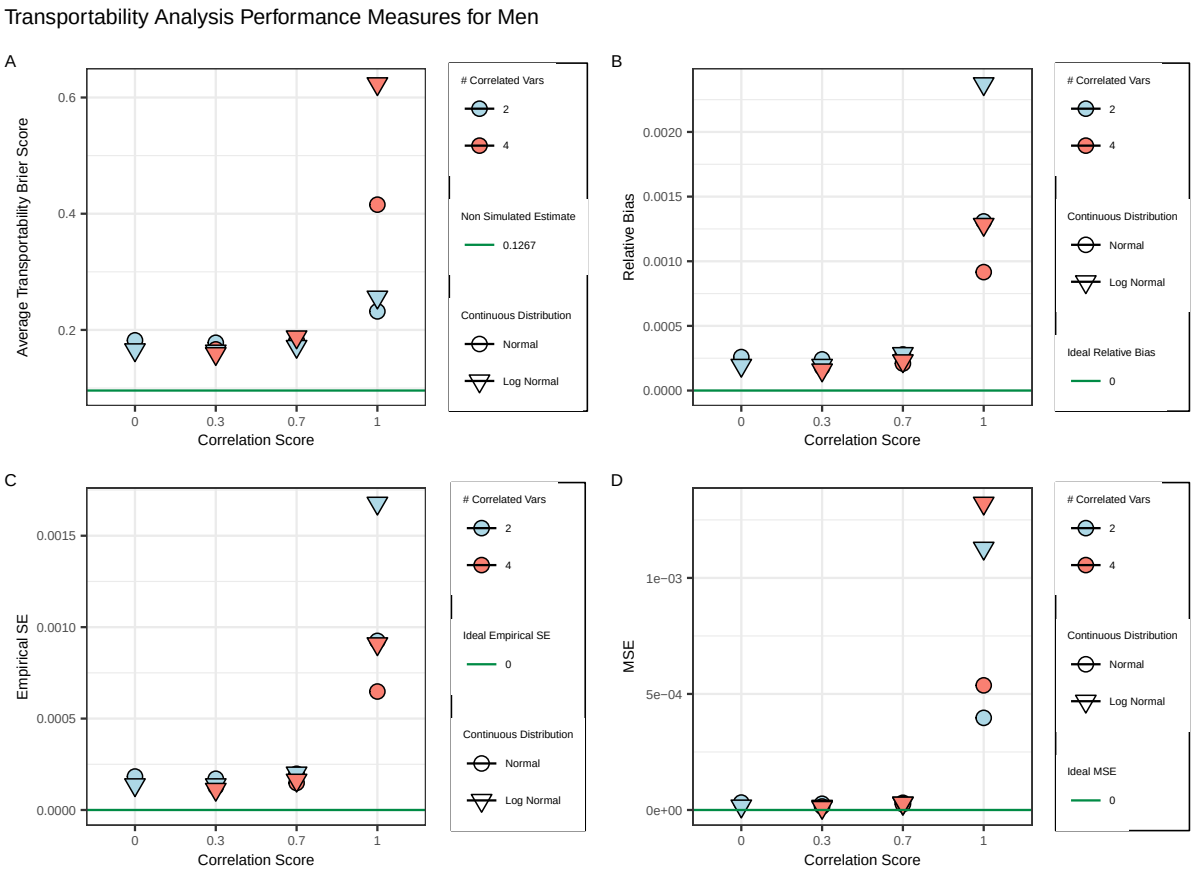


Figure 3. Transportability Analysis Performance Measures for Men

The results in Figure 3 and Figure 4 showed that models for women still perform better, and the best performing combinations for both men and women were those with distribution shapes similar to the original Framingham dataset and with no or moderate correlation. In particular, the men model has the lowest relative bias, empirical SE, MSE, and average brier score (0.00016, 0.00011, 0.00001, and 0.1592) when the dataset has moderate correlation (0.3) among all the continuous distribution and the continuous distribution follows the framingham's distribution shape (lognormal). In comparison, the women model perform best (0.00007 relative bias, 0.00005 empirical SE, 0.0000 MSE, 0.0974 average brier score) when the continuous variables do not have any correlation among each other and the continuous distribution follows the framingham's distribution shape. However the difference with the dataset that follows normal distribution and has no correlation among the continuous variables is close.

In general, both models tend to perform better (with smaller MSE, relative bias, and empirical SE) when they adhere to the original distribution of the source population and has minimal correlation among the continuous columns. This makes sense, as the dataset closely mimics the

Transportability Analysis Performance Measures for Women

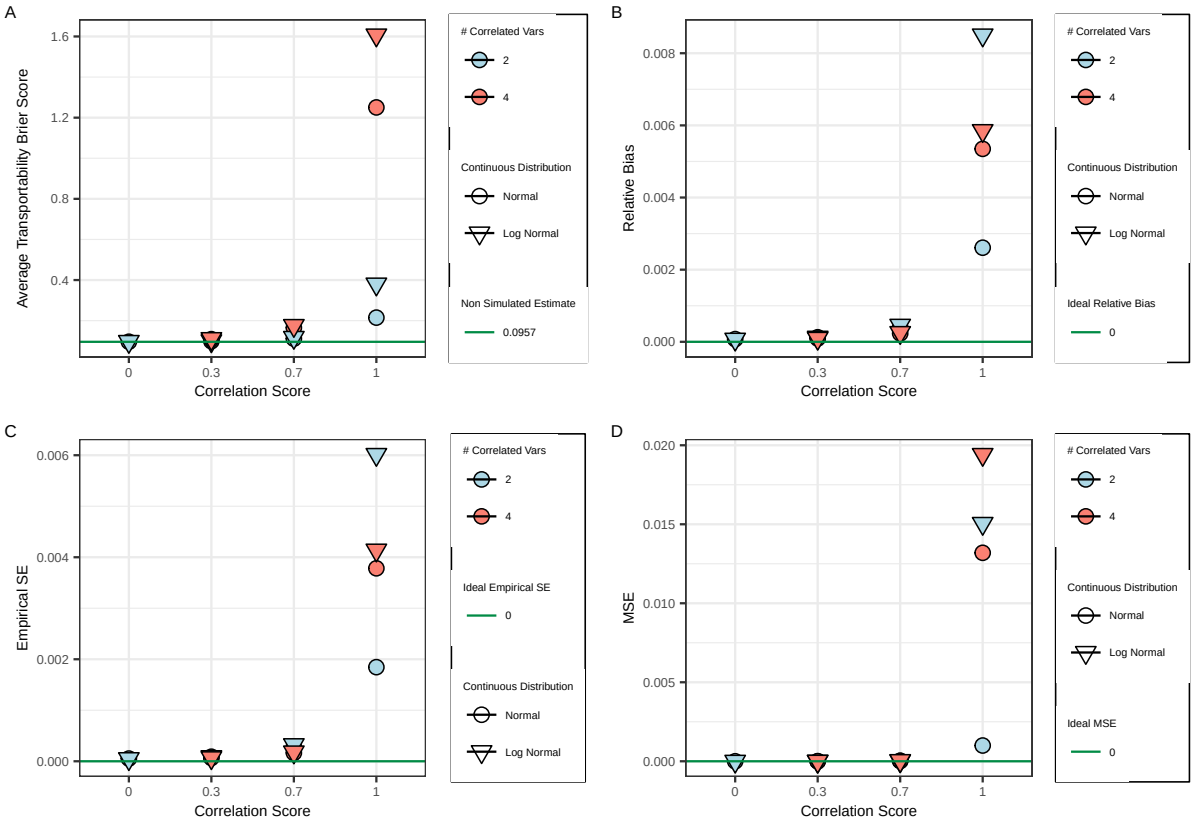


Figure 4. Transportability Analysis Performance Measures for Women

distribution and correlation patterns found in the source population. It also suggests that by closely following the source population's distribution and correlation, we can achieve estimates that are quite close to those of the non-simulated target population MSE/Brier Score. It is also evident that overestimating the correlation in the data leads to poorer outcomes across all metrics compared to underestimating the correlation.

In conclusion, to simulate a target population effectively for a cardiovascular disease prediction model, it's best to mimic the source population as closely as possible, both in terms of the continuous distribution's shape and the correlation patterns.

5. Limitation

This study is subject to several limitations. Firstly, our analysis is limited to assessing whether the continuous distribution of the data corresponds with that of the source population or adheres to a normal distribution. Secondly, there is an underlying assumption that all continuous variables in the Framingham data follow a lognormal distribution, based on the apparent closeness of fit. However, it's important to note that the age data might not strictly conform to a lognormal distribution in reality. Thirdly, in our approach, we uniformly apply the same correlation values across all correlated columns, a scenario that is highly improbable in real-world data. Lastly, we only incorporate positive correlations. Improvement in these areas could could be crucial for a more accurate and comprehensive analysis.

Supplemental Material

Supplemental material can be seen in this github page

Reference

Wilson, P. W., D'Agostino, R. B., Levy, D., Belanger, A. M., Silbershatz, H., & Kannel, W. B. (1998). Prediction of coronary heart disease using risk factor categories. Circulation, 97(18), 1837-1847. https://doi.org/10.1161/01.cir.97.18.1837

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

Centers for Disease Control and Prevention. (1999-2004). National Center for Health Statistics. National Health and Nutrition Examination Survey Data. U.S. Department of Health and Human Services. http://www.cdc.gov/nchs/nhanes.htm

Code Appendix

```
##############
### SETUP ###
##############

library(formatR)

knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(message = F)
knitr::opts_chunk$set(warning = F)
knitr::opts_chunk$set(fig.align="center")
knitr::opts_chunk$set(fig.width=8, fig.height=6)

###############
### LIBRARY ###
###############
library(riskCommunicator)
library(tidyverse)
library(tableone)
library(MASS)
library(truncnorm)
library(ggplot2)
library(kableExtra)
library(patchwork)
# The NHANES data here finds the same covariates among this national survey data
library(nhanesA)
#############
### SEED ###
#############

set.seed(7)
########################
### DEFINE FUNCTIONS ###
########################

brier_transport = function(df) {
  #' Calculate the target population MSE/Brier Score based on Steingrimsson (2023)
  #' @param df, dataset including source population and target population
  #' @param return, MSE/Brier Score from df
```

```r
  # Separate Source and Target Population
  s1 = df %>% filter(S==1)
  s0 = df %>% filter(S==0)

  # Calculate the nominator from the target population MSE/Brier Score
  s1 = s1 %>%
    mutate(noms = WEIGHT * (CVD - CVD_HAT)^2)
  nom = sum(s1$noms)

  # Calculate the denominator from the target population MSE/Brier Score
  denom = nrow(s0)

  # Calculate target population MSE/Brier Score
  result = nom/denom
  return(result)
}

transportability_analysis <- function(source_pop, target_pop, dist_shape='normal') {
  #' Calculate the target population MSE/Brier Score from specified source
  #'    population dataset and target population dataset for men and women
  #' @param source_pop, dataset of source population
  #' @param target_pop, dataset of target population
  #' @param return, brier score for target population for men and women

  # Combine source and target population
  combined_df = rbind(source_pop, target_pop)
  # Separate men and women data from the combined dastaset
  combined_df_men = combined_df %>% filter(SEX == 1)
  combined_df_women = combined_df %>% filter(SEX == 2)

  # Make the membership probability model for bot men and women
  pr_s_mod_men <- glm(S ~ log(HDLC)+log(TOTCHOL)+log(AGE)+log(SYSBP_UT+1)
                      +log(SYSBP_T+1)+CURSMOKE+DIABETES
                      , data = combined_df_men, family= "binomial")

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

  # Only take relevant columns to be modeled
  combined_df_mod = combined_df[,c('HDLC','TOTCHOL','AGE','SYSBP_UT'
```

```
                                    ,'SYSBP_T','CURSMOKE','DIABETES')]

  # Get the Pr(S=1|X) for men and women
  pr_S_m = predict(pr_s_mod_men, newdata = combined_df_mod, type = 'response')
  pr_S_f = predict(pr_s_mod_women, newdata = combined_df_mod, type = 'response')
  # Get the inverse weight for men and women
  weight_m =  (1-pr_S_m) / pr_S_m
  weight_f =  (1-pr_S_f) / pr_S_f

  # Get the CVD_hat (y_hat) for men and women
  cvd_hat_m = predict(mod_men, newdata = combined_df_mod , type = 'response')
  cvd_hat_f = predict(mod_women, newdata = combined_df_mod , type = 'response')

  # Put the inverse weight and CVD_hat to the combined dataset
  combined_df = combined_df %>%
    mutate(WEIGHT_M = weight_m
           ,WEIGHT_F = weight_f
           ,CVD_HAT_M = cvd_hat_m
           ,CVD_HAT_F = cvd_hat_f) %>%
    mutate(WEIGHT = ifelse(SEX == 1, WEIGHT_M, WEIGHT_F)
           ,CVD_HAT = ifelse(SEX == 1, CVD_HAT_M, CVD_HAT_F)) %>%
    dplyr::select(-c(WEIGHT_M,WEIGHT_F,CVD_HAT_M,CVD_HAT_F))


  # Calculate Target Population MSE/Brier Score
  result_m = brier_transport(combined_df %>% filter(SEX == 1))
  result_f = brier_transport(combined_df %>% filter(SEX == 2))
  return(c(result_m,result_f))
}

data_gen <- function(n, corr_any, corr_n, corr, dist_shape) {
  #' Generate dataset based on NHANES statistics summary
  #'    and use Framingham continuous distribution shape (ori) or use normal dist
  #' @param n, number of observations in dataset
  #' @param corr_any, apply correlation in continuous dataset, Yes or No
  #' @param corr_n, how many continuous columns that have correlation
  #' @param corr, correlation magnitude
  #' @param dist_shape, distribution shape for the continuous columns
  #' @param return, final dataset

  # Stopping Criterions
```

```
if(!corr_any %in% c(TRUE, FALSE)) {
  stop("corr_any must be TRUE/FALSE")
}

if((corr_n == FALSE) & (corr_n !=0)) {
  stop("corr_n must be 0 if corr_any is FALSE")
}

if((corr_n == FALSE) & (!corr_n %in% c(2,3,4))) {
  stop("corr_n must be 2/3/4 if corr_any is TRUE")
}

if(abs(corr) > 1) {
  stop("Corr cannot be more than 1 or less than -1")
}

if(!dist_shape %in% c('normal', 'ori')) {
  stop("corr_any must be ori/normal")
}

# Determine the columns that has correlation
if(corr_any == TRUE) {
  corr_idxs = sample(1:4, corr_n)
}

# Initialize final dataset
final_df = data.frame()
# Generate data for men (SEX = 1) and women (SEX = 2)
for (i in 1:2) {
  if (i == 1) {
    # Generate categorical columns for men bases on NHANES statistics summary
    n_sex = round(n * 0.50)
    sex = rep(i, n_sex)
    bpmeds = rbinom(n_sex, 1, 0.85)
    cursmoke = rbinom(n_sex, 1, 0.17)
    diabetes = rbinom(n_sex, 1, 0.35)

    # Specify mean and sd for men bases on NHANES statistics summary
    # 1)TOTCHOL, 2)SYSBP, 3)AGE, 4)HDLC
    # If distribution shape is similar to framingham/lognormal
    if(dist_shape == 'ori') {
```

```
    means = c(5.15, 4.89, 4.10, 3.81)
    sds = c(0.24, 0.14, 0.25, 0.27)
  }

  # If distribution shape is normal
  if(dist_shape == 'normal') {
    means = c(177.14, 134.79, 61.88, 47.08)
    sds = c(42.36, 18.90, 13.45, 14.15)
  }

}
else {
  # Generate categorical columns for women bases on NHANES statistics summary
  n_sex = n - round(n * 0.49)
  sex = rep(i, n_sex)
  bpmeds = rbinom(n_sex, 1, 0.86)
  cursmoke = rbinom(n_sex, 1, 0.14)
  diabetes = rbinom(n_sex, 1, 0.09)

  # Specify mean and sd for women bases on NHANES statistics summary
  # 1)TOTCHOL, 2)SYSBP, 3)AGE, 4)HDLC
  # If distribution shape is similar to framingham/lognormal
  if(dist_shape == 'ori') {
    means = c(5.25, 4.92, 4.12, 4.01)
    sds = c(0.21, 0.15, 0.24, 0.27)
  }

  # If distribution shape is normal
  if(dist_shape == 'normal') {
    means = c(194.63,138.58,62.98,57.43)
    sds = c(42.21,21.52,12.87,16.36)
  }
}

# Generate continuous columns based on determined dist_shape
continuous_list = vector('list',4)
for (j in 1:4) {
  # Get mean and sd for the corresponding continuous columns
  # 1)TOTCHOL, 2)SYSBP, 3)AGE, 4)HDLC
  m = means[j]
  s = sds[j]
```

```
  # If dist_shape == 'ori' then use framingham distribution shape, lognormal
  if (dist_shape == 'ori') {
    dat = rnorm(n_sex, m, s)
  }

  # If dist_shape == 'normal' then use normal distribution shape
  if (dist_shape == 'normal') {
    dat = rnorm(n_sex, m, s)
    # For age, truncate the data, min = 1, max = 81 (based on framingham)
    if(j == 3) {dat = rtruncnorm(n = n_sex,a = 1,b = 81,mean = m,sd = s)}
  }

  # Combine all continuous columns
  continuous_list[[j]] = dat
}

# Make the continuous columns correlated if corr_any == TRUE
if (corr_any == TRUE) {

  # Generate multivariate normal dataset that correspond with the corr value
  corr_mat = matrix(corr, ncol = corr_n, nrow = corr_n)
  diag(corr_mat) = 1
  mvdat = mvrnorm(n = n_sex, mu = rep(0,corr_n), Sigma = corr_mat, empirical = TRUE)

  # Apply the correlation to the existing continuous columns by using the
  #   sorted position of the multivariate normal dataset
  k = 1
  for (corr_idx in corr_idxs) {

    # Sort multivariate normal dataset
    rnk = rank(mvdat[ , k], ties.method = "first")
    # Arrange the continuous columns based on the previous sorting
    dat_sorted = sort(continuous_list[[corr_idx]])
    # Replace the original continuous columns
    continuous_list[[corr_idx]] = dat_sorted[rnk]
    k = k + 1
  }

  # Remove mvdat var to avoid mvrnomr overwrite error
  rm(mvdat)
}
```

```
    # Make dataset based on the continuous and categorical columns
    continuous_df = data.frame(TOTCHOL = continuous_list[[1]]
                               , SYSBP = continuous_list[[2]]
                               , AGE = continuous_list[[3]]
                               , HDLC = continuous_list[[4]])
    if(dist_shape == 'ori') {
      continuous_df = continuous_df %>%
        mutate(TOTCHOL = exp(TOTCHOL)
               , SYSBP = exp(SYSBP)
               , AGE = exp(AGE)
               , HDLC = exp(HDLC))
    }
    continuous_df = continuous_df[sample(nrow(continuous_df)),]

    df_tmp = data.frame(SEX = sex
                        , BPMEDS = bpmeds
                        , CURSMOKE = cursmoke
                        , DIABETES = diabetes
                        , continuous_df)

    # Append to final dataset
    final_df = rbind(final_df, df_tmp)
  }

  # Preprocess dataset so the structure is accepted by the models
  final_df =  final_df %>%
    mutate(S = 0
           , CVD = NA
           , SYSBP_UT = ifelse(BPMEDS == 0, SYSBP, 0)
           , SYSBP_T = ifelse(BPMEDS == 1, SYSBP, 0)
    ) %>%
    dplyr::select(HDLC, TOTCHOL, AGE, SYSBP, BPMEDS, SYSBP_UT, SYSBP_T
                  , CURSMOKE, DIABETES, SEX, S, CVD)

  # Return dataset
  return(final_df)

}

# save(brier_transport, file = 'brier_transport.Rda')
# save(transportability_analysis, file = 'transportability_analysis.Rda')
```

```r
# save(data_gen, file = 'data_gen.Rda')
load('brier_transport.Rda')
load('transportability_analysis.Rda')
load('data_gen.Rda')
#########################
### PRERPOCESS 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))
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)

save(framingham_df, file = 'framingham_df.Rda')

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

```r
save(framingham_df_men, file = 'framingham_df_men.Rda')
save(framingham_df_women, file = 'framingham_df_women.Rda')

# 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 = ifelse(BPQ050A == 1, 1, 0)) %>%
  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")

save(df_2017, file = 'df_2017.Rda')
# CreateTableOne(data = df_2017, strata = c("SEX"))

#################
### FIT MODEL ###
#################

# 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_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")

save(mod_men, file = 'mod_men.Rda')

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")

save(mod_men, file = 'mod_women.Rda')

load('framingham_df.Rda')
load('framingham_df_men.Rda')
load('framingham_df_women.Rda')
load('mod_men.Rda')
load('mod_women.Rda')
load('df_2017.Rda')
########################################
### PREPARE SOURCE POP DATA FOR MODEL ###
########################################

# Prepare the source population dataset for the models
framingham_prep = framingham_df %>%
  dplyr::select(c(HDLC,TOTCHOL,AGE,SYSBP,BPMEDS,CURSMOKE,DIABETES,SEX,CVD)) %>%
  # Generate SYSBP_UT and SYSBP_T, combination of SYSBP and BPMEDS
  # Generate S, the membership indication
```

```r
  mutate(SYSBP_UT = ifelse(BPMEDS == 0, SYSBP, 0)
         ,SYSBP_T = ifelse(BPMEDS == 1, SYSBP, 0)
         ,S = 1) %>%
  dplyr::select(c(HDLC,TOTCHOL,AGE,SYSBP,BPMEDS,SYSBP_UT,SYSBP_T,CURSMOKE
                  ,DIABETES,SEX,S,CVD))

# save(framingham_prep, file = 'framingham_prep.Rda')
#######################################################
### PREPARE TARGET POP (NON SIM) DATA FOR MODEL ###
#######################################################

# Prepare the non simulated target population dataset for the models
nhanes_prep = df_2017 %>%
  dplyr::select(c(HDLC,TOTCHOL,AGE,SYSBP,BPMEDS,CURSMOKE,DIABETES,SEX)) %>%
  # Take complete data only
  na.omit() %>%
  # Generate SYSBP_UT and SYSBP_T, combination of SYSBP and BPMEDS
  # Generate S, the membership indication
  mutate(SYSBP_UT = ifelse(BPMEDS == 0, SYSBP, 0)
         ,SYSBP_T = ifelse(BPMEDS == 1, SYSBP, 0)
         ,S = 0) %>%
  mutate(CVD = NA) %>%
  dplyr::select(c(HDLC,TOTCHOL,AGE,SYSBP,BPMEDS,SYSBP_UT,SYSBP_T,CURSMOKE
                  ,DIABETES,SEX,S,CVD))

# save(nhanes_prep, file = 'nhanes_prep.Rda')
load('framingham_stat_tb.Rda')
load('nhanes_stat_tb.Rda')

# Print statistics summary of Framingham
framingham_stat_tb %>%
  kableExtra::kbl(caption = 'Framingham Statistics Summary Stratified by Sex'
                  , booktabs = T
                  , escape = T
                  , align = 'c'
                  , col.names = c('Sex=1(Male)','Sex=2(Female)', 'P-Value','')) %>%
  kableExtra::kable_classic(full_width = F
                            , html_font = 'Cambria'
                            , font_size = 5
                            # , position = 'float_left'
                            , latex_options = 'scale_down')
```

```r
# Print statistics summary of NHANES
nhanes_stat_tb %>%
  kableExtra::kbl(caption = 'NHANES Statistics Summary Stratified by Sex'
                  , booktabs = T
                  , escape = T
                  , align = 'c'
                  , col.names = c('Sex=1(Male)','Sex=2(Female)', 'P-Value','')) %>%
  kableExtra::kable_classic(full_width = F
                            , html_font = 'Cambria'
                            , font_size = 5
                            # , position = 'right'
                            , latex_options = 'scale_down')
##############################
### GET NHANES LOG SUMMARY ###
##############################

nhanes_log = nhanes_prep %>%
  mutate(LOG.TOTCHOL = log(TOTCHOL)
         ,LOG.SYSBP = log(SYSBP)
         ,LOG.AGE = log(AGE)
         ,LOG.HDLC = log(HDLC)) %>%
  dplyr::select(SEX,LOG.HDLC,LOG.TOTCHOL,LOG.AGE,LOG.SYSBP,CURSMOKE,DIABETES,BPMEDS)

lognhanes_stat = CreateTableOne(data = nhanes_log, strata = c("SEX"))
lognhanes_stat_tb = print(lognhanes_stat$ContTable)
# save(lognhanes_stat_tb,file='lognhanes_stat_tb.Rda')
load('lognhanes_stat_tb.Rda')

# Print statistics summary of NHANES
lognhanes_stat_tb %>%
  kableExtra::kbl(caption = 'NHANES Statistics Summary Stratified by Sex'
                  , booktabs = T
                  , escape = T
                  , align = 'c'
                  , col.names = c('Sex=1(Male)','Sex=2(Female)', 'P-Value','')) %>%
  kableExtra::kable_classic(full_width = F
                            , html_font = 'Cambria'
                            , font_size = 5
                            , latex_options = 'scale_down')
##################################################
### OBSERVE DISTRIBUTIONS FROM SOURCE POP DATA ###
```

```
##################################################

# Non Log
# Get histogram from the framingham dataset continuous columns
fram_hist = vector('list',4)
cont_cols = c('TOTCHOL', 'SYSBP', 'AGE', 'HDLC')
it = 1
sex_label = c(`1` = 'Men',`2`='Women')

for (colname in cont_cols) {
  pl = ggplot(framingham_df) +
    geom_histogram(aes_string(x = colname)) +
    facet_grid(~SEX, labeller = as_labeller(sex_label)) +
    theme_bw() +
    labs(x = colname
         ,y = 'Count') +
    theme(text = element_text(size=7))

  fram_hist[[it]] = pl
  it = it + 1
}

patchworkHist = (fram_hist[[1]] + fram_hist[[2]] + fram_hist[[3]]
                 + fram_hist[[4]]) + plot_layout(nrow = 2, ncol = 2)
save(patchworkHist, file='patchworkHist.Rda')

# Log
# Get histogram from the framingham dataset continuous columns
fram_hist2 = vector('list',4)
cont_cols2 = c('LOG.TOTCHOL', 'LOG.SYSBP', 'LOG.AGE', 'LOG.HDLC')
it = 1
sex_label = c(`1` = 'Men',`2`='Women')

for (colname in cont_cols2) {
  pl = ggplot(framingham_log) +
    geom_histogram(aes_string(x = colname)) +
    facet_grid(~SEX, labeller = as_labeller(sex_label)) +
    theme_bw() +
    labs(x = colname
         ,y = 'Count') +
    theme(text = element_text(size=7))
```

```
  fram_hist2[[it]] = pl
  it = it + 1
}

patchworkHist2 = (fram_hist2[[1]] + fram_hist2[[2]] + fram_hist2[[3]]
                  + fram_hist2[[4]]) + plot_layout(nrow = 2, ncol = 2)
# save(patchworkHist2, file='patchworkLogHist2.Rda')
load('patchworkHist.Rda')
patchworkHist +
  plot_annotation(tag_levels = 'A'
                  ,title = "Histogram of Framingham Continuous Variables"
                  ,caption = 'Figure 1. Histogram of Framingham Continuous Variables'
                  ,theme = theme(plot.title = element_text(size = 10)
                                 ,plot.tag = element_text(size = 3)))

load('patchworkHist2.Rda')
patchworkHist2 +
  plot_annotation(tag_levels = 'A'
                  ,title = "Histogram of Framingham Log Continuous Variables"
                  ,caption = 'Figure 2. Histogram of Framingham Log Continuous Variables'
                  ,theme = theme(plot.title = element_text(size = 10)
                                 ,plot.tag = element_text(size = 3)))
########################################
### OBSERVE CORR FROM SOURCE POP DATA ###
########################################

cor(framingham_df[,c('TOTCHOL','HDLC','AGE','SYSBP')]) %>%
  kableExtra::kbl(caption = 'Correlation of Source Population Continuous Data'
                  , booktabs = T
                  , escape = T
                  , align = 'c') %>%
  kableExtra::kable_classic(full_width = F
                            , font_size = 5
                            , html_font = 'Cambria'
                            , latex_options = 'HOLD_position')
####################################################
### GENERATE INITIAL ESTIMATES FOR EXTREME CASES ###
####################################################

init_briers = replicate(2000, {
  dataa = data_gen(n=2359, corr_any=TRUE, corr_n=4, corr=0.99, dist_shape='normal')
```

```
  transportability_analysis(framingham_prep, dataa)
})

# save(init_briers, file = 'init_briers.Rda')
load('init_briers.Rda')

# Calculate number of iterations for each simulation
n_iters = var(init_briers)/(0.001)^2
load('framingham_prep.Rda')
load('nhanes_prep.Rda')

##################################################
### TRANSPORTABILITY ANALYSIS FOR NON SIM DATA ###
##################################################

# Apply transportability analysis to the non simulated target population dataset
nonsimulated_result = transportability_analysis(framingham_prep, nhanes_prep)
# nonsimulated_result # men 0.12673996 women 0.09572181
################################################
### TRANSPORTABILITY ANALYSIS FOR SIM DATA ###
################################################

corr_ns = c(2,4)
corrs = c(0, 0.3, 0.7, 1)
shapes = c('ori','normal')
num_rep = 3154
num_obs = 2539

# Simulation to get the target population MSE/Brier score based on the
#   specified values of shapes and corrs
outputs = data.frame()
for (shape in shapes) {
  for (corr in corrs) {
    for (corr_n in corr_ns) {

      if (corr == 0) {corr_any = FALSE}
      else {corr_any = TRUE}

      results = replicate(num_rep, {
        dataa = data_gen(n = num_obs
                         , corr_any = corr_any, corr_n = corr_n, corr = corr
```

```
                          , dist_shape = shape)
        transportability_analysis(framingham_prep, dataa)
      })
      outputs_tmp <- data.frame(ta = results, shape = shape, corr = corr, corr_n = corr_n)
      outputs = rbind(outputs, outputs_tmp)
    }
  }
}

# save(outputs, file = 'outputs2.Rda')
load('outputs2.Rda')
mc_bias <- function(estimates) {
  #' Calculate monte carlo relative bias using the mean estimates instead of
  #'    true estimates
  #' @param estimates, set of estimates
  #' @param return, monte carlo relative bias of the estimates
  nsim = length(estimates)
  mean_estimates = mean(estimates)
  est_diff_squared = (estimates - mean_estimates)^2
  result = sqrt(1/(nsim *(nsim-1)) * sum(est_diff_squared))
  return(result)
}

mc_empSE <- function(estimates) {
  #' Calculate monte carlo empirical SE
  #' @param estimates, set of estimates
  #' @param return, monte carlo empirical SE
  nsim = length(estimates)
  mean_estimates = mean(estimates)
  est_diff_squared = (estimates - mean_estimates)^2
  est_emp_SE = sqrt(1/(nsim-1) * sum(est_diff_squared))
  result = est_emp_SE / sqrt(2*(nsim-1))
  return(result)
}

mc_MSE <- function(estimates, true_estimate) {
  #' Calculate monte carlo MSE
  #' @param estimates, set of estimates
  #' @param return, monte carlo MSE
  nsim = length(estimates)
  est_diff_squared = (estimates - true_estimate)^2
```

```
  est_MSE = (1/(nsim) * sum(est_diff_squared))
  result = sqrt( sum((est_diff_squared - est_MSE)^2) / ((nsim)*(nsim-1)))
  return(result)
}

# Calculate the performance measures
eval_summary <- outputs %>%
  group_by(shape, corr, corr_n) %>%
  mutate(relative_bias_m = mc_bias(ta_m)
         ,empSE_m = mc_empSE(ta_m)
         ,MSE_m = mc_MSE(ta_m, nonsimulated_result[1])
         ,avg_ta_m = mean(ta_m)
         ,relative_bias_f = mc_bias(ta_f)
         ,empSE_f = mc_empSE(ta_f)
         ,MSE_f = mc_MSE(ta_f, nonsimulated_result[2])
         ,avg_ta_f = mean(ta_f)
  )%>%
  ungroup() %>%
  dplyr::select(-c('ta_m','ta_f')) %>%
  group_by(shape, corr, corr_n) %>%
  slice(1) %>%
  mutate_if(is.numeric, round, 6)
load('eval_summary.Rda')
# Initialization
all_plot_men = vector('list',4)
all_plot_women = vector('list',4)
iter = 1

for (i in c('M','F')) {
  if(i == 'M') {
    metrics_col = c('relative_bias_m', 'empSE_m', 'MSE_m', 'avg_ta_m')
    nonsim_res = 0.1267

    # Avg TA Brier Score
    plot1 = ggplot(eval_summary) +
      geom_point(aes_string(x = 'factor(corr)', y = metrics_col[4]
                            , fill = 'factor(corr_n)', shape = 'shape')
                 , size = 3) +
      geom_hline(data = eval_summary, aes(yintercept = nonsim_res
                                          , colour = "0.1267"), show_guide=TRUE) +
      scale_shape_manual(name= 'Continuous Distribution'
```

```
                        ,values = c(21,25),labels=c('Normal','Log Normal')) +
    scale_fill_manual(name= '# Correlated Vars'
                      ,values = c('lightblue','salmon')) +
    scale_color_manual(name = 'Non Simulated Estimate'
                       , values = c("0.1267" = "springgreen4"))+
    theme_bw() +
    labs(x = 'Correlation Score'
         ,y = 'Average Transportability Brier Score') +
    theme(text = element_text(size=7)
          ,legend.title=element_text(size=5)
          ,legend.text=element_text(size=5)
          ,legend.box.background = element_rect(color = "black")) +
    guides(fill = guide_legend("# Correlated Vars"
                               , override.aes = list(shape = 21)))
}

if(i == 'F') {
  metrics_col = c('relative_bias_f', 'empSE_f', 'MSE_f', 'avg_ta_f')
  nonsim_res = 0.0957

  # Avg TA Brier Score
  plot1 = ggplot(eval_summary) +
    geom_point(aes_string(x = 'factor(corr)', y = metrics_col[4]
                          , fill = 'factor(corr_n)', shape = 'shape')
               , size = 3) +
    geom_hline(data = eval_summary, aes(yintercept = nonsim_res
                                        , colour = "0.0957"), show_guide=TRUE) +
    scale_shape_manual(name= 'Continuous Distribution'
                       ,values = c(21,25),labels=c('Normal','Log Normal')) +
    scale_fill_manual(name= '# Correlated Vars'
                      ,values = c('lightblue','salmon')) +
    scale_color_manual(name = 'Non Simulated Estimate'
                       , values = c("0.0957" = "springgreen4"))+
    theme_bw() +
    labs(x = 'Correlation Score'
         ,y = 'Average Transportability Brier Score') +
    theme(text = element_text(size=7)
          ,legend.title=element_text(size=5)
          ,legend.text=element_text(size=5)
          ,legend.box.background = element_rect(color = "black")) +
    guides(fill = guide_legend("# Correlated Vars"
```

```
                                , override.aes = list(shape = 21)))
}

# Avg Relative Bias
plot2 = ggplot(eval_summary) +
  geom_point(aes_string(x = 'factor(corr)', y = metrics_col[1]
                        , fill = 'factor(corr_n)', shape = 'shape')
             , size = 3) +
  geom_hline(data = eval_summary, aes(yintercept = 0
                                      , colour = "0"), show_guide=TRUE) +
  scale_shape_manual(name= 'Continuous Distribution'
                     ,values = c(21,25),labels=c('Normal','Log Normal')) +
  scale_fill_manual(name= '# Correlated Vars'
                    ,values = c('lightblue','salmon')) +
  scale_color_manual(name = 'Ideal Relative Bias'
                     , values = c("0" = "springgreen4"))+
  theme_bw() +
  labs(x = 'Correlation Score'
       ,y = 'Relative Bias') +
  theme(text = element_text(size=7)
        ,legend.title=element_text(size=5)
        ,legend.text=element_text(size=5)
        ,legend.box.background = element_rect(color = "black")) +
  guides(fill = guide_legend("# Correlated Vars"
                             , override.aes = list(shape = 21)))

# Avg EmpSE
plot3 = ggplot(eval_summary) +
  geom_point(aes_string(x = 'factor(corr)', y = metrics_col[2]
                        , fill = 'factor(corr_n)', shape = 'shape')
             , size = 3) +
  geom_hline(data = eval_summary, aes(yintercept = 0
                                      , colour = "0"), show_guide=TRUE) +
  scale_shape_manual(name= 'Continuous Distribution'
                     ,values = c(21,25),labels=c('Normal','Log Normal')) +
  scale_fill_manual(name= '# Correlated Vars'
                    ,values = c('lightblue','salmon')) +
  scale_color_manual(name = 'Ideal Empirical SE'
                     , values = c("0" = "springgreen4"))+
  theme_bw() +
  labs(x = 'Correlation Score'
```

```
      ,y = 'Empirical SE') +
  theme(text = element_text(size=7)
        ,legend.title=element_text(size=5)
        ,legend.text=element_text(size=5)
        ,legend.box.background = element_rect(color = "black")) +
  guides(fill = guide_legend("# Correlated Vars"
                             , override.aes = list(shape = 21)))

# Avg MSE
plot4 = ggplot(eval_summary) +
  geom_point(aes_string(x = 'factor(corr)', y = metrics_col[3]
                        , fill = 'factor(corr_n)', shape = 'shape')
             , size = 3) +
  geom_hline(data = eval_summary, aes(yintercept = 0
                                      , colour = "0"), show_guide=TRUE) +
  scale_shape_manual(name= 'Continuous Distribution'
                     ,values = c(21,25),labels=c('Normal','Log Normal')) +
  scale_fill_manual(name= '# Correlated Vars'
                    ,values = c('lightblue','salmon')) +
  scale_color_manual(name = 'Ideal MSE'
                     , values = c("0" = "springgreen4"))+
  theme_bw() +
  labs(x = 'Correlation Score'
       ,y = 'MSE') +
  theme(text = element_text(size=7)
        ,legend.title=element_text(size=5)
        ,legend.text=element_text(size=5)
        ,legend.box.background = element_rect(color = "black")) +
  guides(fill = guide_legend("# Correlated Vars"
                             , override.aes = list(shape = 21)))


if (i == 'M') {
  all_plot_men[[1]] = plot1
  all_plot_men[[2]] = plot2
  all_plot_men[[3]] = plot3
  all_plot_men[[4]] = plot4
}
if (i == 'F') {
  all_plot_women[[1]] = plot1
  all_plot_women[[2]] = plot2
```

```
    all_plot_women[[3]] = plot3
    all_plot_women[[4]] = plot4
  }
}

# Combine plots for men
patchwork_men = (all_plot_men[[1]] + all_plot_men[[2]] + all_plot_men[[3]]
                 + all_plot_men[[4]]) + plot_layout(nrow = 2, ncol = 2)
# save(patchwork_men, file='patchwork_men.Rda')

# Combine plots for men
patchwork_women = (all_plot_women[[1]] + all_plot_women[[2]] + all_plot_women[[3]]
                   + all_plot_women[[4]]) + plot_layout(nrow = 2, ncol = 2)
# save(patchwork_women, file='patchwork_women.Rda')
load('patchwork_men.Rda')
load('patchwork_women.Rda')

options(repr.plot.width=5, repr.plot.height=3)
patchwork_men +
  plot_annotation(tag_levels = 'A'
                  ,title = "Transportability Analysis Performance Measures for Men"
                  ,caption = 'Figure 3. Transportability Analysis Performance Measures for
                  ,theme = theme(plot.title = element_text(size = 10)
                                 ,plot.tag = element_text(size = 3)))


options(repr.plot.width=5, repr.plot.height=3)
patchwork_women +
  plot_annotation(tag_levels = 'A'
                  ,title = "Transportability Analysis Performance Measures for Women"
                  ,caption = 'Figure 4. Transportability Analysis Performance Measures for
                  ,theme = theme(plot.title = element_text(size = 10)
                                 ,plot.tag = element_text(size = 3)))
```

Predictive Modeling for Tracheostomy and Mortality Outcomes in Bronchopulmonary Dysplasia

Abstract

This study focused on developing predictive models for tracheostomy and mortality outcomes in infants with Bronchopulmonary Dysplasia (BPD), leveraging data from the BPD Collaborative Registry. The registry comprised infants born before 32 weeks of gestational age, diagnosed with severe BPD. Data analysis involved exploring missing data patterns, where discrepancies in surfactant administration across various centers were noted, and implementing multiple imputation to handle missing-at-random data. The study identified key variables for outcomes through exploratory analysis, including high correlations and potential confounders like gestational age and positive end-expiratory pressure.

The dataset was split into training and testing sets, with a 70-30 proportion, to ensure robust model validation. Two types of models were developed for each outcome: one incorporating significant coefficients with a random effect from medical centers and another including interactions between time and clinical variables at 36 and 44 weeks. However, the addition of interaction terms did not significantly improve the models. The performance of the models was assessed using Area Under the Curve (AUC) and F1 scores, revealing satisfactory results for tracheostomy but poor F1 scores for death models, indicating an area for further model improvement.

The study's final multilevel model, chosen for its simplicity and effectiveness, combined significant variables and accounted for variations due to center-specific practices. While the models identified several key characteristics influencing tracheostomy and mortality, the study acknowledges limitations, including the absence of diverse interaction terms and a limited scope of random effects. Future research could address these limitations by exploring more interaction terms and broadening the scope of random effects. Despite these limitations, the study provides crucial insights into the factors influencing tracheostomy and mortality in neonates with BPD, guiding targeted interventions and policy decisions to improve neonatal care outcomes.

Keywords: Bronchopulmonary Dysplasia, Tracheostomy, Mortality, Predictive Modeling, Multilevel Models, Neonatal Care

1. Introduction

Bronchopulmonary Dysplasia (BPD) is a complication associated with prematurity, impacting a significant number of infants each year, particularly in its severe form, affecting 10,000-15,000 newborns annually. Various factors, including genetics and epigenetics, influence the development of BPD. This condition manifests as a persistent lung ailment, primarily afflicting prematurely born infants, necessitating oxygen therapy for their respiratory support. In BPD, there is notable damage to the lungs and airways (bronchi), leading to tissue damage (dysplasia) in the small air sacs of the lungs (alveoli). The severity of BPD is categorized into different grades, with Grade 3 BPD marking a critical point where reliance on a ventilator is necessary at 36 weeks corrected gestational age. Notably, 75% of infants with Grade 3 BPD continue to require ventilator support upon discharge, while 25% do not. For those who need ventilator support upon discharge, a tracheostomy involving a surgical opening in the neck facilitating connection to a ventilator becomes a prerequisite. The incidence of tracheostomy in infants with BPD ranges from 2-4%, escalating to 12% in cases of severe or Grade 3 BPD.

While the advantages of performing a tracheostomy include ensuring a stable airway, improving ventilator synchronization, and fostering growth, it is essential to acknowledge the associated risks. These risks include an increased likelihood of mortality compared to cases without tracheostomy, the potential for accidental decannulation leading to fatal outcomes, cannula obstruction with similar dire consequences, elevated rates of infection affecting the skin, trachea, and lungs, and the development of tracheal stenosis.

Given these considerations, cautious decision-making is crucial when contemplating the implementation of tracheostomy in infants diagnosed with BPD. Consequently, this study aims to develop statistical models utilizing clinical data collected at 36 and 44 weeks post-menstrual age (PMA). These models aim to predict the eventual necessity for tracheostomy or the likelihood of mortality preceding discharge, providing a valuable framework for informed decision-making in the management of BPD in newborns.

2. Data

Participants in this study were sourced from the BPD Collaborative Registry, a collaborative network of BPD programs in the United States and Sweden. The consortium was established to bridge evidence gaps and advance research for improving care in children affected by severe bronchopulmonary dysplasia (BPD). The registry focuses on infants born with a gestational age of less than 32 weeks and diagnosed with sBPD, defined according to the 2001 NHLBI criteria, specifically requiring FiO2 0.3 or positive pressure ventilation (invasive or non-invasive) at 36 weeks post-menstrual age (PMA). Standard demographic and clinical data are routinely

collected at four key time points: birth, 36 weeks PMA, 44 weeks PMA, and discharge. For this study, we extracted data from the registry for patients with BPD and complete growth information, covering the period from January 1 to July 19, 2021. At the time of analysis, 10 BPD Collaborative centers had contributed data that meets the study inclusion criteria

The dataset is structured around individual record_id entries representing premature infants. Key characteristics at birth include gender, corrected gestational age, originating center, birth measurements (weight, length, head circumference), and maternal characteristics such as race and ethnicity. Additional birth-related information, including delivery method, Prenatal Corticosteroids administration, Maternal Chorioamnionitis presence, and surfactant administration within the first 72 hours, is also captured.

Data at 36 and 44 weeks includes information on baby weight, Level of support, PEEP (Positive End-Expiratory and Pressure), Fraction of inspired O2, Peak inspiratory pressure, and medication administration for Pulmonary Hypertension. This dataset's primary outcomes of interest are whether infants underwent a tracheostomy at discharge and their mortality status.

3. Methodology

3.1. Analysis Plan

Given this study's dual outcomes of tracheostomy and death, we could make distinct models were constructed for each outcome. But in this study we will not create model for death outcome due to the small proportion of death cases in the dataset (only 5%). Concentrating on tracheostomy prediction was more urgent and feasible, as modeling rare events like death would require advanced techniques like resampling, which was beyond the scope of this study.

Initial steps involved an exploratory analysis of the dataset to identify visible patterns and aid in selecting significant variables pertinent to each outcome type. Furthermore, missing data was assessed, and suitable strategies for handling these gaps were determined.

Considering the many observed variables, a variable selection process was implemented to streamline the model, facilitating improved generalizability. The dataset was partitioned into two subsets – a training set and a testing set, with proportions of 0.7 and 0.3, respectively, to validate the model.

We will use two different types of dataset. One with only 36-week data and another including both 36-week and 44-week data, to determine if the 36-week data alone was sufficient for prediction or not. After that I will fit Lasso and Ridge regression model to both data. We regard the lasso and rigde model as the prediction model candidate and also as variables selection method. The selected variables from the best performing model will also be used to fit a multilevel model since we recognize that the dataset has multilevel structure. After that we will compare the performance of the Lasso, Ridge, and Multilevel model among the two dataset.

Given the binary nature of the outcomes, the performance of the models was assessed using the Area Under the Curve (AUC), F1 Score, Sensitivity, Specificity, and Precision. AUC is used to assess the general performance of the models in the dataset. Sensitivity is used to identifying true positives, crucial in avoiding missed tracheostomy needs. Precision is added to assess the ratio of true positives to predicted positives, an important factor considering the high cost of false positives leading to unnecessary procedures. Specificity is also included to ensure accurate identification of true negatives. F1 score is added to see the harmony of precision and sensitivity. This evaluation strategy ensures a comprehensive understanding of the model's accuracy and predictive capability.

3.2. Exploratory Data Analysis

The initial step involved checking for duplicate data, and we identified one record_id with duplicates, promptly removing them from the dataset. Moving on to missing data analysis, many missing values were observed, particularly in the dataset related to 44 weeks. The missing values for those variables reached approximately 40% for each, with the surfactant indicator also showing notable gaps. Upon closer inspection, it appeared that infants missing week 44 data were primarily those discharged before week 44. Discrepancies were also noted in surfactant administration, which was missing values across different centers. Centers 3, 5, and 12 had fewer missing values, while Centers 4 and 7 exhibited a higher likelihood. Divergent data recording practices were also apparent, with some centers showing low completion rates for 36 and 44-week data. Again, this reinforces our belief that the data is Missing-at-Random, and we think that multiple imputation is the appropriate method to impute the missing values.

Table 1: Missing Data Proportion for Each Variable

Variable	Observation Missing	Proportion Missing
inspired_oxygen.44	448	44.9799197
p_delta.44	448	44.9799197
weight_today.44	446	44.7791165
peep_cm_h2o.44	446	44.7791165
any_surf	433	43.4738956
ventilation_support_level.44	424	42.5702811
med_ph.44	424	42.5702811
com_prenat_ster	193	19.3775100
p_delta.36	128	12.8514056
hosp_dc_ga	124	12.4497992
peep_cm_h2o.36	117	11.7469880
weight_today.36	92	9.2369478
inspired_oxygen.36	92	9.2369478
blength	78	7.8313253
birth_hc	77	7.7309237
mat_chorio	62	6.2248996
mat_ethn	57	5.7228916
mat_race	56	5.6224900
prenat_ster	35	3.5140562
ventilation_support_level.36	30	3.0120482
med_ph.36	30	3.0120482
sga	15	1.5060241
center	10	1.0040161
gender	4	0.4016064
del_method	3	0.3012048
death	2	0.2008032

During univariate analysis, considering missing values, the dataset exhibited high imbalance, especially for death. Most infants did not undergo tracheostomy (85%) and were alive (95%). Most did not receive pulmonary hypertension medication at weeks 36 and 44 (93% and 83%). Prenatal corticosteroids were administered to most mothers (87%), and Maternal Chorioamnionitis was mostly absent (83%). Discharge after 44 weeks was predominant (64%), and most infants received surfactant in the first 72 hours (82%). Continuous data displayed outliers, such as unusually high birth weights (mean: 806, max: 2725), peak inspiratory pressures at 36 weeks (mean: 5.3, max: 46), peak inspiratory pressures at 44 weeks (mean: 7.6, max: 52), and discharge ages (mean: 52.8, max: 573.9).

Further examination identified variables with high correlation (above 75%), and one representative variable was retained to mitigate multicollinearity. We removed birth length and head circumference since they are highly correlated to birth weight. Additionally, variables with moderate correlation (around 35% - 75%) were noted for potential consideration, intending to incorporate ridge and lasso regularization during variable selection.

Summarizing covariates concerning outcomes (tracheostomy) revealed significant differences in some covariates between the two groups, suggesting potential confounders. Covariates with visually different boxplot distributions among outcome groups, such as gestational age and positive end-expiratory pressure (cm H2O) at 36 weeks, were considered potential predictors. Variables lacking significant differences, visual distinctiveness, and high correlation with other

variable were excluded from the initial variable selection. The variables dropped based on those reason are: maternal ethnicity, gestational age, birth length, birth head circumference, complete prenatal steroid, maternal chorioamnionitis, gender, surfactant, discharge time, death, and birth weight.

Below is the example of the summary table stratified by tracheostomy.

			Trach		
Variable	**N**	**Overall**, N = 996	**0**, N = 850	**1**, N = 146	**p-value**
center	986				
1		55 / 986 (5.6%)	32 / 844 (3.8%)	23 / 142 (16%)	
2		630 / 986 (64%)	566 / 844 (67%)	64 / 142 (45%)	
3		57 / 986 (5.8%)	56 / 844 (6.6%)	1 / 142 (0.7%)	
4		60 / 986 (6.1%)	49 / 844 (5.8%)	11 / 142 (7.7%)	
5		40 / 986 (4.1%)	35 / 844 (4.1%)	5 / 142 (3.5%)	
7		32 / 986 (3.2%)	31 / 844 (3.7%)	1 / 142 (0.7%)	
12		69 / 986 (7.0%)	34 / 844 (4.0%)	35 / 142 (25%)	
16		38 / 986 (3.9%)	37 / 844 (4.4%)	1 / 142 (0.7%)	
20		4 / 986 (0.4%)	4 / 844 (0.5%)	0 / 142 (0%)	
21		1 / 986 (0.1%)	0 / 844 (0%)	1 / 142 (0.7%)	
Unknown		10	6	4	
mat_race	940				0.010
0		538 / 940 (57%)	474 / 804 (59%)	64 / 136 (47%)	
1		290 / 940 (31%)	243 / 804 (30%)	47 / 136 (35%)	
2		112 / 940 (12%)	87 / 804 (11%)	25 / 136 (18%)	
Unknown		56	46	10	
mat_ethn	939				0.35
1		74 / 939 (7.9%)	66 / 803 (8.2%)	8 / 136 (5.9%)	
2		865 / 939 (92%)	737 / 803 (92%)	128 / 136 (94%)	
Unknown		57	47	10	
bw	996	806 (297)	814 (295)	761 (303)	0.004
ga	996	25.77 (2.14)	25.76 (2.14)	25.84 (2.13)	0.73
blength	918	32.5 (3.8)	32.6 (3.8)	32.0 (3.9)	0.15
Unknown		78	48	30	
birth_hc	919	23.19 (2.76)	23.22 (2.71)	22.99 (3.07)	0.12
Unknown		77	46	31	
del_method	993				0.035
1		285 / 993 (29%)	254 / 848 (30%)	31 / 145 (21%)	
2		708 / 993 (71%)	594 / 848 (70%)	114 / 145 (79%)	
Unknown		3	2	1	
prenat_ster	961				0.011
0		126 / 961 (13%)	118 / 830 (14%)	8 / 131 (6.1%)	
1		835 / 961 (87%)	712 / 830 (86%)	123 / 131 (94%)	
Unknown		35	20	15	
com_prenat_ster	803				0.97
0		193 / 803 (24%)	166 / 690 (24%)	27 / 113 (24%)	
1		610 / 803 (76%)	524 / 690 (76%)	86 / 113 (76%)	
Unknown		193	160	33	
mat_chorio	934				0.81
0		774 / 934 (83%)	662 / 800 (83%)	112 / 134 (84%)	
1		160 / 934 (17%)	138 / 800 (17%)	22 / 134 (16%)	
Unknown		62	50	12	
gender	992				>0.99
0		408 / 992 (41%)	348 / 846 (41%)	60 / 146 (41%)	
1		584 / 992 (59%)	498 / 846 (59%)	86 / 146 (59%)	

Unknown		4	4	0	
sga	981				0.005
0		778 / 981 (79%)	678 / 839 (81%)	100 / 142 (70%)	
1		203 / 981 (21%)	161 / 839 (19%)	42 / 142 (30%)	
Unknown		15	11	4	
any_surf	563				0.14
0		102 / 563 (18%)	93 / 488 (19%)	9 / 75 (12%)	
1		461 / 563 (82%)	395 / 488 (81%)	66 / 75 (88%)	
Unknown		433	362	71	
weight_today.36	904	2,121 (413)	2,132 (410)	2,024 (432)	0.025
Unknown		92	38	54	
ventilation_support_level.36	966				<0.001
0		117 / 966 (12%)	111 / 839 (13%)	6 / 127 (4.7%)	
1		589 / 966 (61%)	560 / 839 (67%)	29 / 127 (23%)	
2		260 / 966 (27%)	168 / 839 (20%)	92 / 127 (72%)	
Unknown		30	11	19	
inspired_oxygen.36	904	0.34 (0.15)	0.32 (0.13)	0.49 (0.20)	<0.001
Unknown		92	37	55	
p_delta.36	868	5 (10)	4 (9)	15 (12)	<0.001
Unknown		128	63	65	
peep_cm_h2o.36	879	6.3 (2.9)	6.2 (2.9)	7.5 (2.8)	<0.001
Unknown		117	59	58	
med_ph.36	966				<0.001
0		900 / 966 (93%)	798 / 839 (95%)	102 / 127 (80%)	
1		66 / 966 (6.8%)	41 / 839 (4.9%)	25 / 127 (20%)	
Unknown		30	11	19	
weight_today.44	550	3,646 (682)	3,667 (665)	3,550 (750)	0.23
Unknown		446	399	47	
ventilation_support_level.44	572				<0.001
0		269 / 572 (47%)	262 / 461 (57%)	7 / 111 (6.3%)	
1		146 / 572 (26%)	128 / 461 (28%)	18 / 111 (16%)	
2		157 / 572 (27%)	71 / 461 (15%)	86 / 111 (77%)	
Unknown		424	389	35	
inspired_oxygen.44	548	0.34 (0.15)	0.32 (0.13)	0.44 (0.19)	<0.001
Unknown		448	398	50	
p_delta.44	548	8 (14)	5 (12)	21 (16)	<0.001
Unknown		448	397	51	
peep_cm_h2o.44	550	4.3 (4.5)	3.4 (4.1)	8.7 (3.6)	<0.001
Unknown		446	396	50	
med_ph.44	572				<0.001
0		473 / 572 (83%)	413 / 461 (90%)	60 / 111 (54%)	
1		99 / 572 (17%)	48 / 461 (10%)	51 / 111 (46%)	
Unknown		424	389	35	
hosp_dc_ga	872	53 (27)	49 (24)	80 (30)	<0.001
Unknown		124	86	38	
death	994				<0.001
0		940 / 994 (95%)	811 / 848 (96%)	129 / 146 (88%)	
1		54 / 994 (5.4%)	37 / 848 (4.4%)	17 / 146 (12%)	
Unknown		2	2	0	

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

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

3.3. Data splitting

We split the data before applying any methods we planned to use. This is done to avoid data

leaking issues that would lead to overfitting. The data training and data testing proportions that will be used in this setting are 0.7 and 0.3. Also, we tried to separate the data that only contained 36-weeks data and the data that contained 36 and 44 weeks data because we seek to determine if the 36-week data alone was sufficient for prediction and also because we do not want to impute missing values in the 44 weeks from the data that does not have any 44 weeks of data. These data will undergo missing data imputation and variable selection separately.

3.4. Missing Data Impulation

In our dataset, missing values were predominantly assumed to follow a Missing-at-Random (MAR) pattern. Addressing this, we employed multiple imputation techniques, specifically using the 'mice' function from the MICE package. Our parameter settings for this function were m = 5 to generate five distinct imputed datasets, with other parameters left at their default settings. The imputation was separately conducted for each outcome-driven dataset (tracheostomy and death), further divided into subsets of 36-week data and a combination of 36 and 44-week data, each inclusive of base characteristics.

3.5. Variable and Model Selection Process

The process of selecting variables and models was methodical and based on preliminary findings from exploratory data analysis. The initial selection was guided by identifying variables that exhibited significant distributional differences or noticeable visual disparities across different outcome groups. We employed a dual approach using Lasso and Ridge regression techniques to refine the model selection. Lasso regression was chosen for its capacity to reduce the influence of less significant variables to zero, thereby simplifying the model. Ridge regression was utilized to address potential multicollinearity issues, a concern due to the suspected correlations between variables. This combination aimed to balance model simplicity and accuracy, ensuring both interpretability and generalizability of the model.

The optimal lambda values for both Lasso and Ridge regressions was determined through a cross-validation process. This process incorporated the five imputed datasets generated during our multiple imputation phase. By fitting a Lasso/Ridge regression model to each dataset, we accounted for the uncertainty inherent in the missing data, reducing bias in our final model. Cross-validation played a crucial role in this phase, helping to fine-tune the lambda parameter in both regression models, optimizing for minimal prediction error and exclusion of non-essential effects.

We calculated the AUC, F1, Sensitivity, Specificity, and Precision for both the Lasso and Ridge models using the test dataset. These results are presented in Figure 1 below. For tracheostomy outcomes, the Lasso model demonstrated superior performance in all the metrics compared to Ridge model in the dataset that include 44 week. Ridge perform better in some metrics like Sensitivity and F1 in the dataset that only has baseline and 36 week data. But overall, we see that Lasso in the 44 week dataset has the best perfromance. It seemed that only using

36 weeks data is not sufficient for making good prediction model. From the best lasso model, we found that there are two variables with 0 beta coefficient: small baby indicator and the delivery method. We excluded these variables in the multilevel model that we built after this process.

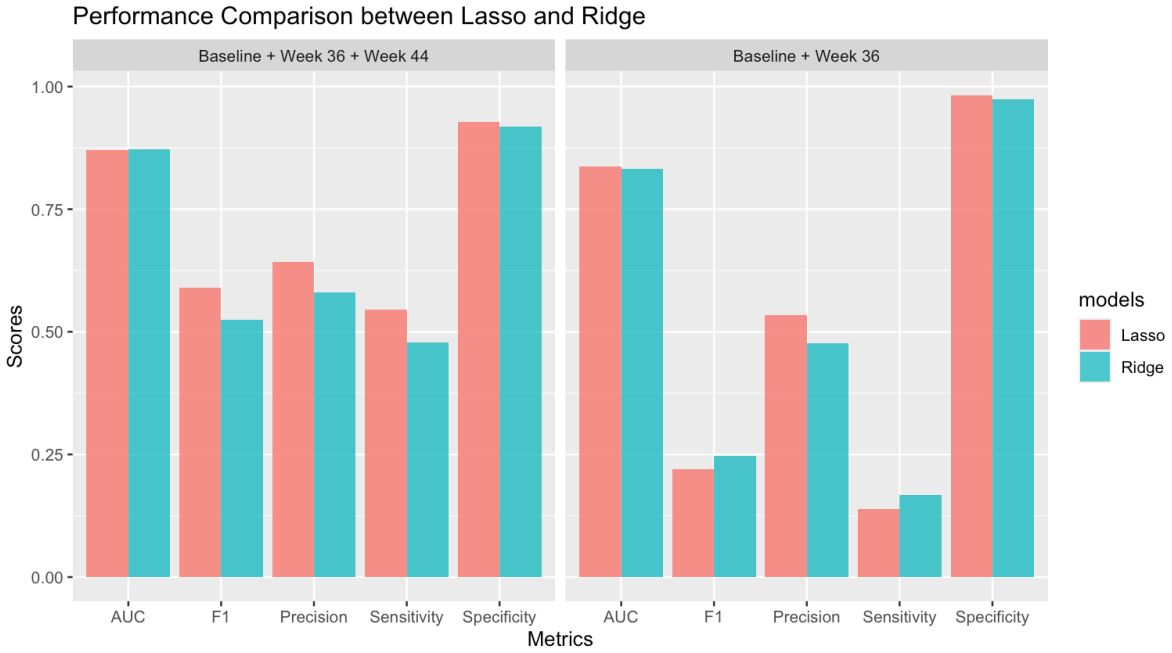


Figure 1: Metrics Comparison between Lasso and Ridge

In the development of our multilevel models, we incorporated significant variables identified from our previous lasso model analysis and introduced a random effect associated with medical centers. In addition, we incorporated time as the fixed effect. This model aimed to capture variations attributable to center-specific practices and characteristics.

Upon validation using the test dataset, we compare the perfromance of the multilevel model with the lasso and rigde. Although all three models showed good results in AUC and specificity (expected, because we have a lot of non event outcome), lasso performed best in sensitivity and precision – key for avoiding unnecessary or missed tracheostomies. Therefore, we chose the lasso model as the final model for this study. The best lasso model had 0.87 AUC, 0.59 F1 Score, 0.55 Sensitivity, 0.93 Specificity, and 0.64 Precision. We noted that our best model has mediocre sensitivity, precision, and F1 score. We suspected that it is due to the high imbalance in the outcome proportion in which we only had around 15% tracheostomy event happened.

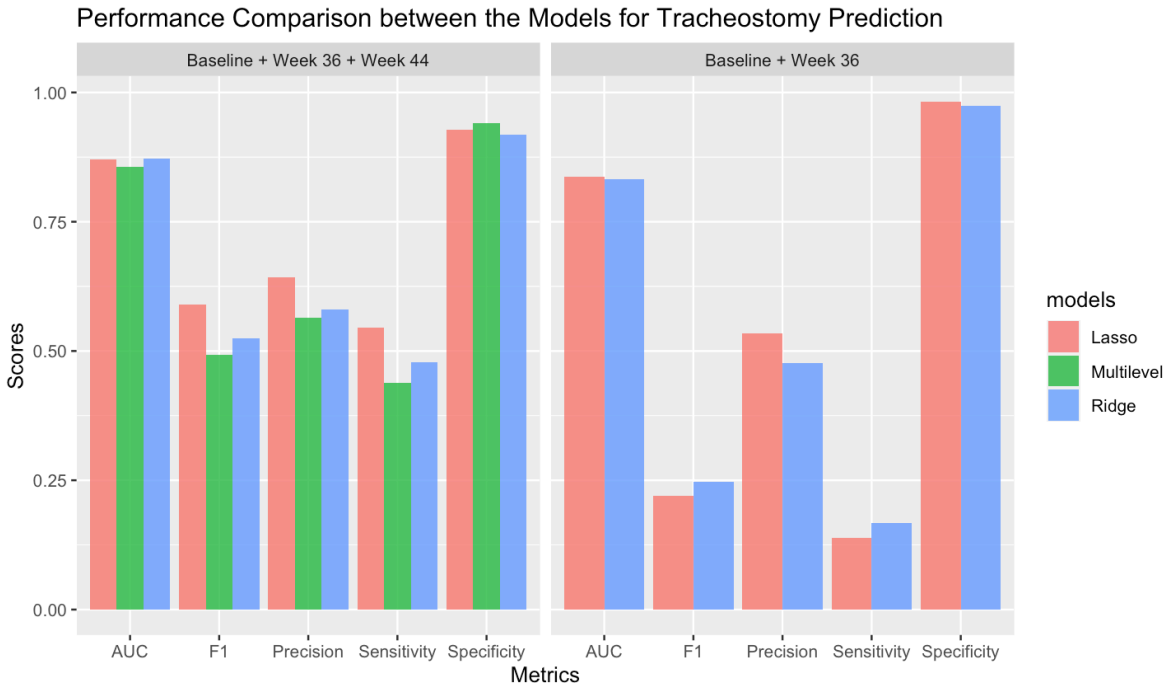


Figure 2: Metrics Comparison between Lasso and Ridge

4. Final model Result and Discussion

The lasso model was chosen as our final model. To finalize our analysis, we repeated the imputation process and the construction of the lasso model, this time employing the full dataset. The coefficients from our final model are listed and discussed in detail below.

Table 2: Estimate of Lasso Coefficients

	Estimate
prenat_ster1	2.175
ventilation_support_level.362	1.993
ventilation_support_level.diff	1.319
med_ph.361	1.259
mat_race2	1.237
inspired_oxygen.36	1.186
med_ph.diff	0.573
mat_race1	0.558
inspired_oxygen.diff	0.316
peep_cm_h2o.36	0.143
peep_cm_h2o.diff	0.067
p_delta.diff	0.000
p_delta.36	-0.014
ventilation_support_level.361	-0.016
weight_today.36	-0.053
weight_today.diff	-0.251
(Intercept)	-5.493

In the case of tracheostomy, our model identified several key characteristics associated with a higher odds of this intervention.

In the case of tracheostomy, our model identified several key characteristics associated with a higher likelihood of this intervention. Notably, Baby whose mother had Prenatal Corticosteroids has significantly higher odds of having tracheostomy compared to baby whose mother did not have Prenatal Corticosteroids. It is the same case with baby that has Invasive positive pressure support level on week 36, they have higher odds compared to baby who has non respiratory support at week 36. These findings suggest a complex interplay of institutional, prenatal, and clinical care factors in the likelihood of tracheostomy among neonates.

In conclusion, our models reveal that tracheostomy in neonates are influenced by a combination of hospital-specific factors, prenatal conditions, and specific medical interventions. The identification of these factors is crucial for understanding the clinical pathways and can guide targeted interventions and policy decisions to improve neonatal care outcomes.

5. Limitation

This study, while providing valuable insights, has several limitations that should be acknowledged.

1. Little to none Interaction Terms

Another limitation is the almost-absence of interaction terms in our model. We only tried to get the interaction terms between the time and the variables at 36 and 44 weeks but it could be that there is exist better interaction terms. Interaction terms can often reveal complex

interdependencies between variables that are not discernible when variables are considered independently. The inclusion of such interactions could potentially enhance the predictive accuracy and explanatory power of the model.

2. Limited Scope of Random Effects

Finally, the model's scope of random effects was restricted to only the medical center. This limited approach may overlook other significant random effects that could influence outcomes, such as patient demographics, staff characteristics, or temporal factors. Broadening the scope of random effects to include these additional variables could capture a more comprehensive range of influences, providing a more accurate and generalizable model.

In conclusion, while our study offers important findings, the aforementioned limitations suggest avenues for future research to enhance the model's comprehensiveness and accuracy.

Supplemental Material

Supplemental material can be seen in this github page

Code Appendix

```
##############
### SETUP ###
##############

library(formatR)

knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(message = F)
knitr::opts_chunk$set(warning = F)
knitr::opts_chunk$set(fig.align="center")
knitr::opts_chunk$set(fig.width=8, fig.height=6)

###############
### LIBRARY ###
###############

library(tidyverse)
library(ggplot2)
library(naniar)
library(gt)
library(gtsummary)
library(kableExtra)
library(GGally)
library(corrplot)
library(patchwork)
library(splitstackshape)
library(mice)
library(glmnet)
library(pROC)
library(lme4)
#####################################
### READ DATA AND PREPROCESSING ###
#####################################

# Set working directory and read data
setwd("/Users/amirahff/Documents/Brown Biostatistics/PHP 2550/project2")
raw_df <- read.csv("project2.csv")

# Check data type
```

```r
str(raw_df)

# Is there any duplicate data
# Yes, record_id = 2000824
raw_df %>%
  group_by(record_id) %>%
  count() %>%
  filter(n > 1)

# Remove duplicate data
df = raw_df %>%
  group_by(record_id) %>%
  distinct() %>%
  ungroup()

# Rename some columns and change data type
df = df %>%
  rename('peep_cm_h2o.36' = 'peep_cm_h2o_modified.36'
         ,'peep_cm_h2o.44' = 'peep_cm_h2o_modified.44'
         ,'ventilation_support_level.44' = 'ventilation_support_level_modified.44'
         ,'trach' = 'Trach'
         ,'death' = 'Death') %>%
  mutate(prenat_ster = case_when(prenat_ster=='Yes'~1,prenat_ster=='No'~0)
         ,com_prenat_ster = case_when(com_prenat_ster=='Yes'~1,com_prenat_ster=='No'~0)
         ,mat_chorio = case_when(mat_chorio=='Yes'~1,mat_chorio=='No'~0)
         ,gender = case_when(gender=='Male'~1,gender=='Female'~0)
         ,sga = case_when(sga=='SGA'~1,sga=='Not SGA'~0)
         ,any_surf = case_when(any_surf=='Yes'~1,any_surf=='No'~0)
         ,death = case_when(death=='Yes'~1,death=='No'~0)) %>%
  mutate(record_id = as.factor(record_id)
         , center = as.factor(center)
         , mat_race = as.factor(mat_race)
         , mat_ethn = as.factor(mat_ethn)
         , del_method = as.factor(del_method)
         , prenat_ster = as.factor(prenat_ster)
         , com_prenat_ster = as.factor(com_prenat_ster)
         , mat_chorio = as.factor(mat_chorio)
         , gender = as.factor(gender)
         , sga = as.factor(sga)
         , ventilation_support_level.36 = as.factor(ventilation_support_level.36)
         , med_ph.36 = as.factor(med_ph.36)
```

```
         , ventilation_support_level.44 = as.factor(ventilation_support_level.44)
         , med_ph.44 = as.factor(med_ph.44)
         , any_surf = as.factor(any_surf)
         , trach = as.factor(trach)
         , death = as.factor(death)) %>%
  mutate(across(where(is.numeric), round, 4))

####################
### MISSING DATA ###
####################

# Variables' missing data proportion
varMissingProp = miss_var_summary(df)
varMissingProp %>%
  filter(n_miss > 0) %>%
  kableExtra::kbl(caption = 'Missing Data Proportion for Each Variable'
                  , booktabs = T
                  , escape = T
                  , align = 'c'
                  , col.names = c('Variable','Observation Missing','Proportion
                                  Missing')) %>%
  kableExtra::kable_classic(full_width = F
                            , html_font = 'Cambria'
                            , font_size = 6
                            , latex_options = 'HOLD_position')
############################
### COMPARE TRACH GROUP ###
############################

#Summary by Trach
df %>%
  dplyr::select(-c(record_id)) %>%
  tbl_summary(by = trach
              , statistic = list(
                all_continuous() ~ "{mean} ({sd})"
                ,all_categorical() ~ "{n} / {N} ({p}%)"
                )) %>%
  add_p(pvalue_fun = ~ style_pvalue(.x, digits = 2)) %>%
  add_overall() %>%
  add_n() %>%
  modify_header(label ~ "**Variable**") %>%
```

```
  modify_spanning_header(c("stat_1", "stat_2") ~ "**Trach**") %>%
  bold_labels() %>%
  as_kable_extra(booktabs = TRUE, longtable = TRUE) %>%
  kableExtra::kable_classic(full_width = F
                            , html_font = 'Cambria'
                            , font_size = 7
                            , latex_options = 'scale_down')
load('final_coef.Rda')
round(final_coef,3) %>%
  arrange(desc(Estimate)) %>%
  kableExtra::kbl(caption = 'Estimate of Lasso Coefficients'
                  , booktabs = T
                  , escape = T
                  , align = 'c') %>%
  kableExtra::kable_classic(full_width = F
                            , font_size = 7
                            , html_font = 'Cambria'
                            , latex_options = 'HOLD_position')
```

Exploratory Analysis of Direct and Indirect Prenatal Smoking Exposure Effects on Children Externalizing Behavior

1. Introduction

The primary research goal of this report is to delve into the effects of smoking during pregnancy (SDP) and exposure to environmental tobacco smoke (ETS) on various aspects of adolescent development, specifically focusing on self-regulation, substance use, and externalizing behaviors. The data for this analysis were sourced from Dr. Lauren Micalizzi of the Brown University Department of Behavioral and Social Sciences. The study initially recruited a cohort of low-income pregnant women, totaling 738 individuals, as part of a smoke avoidance intervention program to reduce maternal smoking and ETS exposure during pregnancy. Furthermore, the study also assessed children's exposure to ETS in the immediate postpartum period. For this research, a subset of 100 adolescents and their mothers was randomly selected for recruitment, forming the core dataset.

The data can be broadly categorized into two sections: child and parent data to gain a comprehensive understanding of the adolescent and parent dynamics. The child section has essential demographic information, including race, age, and sex. Additionally, there are scores related to attention, internalizing, externalizing problems, and emotion regulation attributes such as cognitive reappraisal and expressive suppression. Moreover, the dataset contains information on child substance use and the parent-child relationship. The parent section mirrors many of the child's data categories. It also includes additional information such as the child's ADHD status, parental demographic details like income, employment, and education, and data related to smoking during pregnancy (SDP) and exposure to environmental tobacco smoke (ETS).

Despite the richness of the dataset, it's important to acknowledge certain limitations that may influence the scope and interpretation of the analysis findings. First, cotinine levels, which serve as a biomarker of nicotine exposure, were only measured at two-time points—34 weeks gestation and 6 months after birth. This limitation necessitates a reliance on self-reported data for nicotine exposure in other instances. Second, the data analyzed in this report are limited to baseline measurements for both children and parents. Therefore, this study does

not cover longitudinal analysis, limiting the ability to draw conclusions about changes over time. Finally, this research's inclusion criteria required data for both the parent and child, resulting in a reduced dataset of 49 parent-child pairs. This reduced sample size may impact the generalizability of the findings to a broader population.

2. Preliminary Analysis

In the initial stages of our research, a crucial process involved getting the raw research data ready for analysis. This meant performing tasks like aggregating answers from related questionnaires to create summarized scores (e.g., parental knowledge score). Furthermore, we combined various pieces of data to develop meaningful indicators (e.g., smoke exposure one year after a child's birth). We also removed irrelevant variables or overly complex data to compact the dataset. We determined the parent-child pairs using the `parent_id`, and any non-paired individuals were removed.

Following the initial data preprocessing, we were left with a dataset consisting of 49 records and 78 variables. As we delved into the data, we noticed certain anomalies in some variables that raised questions. For instance, within the variable indicating the biological sex of the parents, there was one record indicating a male. However, given that all respondents were mothers, it is likely that this occurrence resulted from a data entry error or a misunderstanding by the respondent. Additionally, we encountered an unusual entry in the income variable, with a record stating '250,000,' which appeared to be a potential data entry error. Moreover, some irregularities surfaced in the daily number of cigarettes smoked by mothers, with entries like '2 black and miles a day,' '44989,' and '20-25.' These data issues are noted for further fixing in our subsequent preprocessing steps, where we will also address the creation of new variables that can be derived from the existing dataset.

3. Missing Data

Furthermore, during our data examination, we observed that 54 variables contained some missing data, and 765 observations had missing values, roughly 20% of the data being incomplete. However, the extent of the missing data varied among the variables. Notably, five variables displayed a missing data proportion exceeding 50%. Four of these variables pertained to the number of cigarettes, marijuana, e-cigarettes, and alcohol consumed by the child in the last 30 days, with 90% or more of their data missing. However, we determined that this missing data could be explained by the fact that only a few children responded to the substance consumption questionnaire. On the other hand, one variable, the autism spectrum disorder indicator, showed a missing data proportion exceeding 50% without a clear explanation, leading us to exclude this particular data from our analysis. We considered the extent of missing data manageable for the remaining variables with varying degrees of missing data, ranging from 0% to 32%. However, we found that the missing data actually contributed by 8 IDs who had more than 40 missing variables (exceeding 50% of the total variables), including critical information

like externalizing behavior scores and self-reported substance usage. We decided to dropped these IDs.

4. Enhancing Dataset

Considering our initial exploration of the data, we have decided to introduce several new variables to enhance our dataset's utility. These new variables consolidate related variables, simplify the values of certain existing variables, and rectify anomalies identified during our preliminary analysis.

One key addition is the variable `num_substance_used`, which summarizes the number of different substances consumed by the children from the four types present in the dataset (cigarettes/e-cigarettes, marijuana, alcohol). For instance, if a child has consumed cigarettes and alcohol, their `num_substance_used` would be recorded as 2. A parallel variable, `pnum_substance_used`, was created to capture a similar summary for the parents.

In relation to smoking during pregnancy (SDP), we introduced variables such as `mom_prenatal_smoke` to indicate if the mother self-reported smoking during any of the follow-ups in gestational weeks. A similar set of variables, `mom_postnatal_smoke`, represents smoking during the postnatal period. To gauge the intensity of mom SDP, we created `mom_prenatal_smoke_consistency` and `mom_postnatal_smoke_consistency`, which count how many times the mother reported smoking during prenatal and postnatal follow-ups, respectively. Furthermore, we recorded the pattern of maternal smoking during both prenatal and postnatal periods in the variable `mom_smoke_pattern`. For example, if a respondent reported smoking during the prenatal period but not during the postnatal period, they would have a value of '1 0' in this column. A parallel set of variables was created to represent environmental tobacco smoke (ETS) exposure. For the ETS we use the self-reported smoke exposure from year 1 through year 5.

5. Univariate Analysis

We conduct univariate analysis for most of the variables used in this study we start with a sanity check using summary statistics (minimum, maximum, median, 1st quartile, 3rd quartile) to get an initial idea of their distribution. If anything unusual is observed, we proceed to plot their distribution. The study sample primarily comprises individuals from Hispanic/Latino and White ethnic backgrounds (children and parents alike), reflecting a diverse population. Most children are male and in the middle school age range, indicating a focus on this particular demographic for adolescent development research. Regarding the parents of the children under study, they usually had the children in the study in their young adult phase (20-25). Most of the parents in the dataset have education levels exceeding high school. While the parents are predominantly employed, their income levels tend to fall below the threshold of $38,133.

Both children and parents exhibit similar distribution patterns in attention problem scores, internalizing problem scores, and externalizing scores, which are right-skewed. This implies that most children and parents scored relatively low on these metrics. Additionally, they share a similar distribution in cognitive reappraisal scores, which is left-skewed. However, there is a difference between children and parents when it comes to expressive suppression. Children's scores show a left-skewed distribution, while parents' scores are right-skewed. It means that children are more likely to have expressive suppression compared to the parents.

Regarding ADHD, the inattentive type exhibits a left-skewed distribution, while the hyperactive type is right-skewed. Furthermore, both children and parents display a right-skewed distribution in the number of substance types used, indicating that most participants consumed relatively few types of substances.

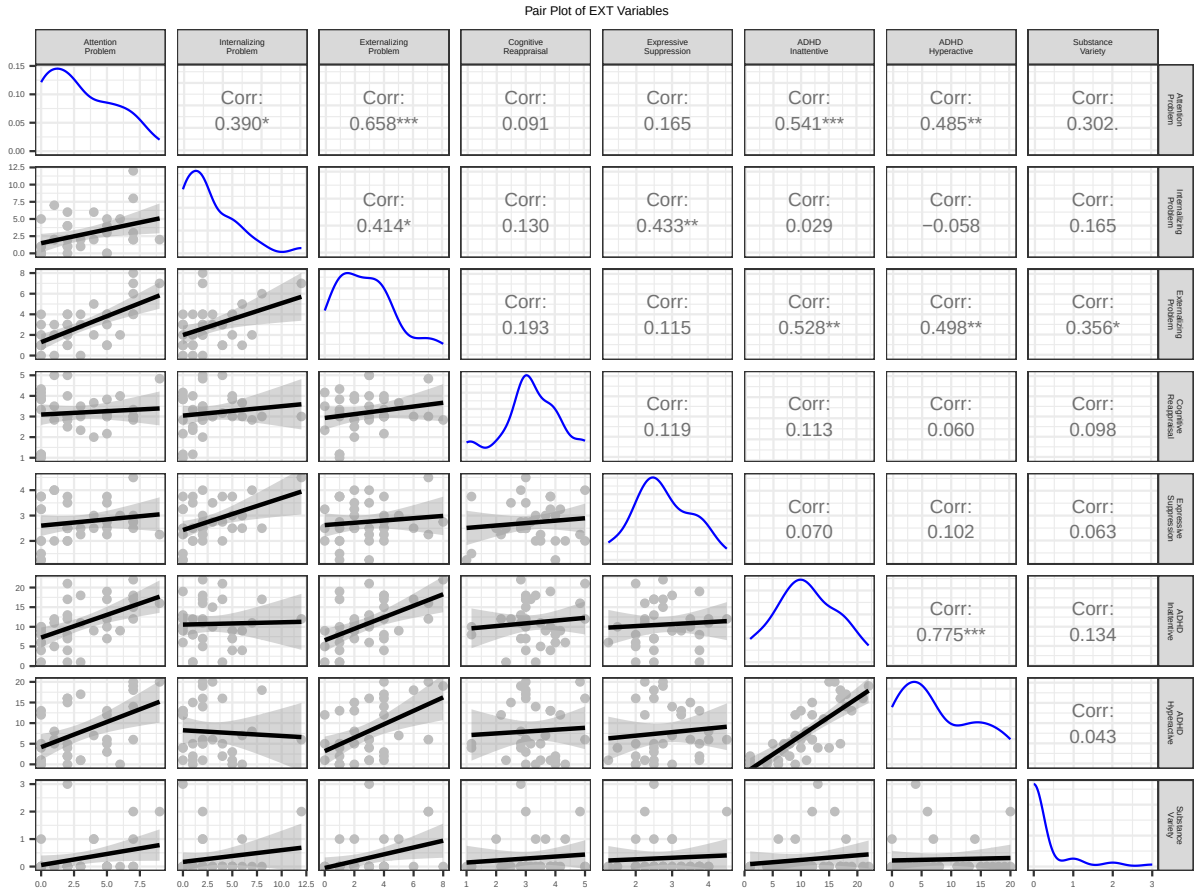
However, it's worth noting that there were minimal reports of substance usage among children, with alcohol being the most reported, totaling 5 records. In contrast, a larger proportion of parents reported consuming both tobacco and alcohol.

The parental monitoring attributes within the dataset exhibit a left-skewed distribution, whether viewed from the children's or the parent's perspective. This finding suggests a harmonious condition in the families, with both displaying a similar level of agreement regarding monitoring behaviors.

Regarding smoking during pregnancy (SDP) and exposure to environmental tobacco smoke (ETS), the data trends show a slight inclination towards "no smoking" and "no smoking exposure" during both prenatal and postnatal periods. However, it is worth noting an interesting observation: there are more reports of "no exposure" compared to "mom smoking" claims. This phenomenon raises questions about the accuracy of the exposure data. Some errors in data entry or misclassification within the exposure data may contribute to this discrepancy between reported maternal smoking and reported exposure levels. Further investigation may be needed to reconcile these disparities and ensure the reliability of the exposure data.

6. Interrelatedness between Children Externalizing Behavior

In our analysis, we delved into the correlation between various Externalizing Behaviors variables, including Attention Problem Score, Internalizing Problem Score, Externalizing Problem Score, Cognitive Reappraisal, Expressive Suppression, ADHD Inattentive SWAN Score, ADHD Hyperactive SWAN Score, and Number of Substance Types Used. Given the nature of these variables, it was anticipated that some of them would be interrelated, such as the potential correlation between ADHD Inattentive and Attention Problem Score or between Internalizing Problems and Expressive Suppression. To explore these relationships, we employed pair plots as seen in Figure 1.


Pair Plot of EXT Variables
Attention Problem
Internalizing Problem
Externalizing Problem
Cognitive Reappraisal
Expressive Suppression
ADHD Inattentive
ADHD Hyperactive
Substance Variety
Corr: 0.390*
Corr: 0.658***
Corr: 0.091
Corr: 0.165
Corr: 0.541***
Corr: 0.485**
Corr: 0.302.
Corr: 0.414*
Corr: 0.130
Corr: 0.433**
Corr: 0.029
Corr: −0.058
Corr: 0.165
Corr: 0.193
Corr: 0.115
Corr: 0.528**
Corr: 0.498**
Corr: 0.356*
Corr: 0.119
Corr: 0.113
Corr: 0.060
Corr: 0.098
Corr: 0.070
Corr: 0.102
Corr: 0.063
Corr: 0.775***
Corr: 0.134
Corr: 0.043

Our analysis revealed notable correlations among these variables: 1) A strong correlation was observed between ADHD Inattentive and ADHD Hyperactive, with both also showing correlations between Attention and Externalizing Problems. This implies that children with high scores in one problem area will likely have elevated scores in related problems. Also, it means that most of the case, children have both ADHD Inattentive and ADHD Hyperactive.
2) We found that both the Attention Problem and Externalizing Problem correlated with the Internalizing Problem, indicating a potential link between these variables.
3) Our analysis confirmed our initial assumption that the Internalizing Problem has a substantial correlation with Expressive Suppression.
4) The variable significantly correlated with the Number of Substance Types Used was the Externalizing Problem.
5) However, it's important to note that Cognitive Reappraisal did not exhibit any significant correlation with the other variables, indicating its distinct nature in the context of this study. Based on these findings, it becomes apparent that externalizing behaviors are interconnected, suggesting a common underlying issue contributing to their correlation.

7. Relationship between Children Externalizing Behavior, Smoking During Pregnancy, and Other Variables.

In our pursuit to address the primary goal of this analysis, we conducted an examination of the relationship between variables related to smoking during pregnancy (SDP) and exposure to environmental tobacco smoke (ETS) with the variables linked to externalizing behaviors (EXT). To illustrate these relationships, we initiated our investigation by comparing the characteristics of EXT variables across distinct groups (e.g., children exposed to smoke during prenatal stages versus those not exposed). The main variables used in this analysis are Mom Smoking (Prenatal) Indicator and Prenatal Smoke Exposure (Only prenatal are chosen since smoking during pregnancy refers to prenatal period). To facilitate this comparison, we constructed summary tables that provided a comprehensive view of key statistics, including medians, interquartile ranges, and p-values for each group. Wilcoxon rank sum test for 2 levels group are used to obtain the p-values. It is crucial to note that the p-values obtained from this analysis are based on a relatively small sample size. Therefore, it is advisable to consider larger sample sizes for more precise p-value estimations.

The results of our investigation unveiled several noteworthy relationships:
1) ADHD Hyperactive scores connected with Mom Smoking (Prenatal) suggest maternal smoking during pregnancy may be associated with children's hyperactivity. Children with smoking moms (prenatal) have 13 ADHD Hyperactive scores in the median compared to 5 ADHD Hyperactive scores in the median.
2) Internalizing problem scores were found to be related to Prenatal Smoke Exposure, with the group that had prenatal smoke exposure scoring 3 in the median and the non-exposed group scoring 1 in the median.
3) Externalizing problem scores exhibited relationships with Prenatal Smoke Exposure, with

Variable	N	Overall, N = 38	Maternal Prenatal Smoking		p-value
			0, N = 24	1, N = 14	
Attention Problem	29	3.00 (1.00, 5.00)	2.50 (0.25, 5.00)	5.00 (2.00, 7.00)	0.24
Internalizing Problem	27	2.00 (0.00, 4.00)	2.00 (0.75, 4.25)	2.00 (0.00, 3.50)	0.92
Externalizing Problem	29	3.00 (1.00, 4.00)	2.50 (1.00, 4.00)	3.00 (1.50, 5.00)	0.49
Cognitive Reappraisal	28	3.00 (2.83, 3.88)	3.00 (2.33, 3.88)	3.00 (3.00, 3.83)	0.41
Expressive Suppression	28	2.63 (2.25, 3.31)	2.50 (2.00, 3.00)	3.25 (2.50, 3.63)	0.084
ADHD Inattentive	30	11.5 (9.0, 15.8)	11.0 (8.0, 14.5)	13.0 (10.0, 16.5)	0.39
ADHD Hyperactive	30	6.0 (3.3, 13.0)	5.0 (1.5, 9.5)	13.0 (6.0, 17.0)	0.025
Substance Variety	38	0.00 (0.00, 0.00)	0.00 (0.00, 0.00)	0.00 (0.00, 0.00)	0.36

[1] Median (IQR)

[2] Wilcoxon rank sum test

Variable	N	Overall, N = 42	Prenatal Environmental Smoking Exposure		p-value
			0, N = 27	1, N = 15	
Attention Problem	35	2.00 (1.00, 5.00)	2.00 (0.00, 4.50)	4.00 (2.00, 7.00)	0.072
Internalizing Problem	33	2.00 (1.00, 4.00)	1.00 (0.00, 3.50)	3.00 (2.00, 5.00)	0.012
Externalizing Problem	35	3.00 (1.00, 4.00)	2.00 (1.00, 3.00)	4.00 (3.00, 4.00)	0.009
Cognitive Reappraisal	34	3.08 (2.88, 3.83)	3.33 (2.83, 4.00)	3.00 (3.00, 3.42)	0.97
Expressive Suppression	34	2.63 (2.25, 3.44)	2.50 (2.00, 3.13)	3.00 (2.50, 3.75)	0.058
ADHD Inattentive	39	11.0 (7.5, 15.0)	10.0 (7.5, 13.5)	12.0 (9.3, 17.0)	0.36
ADHD Hyperactive	39	6.0 (2.5, 13.0)	5.0 (3.0, 12.0)	6.0 (2.8, 16.3)	0.36
Substance Variety	42	0.00 (0.00, 0.00)	0.00 (0.00, 0.00)	0.00 (0.00, 0.00)	0.90

[1] Median (IQR)

[2] Wilcoxon rank sum test

Variable	N	Overall, N = 43	Postnatal Environmental Smoking Exposure		p-value
			0, N = 26	1, N = 17	
Attention Problem	35	2.00 (1.00, 5.00)	2.00 (0.00, 4.75)	3.00 (2.00, 5.00)	0.16
Internalizing Problem	33	2.00 (1.00, 4.00)	2.00 (0.00, 4.00)	2.50 (2.00, 4.25)	0.14
Externalizing Problem	35	3.00 (1.00, 4.00)	2.00 (1.00, 4.00)	3.00 (2.00, 4.00)	0.21
Cognitive Reappraisal	34	3.00 (2.83, 3.83)	3.33 (2.83, 4.00)	3.00 (3.00, 3.58)	0.76
Expressive Suppression	34	2.50 (2.25, 3.44)	2.50 (2.00, 3.00)	3.50 (2.50, 3.75)	0.011
ADHD Inattentive	38	10.5 (7.3, 14.8)	9.0 (7.0, 12.8)	12.0 (10.8, 17.0)	0.12
ADHD Hyperactive	38	6.0 (2.3, 13.0)	5.0 (2.3, 11.8)	9.5 (5.0, 16.3)	0.14
Substance Variety	43	0.00 (0.00, 0.00)	0.00 (0.00, 0.00)	0.00 (0.00, 0.00)	0.54

[1] Median (IQR)

[2] Wilcoxon rank sum test

the group with prenatal smoke exposure scoring 4 in the median and the non-exposed group scoring 2 in the median.
5) Expressive Suppression scores displayed connections with Smoke Exposure (Postnatal). The group with postnatal smoke exposure scored 3.5 in the median, and the non-exposed group scored 2.5 in the median.

It seems that both prenatal smoke exposure and mom smoking during prenatal has an impact on Externalizing Behavior such as ADHD Hyperactive, Internalizing Problem, Externalizing Problem, and Expressive Suppression. Indirectly, it could impact substance use, too, since Externalizing Problems and Substance Use are related. We can conclude that the group exposed to smoke (from mom or secondary exposure) was associated with higher externalizing behavior scores and, therefore, more prone to worse externalizing behavior.

We also compared the Externalizing Behavior of different races (children & parents), sexes (children), higher education indicators, income levels, and employment status using the same method. Groups associated with higher attention problem scores and internalizing problems compared to other races children are white children. Surprisingly, children with part-timing moms have lower internalizing problem scores than other groups. It could be because that group only has a small sample size (7 children), and we need more samples to get more accurate conclusions. These relationship can be seen in the Figure 2.

Table 1: Correlation Between EXT and Non EXT Variables

Other Variables	Externalizing Behavior Variables	Correlation
Parent Attention Problem	Substance Variety	0.44
Maternal Postnatal Smoking Consistency	Substance Variety	0.38
Parental Knowledge	Substance Variety	-0.56
Child Disclosure	Substance Variety	-0.40
Parental Control	Substance Variety	-0.37
Prenatal Environmental Smoking Exposure Consistency	Internalizing Problem	0.36
Parent Attention Problem	Externalizing Problem	0.56
Parent Internalizing Problem	Externalizing Problem	0.39
Prenatal Environmental Smoking Exposure Consistency	Expressive Suppression	0.39
Postnatal Environmental Smoking Exposure Consistency	Expressive Suppression	0.54
Parent Attention Problem	Attention Problem	0.60
Parent Internalizing Problem	Attention Problem	0.37
Parent Attention Problem	ADHD Inattentive	0.36
Parent Internalizing Problem	ADHD Inattentive	0.37
Child Disclosure	ADHD Inattentive	-0.40
Parent Attention Problem	ADHD Hyperactive	0.37
Parent Internalizing Problem	ADHD Hyperactive	0.53
Maternal Prenatal Smoking Consistency	ADHD Hyperactive	0.41

Furthermore, we also want to explore the other continuous variables influencing externalizing behavior. We want to know if factors such as the age gap between parents and children could impact these scores or if there might be an inheritance of externalizing behavior from parents to children. Additionally, we aimed to investigate whether the consistency of smoking behavior in the prenatal and postnatal periods affected externalizing behavior. To accomplish

Child Internalizing Problem Score vs SDP & Other Potential Confounders

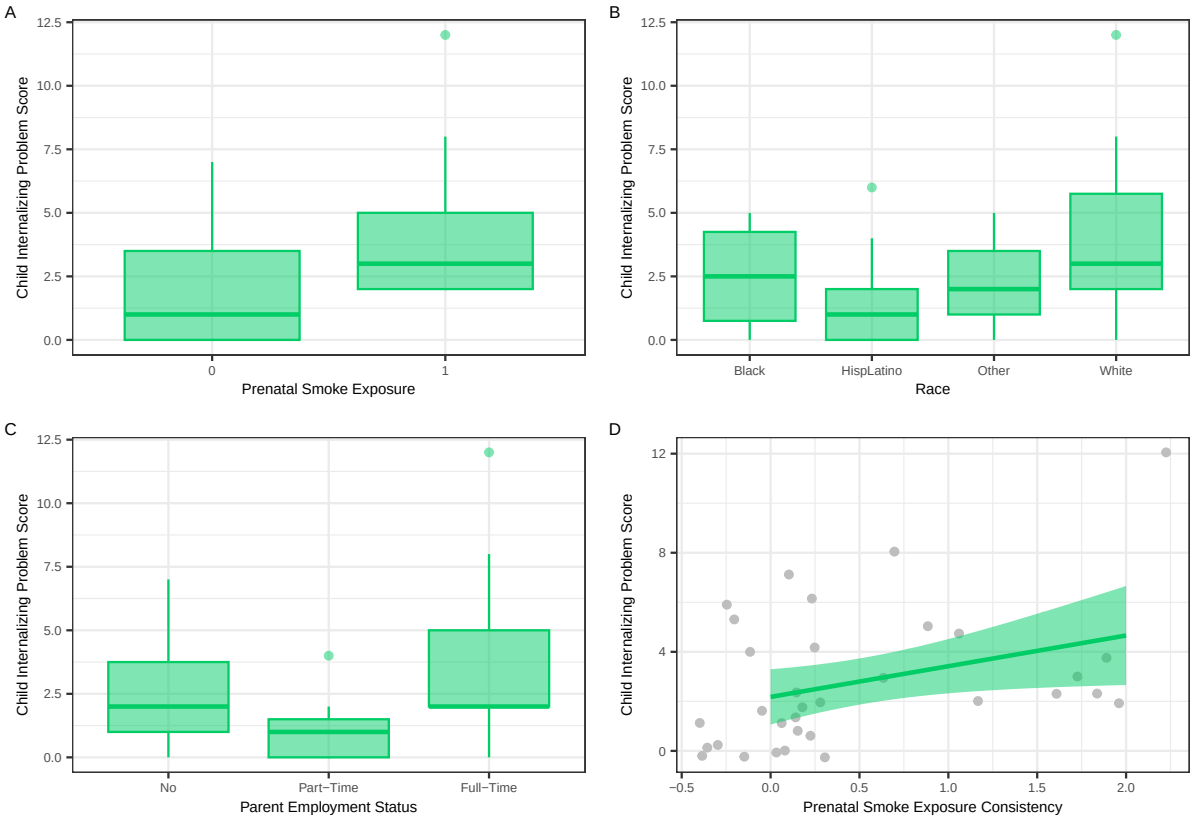


Figure 2. Child Internalizing Problem Score vs SDP & Other Potential Confounders

these objectives, we conducted correlation analyses between externalizing behavior variables and the other continuous variables in our dataset. The result can be seen in Table 3.

Our analysis revealed intriguing findings. We observed multiple correlations between parental attention and internalizing problem scores with children's externalizing behavior variables. This suggests that parental attributes related to attention and internalizing problems influence the externalizing behavior of their children. Similarly, we identified a correlation between smoking behavior consistency and externalizing behavior, implying that the consistency of parental smoking habits in prenatal and postnatal may play a role in shaping their children's externalizing behavior. Also, we could see that the higher the parental knowledge, child disclosure, and parental control, the children are less likely to try any substances. We can see this relationships in the Figure 3.

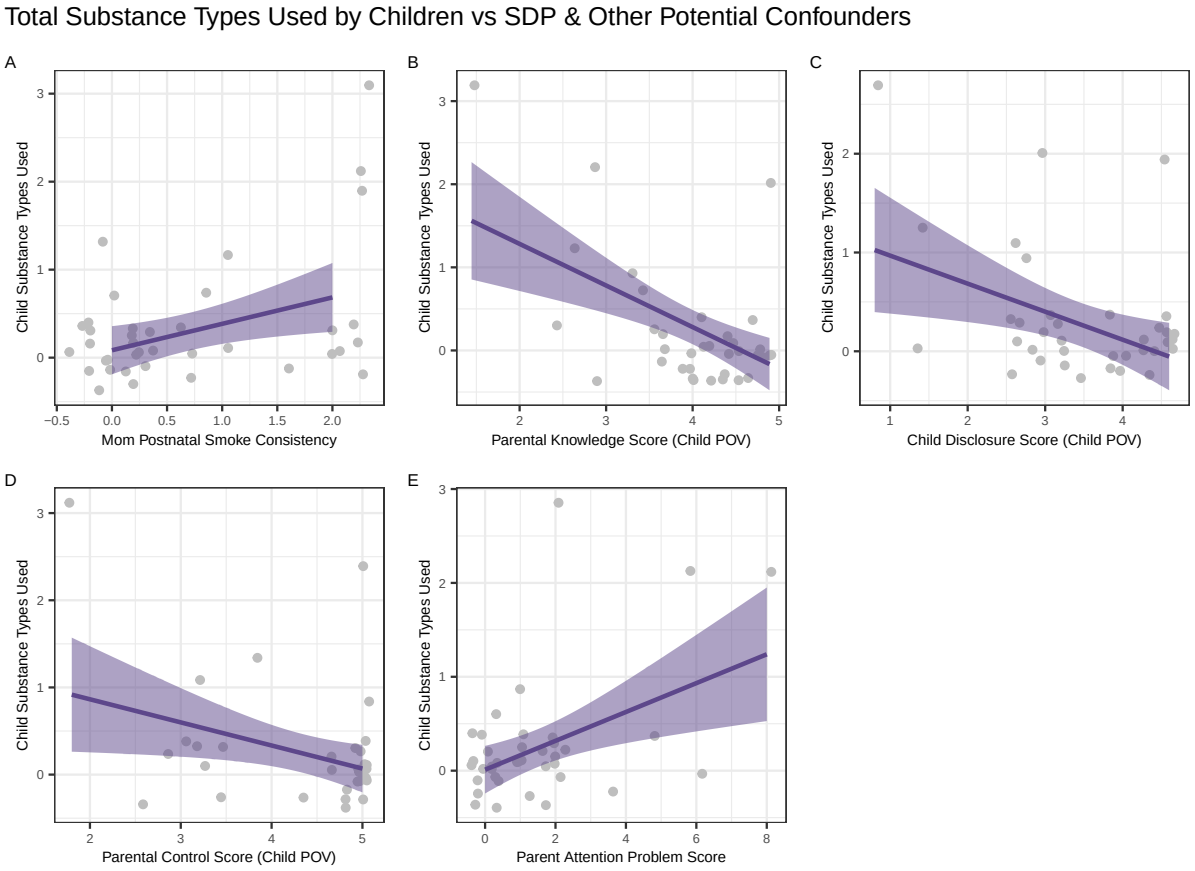


Figure 3. Total Substance Types Used by Children vs SDP & Other Potential Confounders

8. Conclusion

Based on our exploratory analysis, it becomes evident that smoking during pregnancy influences certain aspects of externalizing behavior. In general, exposure to smoking during pregnancy, whether from mother smoking or secondary exposure within the environment, tends to correlate with elevated externalizing behavior scores among children. Additionally, we observed a positive correlation between parental externalizing behavior and that of their children. However, it's important to acknowledge that the sample size utilized in this analysis is relatively small. To gain a more comprehensive understanding of the relationship between externalizing behavior scores and potential confounding factors, acquiring larger sample sizes would facilitate more extensive analyses, including regression analyses. Such analyses could provide deeper insights into the intricate relationships between externalizing behavior and the various factors in this context.

Supplemental Material

Supplemental material can be seen in this github page

Code Appendix

```
##############
### SETUP ###
##############

library(formatR)

knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(message = F)
knitr::opts_chunk$set(warning = F)
knitr::opts_chunk$set(fig.align="center")
knitr::opts_chunk$set(fig.width=8, fig.height=6)

###############
### LIBRARY ###
###############

library(tidyverse)
library(ggplot2)
library(naniar)
library(gt)
library(gtsummary)
library(kableExtra)
library(GGally)
library(corrplot)
library(patchwork)
############################
### PREPROCESS RAW DATA ###
############################

# first process child data
setwd("/Users/amirahff/Documents/Brown Biostatistics/PHP 2550/project1")
child_df <- read.csv("K01BB.csv")

child_df <- child_df %>%
  select(c(participant_id:su_interview_complete))  %>%
  filter(redcap_event_name == "child_baseline_arm_1")

# select demographic variables
child_df <- child_df %>%
```

```
  select(-c(participant_id, part, lastgrade, redcap_event_name, famid,
            visit_date, time, redcap_survey_identifier, enroll_timestamp,
            handednesst, tgender, sexorient, whichlang, nativelang, traceoth,
            usborn, relation, guardian, livewith___0:livewith___7,
            attendance, demographics_complete, langpref, pacemaker,
            longlive)) %>%
  rename(taian = trace___0, tasian = trace___1, tnhpi = trace___2,
         tblack = trace___3, twhite = trace___4, trace_other = trace___5)

# drop brief because scoring difficult
child_df <- child_df %>%
  select(-c(brief_ysr_timestamp:brief_ysr_complete))

# cigarette usage summarize
child_df <- child_df %>%
  mutate(cig_ever = suc1, num_cigs_30 = suc11) %>%
  select(-c(suc1:honc10))

# e-cig usage summarize
child_df <- child_df %>%
  mutate(e_cig_ever = ecig1, num_e_cigs_30 = ecig4) %>%
  select(-c(ecig1:ehonc10))

# marijuana usage summarize
child_df <- child_df %>%
  mutate(mj_ever = mj1, num_mj_30 = mj8) %>%
  select(-c(mj1:mpi29))

# alchohol usage summarize
child_df <- child_df %>%
  mutate(alc_ever = alc2, num_alc_30 = alc7) %>%
  select(-c(alc1:alcsus3))

# other drugs and norms - dropping
child_df <- child_df %>%
  select(-c(odrg1:othdrglist,
            perceived_norms_peers_timestamp:perceived_norms_peers_complete,
            substance_use_cigarettes_timesta:substance_use_other_drug_use_com))

# brief problem monitor scoring
child_df <- child_df %>%
```

```
  mutate(bpm_att = rowSums(dplyr::select(., c(bpm1,bpm3,bpm4,bpm5,bpm10))),
         bpm_ext = rowSums(dplyr::select(., c(bpm2,bpm6,bpm7,bpm8,bpm15,
                                              bpm16,bpm17))),
         bpm_int = rowSums(dplyr::select(., c(bpm9,bpm11,bpm12,bpm13,bpm18,
                                              bpm19)))) %>%
  select(-c(brief_problem_monitor_timestamp:brief_problem_monitor_complete))

# emotional regulation
child_df <- child_df %>%
  mutate(erq_cog = rowMeans(dplyr::select(., c(erq1,erq3,erq5,erq7,
                                               erq8,erq10))),
         erq_exp = rowMeans(dplyr::select(., c(erq2,erq4,erq6,
                                               erq9)))) %>%
  select(-c(emotion_regulation_questionnaire:emotion_regulation_questionnair1))

# physical - dropping for the purpose of this research
child_df <- child_df %>%
  select(-c(physical_development_scale_ysr_t:physical_development_scale_ysr_c,
            height1:body_measurements_complete))

# life stress - dropping for the purpose of this research
child_df <- child_df %>%
  select(-c(life_stress_ysr_timestamp:life_stress_ysr_complete))

# parental monitoring scoring
child_df <- child_df %>%
  mutate(pmq_parental_knowledge = (pmq1+pmq2+pmq3+pmq4+pmq5+pmq6+
                                     pmq7+pmq8+(5-pmq9))/9,
         pmq_child_disclosure = (pmqcd1+pmqcd2+(5-pmqcd3)+(5-pmqcd4)+pmqcd5)/5,
         pmq_parental_solicitation = rowMeans(dplyr::select(., pmqps1:pmqps5)),
         pmq_parental_control = rowMeans(dplyr::select(., pmqpc1:pmqpc5))) %>%
  select(-c(parental_monitoring_questionnair:parental_monitoring_questionnai1))

# dysregulation - drop to simplify analysis
child_df <- child_df %>%
  select(-c(dysregulation_inventory_ysr_time:dysregulation_inventory_ysr_comp))

# early adolescent temperament - drop to simplify analysis
child_df <- child_df %>%
  select(-c(early_adolescent_temperament_que:early_adolescent_temperament_qu1))
```

```r
# alcohol and substance abuse - too few observed so remove
child_df <- child_df %>%
  select(-c(miniaud1:minikid_sud_2_complete))

# remove remaining diet questions for purposes of this research
child_df <- child_df %>%
  select(-c(intuitive_eating_scale_timestamp:su_interview_complete))

# parent df
parent_df <- read.csv("K01BB.csv") %>%
  filter(redcap_event_name == "parent_baseline_arm_2") %>%
  select(c(parent_id, page:chart23))

# demographics
parent_df <- parent_df %>%
  select(-c(pgender, marstat, handednessp, plang1:plang3,
            praceoth, ppacemaker, pusa, pedu1:pedu3,
            prelation:parent_demographics_complete, govtasst___0:govtasst___5,
            parent_demographics_asd_timestam,
            parent_demographics_asd_complete)) %>%
  rename(paian = prace___0, pasian = prace___1, pnhpi = prace___2,
         pblack = prace___3, pwhite = prace___4, prace_other = prace___5)

# brief - dropping for difficulty scoring
parent_df <- parent_df %>%
  select(-c(brief_p_on_c_timestamp:brief_p_on_c_complete))

# swan - p on c
parent_df <- parent_df %>%
  mutate(swan_inattentive = rowSums(dplyr::select(., swan1:swan9),
                                    na.rm=TRUE),
         swan_hyperactive = rowSums(dplyr::select(., swan10:swan18),
                                    na.rm=TRUE)) %>%
  select(-c(swan_p_on_c_timestamp:swan_p_on_c_complete))

# connors - drop because swan will be similar
parent_df <- parent_df %>%
  select(-c(connors_p_on_c_timestamp:connors_p_on_c_complete))

# pbpm - parent answering about child
parent_df <- parent_df %>%
```

```
  mutate(bpm_att_p = rowSums(dplyr::select(., c(pbpm1,pbpm3,pbpm4,pbpm5,pbpm10))),
         bpm_ext_p = rowSums(dplyr::select(., c(pbpm2,pbpm6,pbpm7,pbpm8,pbpm15,
                                                pbpm16,pbpm17))),
         bpm_int_p = rowSums(dplyr::select(., c(pbpm9,pbpm11,pbpm12,pbpm13,pbpm18,
                                                pbpm19)))) %>%
  select(-c(bpm_p_on_c_timestamp:bpm_p_on_c_complete))

# alc and drug use
parent_df <- parent_df %>%
  mutate(magic2 = ifelse(magic1 == 0, 0, magic2),
         magic5 = ifelse(magic4 == 0, 0, magic5),
         smoke_exposure_6mo = pmax(magic2, magic5),
         magic8 = ifelse(magic7 == 0, 0, magic8),
         magic11 = ifelse(magic10 == 0, 0, magic11),
         smoke_exposure_12mo = pmax(magic8, magic11),
         magic14 = ifelse(magic13 == 0, 0, magic14),
         magic17 = ifelse(magic16 == 0, 0, magic17),
         smoke_exposure_2yr = pmax(magic14, magic17),
         magic20 = ifelse(magic19 == 0, 0, magic20),
         magic23 = ifelse(magic22 == 0, 0, magic23),
         smoke_exposure_3yr = pmax(magic20, magic23),
         magic26 = ifelse(magic25 == 0, 0, magic26),
         magic29 = ifelse(magic28 == 0, 0, magic29),
         smoke_exposure_4yr = pmax(magic26, magic29),
         magic32 = ifelse(magic31 == 0, 0, magic32),
         magic35 = ifelse(magic34 == 0, 0, magic35),
         smoke_exposure_5yr = pmax(magic32, magic35)
          ) %>%
  select(-c(nidaliftetime___1:inject,penncig2:penn_state_ecigarette_dependenc1,
            penn_state_cigarette_dependence_,
            nida_quick_screen_timestamp,
            nida_quick_screen_complete, magic_timestamp:magic_complete)) %>%
  rename(mom_numcig = penncig1)



# brief - dropping because difficulty scoring
parent_df <- parent_df %>%
  select(-c(briefa_timestamp:briefa_complete))

# parental monitoring - parent answering on child
```

```
parent_df <- parent_df %>%
  mutate(ppmq_parental_knowledge = (ppmq1+ppmq2+ppmq3+ppmq4+ppmq5+ppmq6+
                                      ppmq7+ppmq8+(5-ppmq9))/9,
         ppmq_child_disclosure = (ppmqcd1+ppmqcd2+(5-ppmqcd3)+(5-ppmqcd4)
                                  +ppmqcd5)/5,
         ppmq_parental_solicitation = rowMeans(dplyr::select(., ppmqps1:ppmqps5)),
         ppmq_parental_control = rowMeans(dplyr::select(., ppmqpc1:ppmqpc5))) %>%
  select(-c(ppmq1:ppmqps5,parental_monitoring_questionnai2,
            parental_monitoring_questionnai3))

# chaos - dropping for purposes of this research
parent_df <- parent_df %>%
  select(-c(chaos_timestamp:chaos_complete))

# bpm adult
parent_df <- parent_df %>%
  mutate(bpm_att_a = rowSums(dplyr::select(., c(abpm1,abpm6,abpm7,abpm8,abpm9,
                                                abpm12))),
         bpm_ext_a = rowSums(dplyr::select(., c(abpm3,abpm13,abpm14,abpm17,
                                                abpm18))),
         bpm_int_a = rowSums(dplyr::select(., c(abpm2,abpm4,abpm5,abpm10,abpm15,
                                                abpm16)))) %>%
  select(-c(brief_problem_monitoradult_times:brief_problem_monitoradult_compl))

# parent emotional regulation
parent_df <- parent_df %>%
  mutate(erq_cog_a = rowMeans(dplyr::select(., c(perq1,perq3,perq5,perq7,
                                                 perq8,perq10))),
         erq_exp_a = rowMeans(dplyr::select(., c(perq2,perq4,perq6,
                                                 perq9)))) %>%
  select(-c(emotion_regulation_questionnair2:emotion_regulation_questionnair3))

# adult temperament - drop to simplify analysis
parent_df <- parent_df %>%
  select(-c(adult_temperament_questionnaire_:adult_temperament_questionnaire1))

# etq - drop to simplify analysis
parent_df <- parent_df %>%
  select(-c(eatq_p_on_c_timestamp:eatq_p_on_c_complete))

# stress - dropping for purposes of this research
```

```r
parent_df <- parent_df %>%
  select(-c(nih_toolbox_stress_timestamp:teen_birthday_complete))

# reported smoking during pregnancy and postpartum
parent_df <- parent_df %>%
  select(-c(BBID:ethn2, bl_6:bl_280, s2_10:s2_280, s3_6:s3_280,
            s4_6:s4_280,  s5_6:s5_280, s6_6:s6_280, s7_6:s7_280,
            chart21A:chart23) )   %>%
  rename(mom_smoke_16wk = bl_5,
         mom_smoke_22wk = s2_5,
         mom_smoke_32wk = s3_5,
         mom_smoke_pp1 = s4_5,
         mom_smoke_pp2 = s5_5,
         mom_smoke_pp12wk = s6_5,
         mom_smoke_pp6mo = s7_5,
         cotimean_34wk = wk34cot_cotimean,
         cotimean_pp6mo = mo6momcot_cotimean,
         cotimean_pp6mo_baby = mo6babcot_cotimean)


new_df <- inner_join(parent_df, child_df, by = "parent_id")
# write.csv(new_df, "project1.csv", row.names=FALSE)

##############################
### GENERATE NEW VARIABLES ###
##############################

# Generate new variables, modify and rename some variables
new_df2 = new_df %>%
         # Simplify mom smoke indicator
  mutate(mom_smoke_16wk = factor(mom_smoke_16wk, levels=c('2=No','1=Yes')
                                 , labels = c(0,1))
         , mom_smoke_22wk = factor(mom_smoke_22wk, levels=c('2=No','1=Yes')
                                   , labels = c(0,1))
         , mom_smoke_32wk = factor(mom_smoke_32wk, levels=c('2=No','1=Yes')
                                   , labels = c(0,1))
         , mom_smoke_pp1 = factor(mom_smoke_pp1, levels=c('2=No','1=Yes')
                                  , labels = c(0,1))
         , mom_smoke_pp2 = factor(mom_smoke_pp2, levels=c('2=No','1=Yes')
                                  , labels = c(0,1))
         , mom_smoke_pp12wk = factor(mom_smoke_pp12wk, levels=c('2=No','1=Yes')
```

```
                              , labels = c(0,1))
, mom_smoke_pp6mo = factor(mom_smoke_pp6mo, levels=c('2=No','1=Yes')
                             , labels = c(0,1))
# Make new indicator combining cig & ecig
, cig = pmax(cig_ever,e_cig_ever)
, drug = mj_ever
, alc = alc_ever
# Make tobacco consumption indicator
, pcig = ifelse(nidatob > 0, 1, 0)
# Make drug consumption indicator
, pdrug = ifelse(pmax(nidapres,nidaill) > 0, 1, 0)
# Make alcohol consumption indicator
, palc = ifelse(nidaalc > 0, 1, 0)
# Simplify child race
, race = case_when(tethnic == 1 ~ 'HispLatino'
                   , twhite == 1 ~ 'White'
                   , taian == 1 ~ 'Other'
                   , tasian == 1 ~ 'Other'
                   , tnhpi == 1 ~ 'Other'
                   , tblack == 1 ~ 'Black'
                   , trace_other == 1 ~ 'Other'
                   , TRUE ~ NA)
# Simplify parent race
, prace = case_when(pethnic == 1 ~ 'HispLatino'
                   , pwhite == 1 ~ 'White'
                   , paian == 1 ~ 'Other'
                   , pasian == 1 ~ 'Other'
                   , pnhpi == 1 ~ 'Other'
                   , pblack == 1 ~ 'Black'
                   , prace_other == 1 ~ 'Other'
                   , TRUE ~ NA)
# Rectify swan record for ids mentioned
, swan_inattentive = ifelse(parent_id %in% c(50502,51202,51602,52302
                                             ,53002,53502,53902,54402
                                             ,54602,54702)
                            ,NA
                            ,swan_inattentive)
, swan_hyperactive = ifelse(parent_id %in% c(50502,51202,51602,52302
                                             ,53002,53502,53902,54402
                                             ,54602,54702)
                            ,NA
```

```
                                    ,swan_hyperactive)
        # Rectify income record
        , income = case_when(income == '' ~ NA
                             , income == '250, 000' ~ 250000
                             , TRUE ~ as.numeric(income))
        # Make higher edu (education after high school) indicator
        , phigheredu = case_when(pedu %in% c(0,1,2) ~ 0
                             , pedu %in% c(3,4,5,6) ~ 1
                             , TRUE ~ NA)
        ) %>%
         # Transform mom smoke indicators into numeric
mutate(mom_smoke_16wk = as.numeric(as.character(mom_smoke_16wk))
        , mom_smoke_22wk = as.numeric(as.character(mom_smoke_22wk))
        , mom_smoke_32wk = as.numeric(as.character(mom_smoke_32wk))
        , mom_smoke_pp1 = as.numeric(as.character(mom_smoke_pp1))
        , mom_smoke_pp2 = as.numeric(as.character(mom_smoke_pp2))
        , mom_smoke_pp12wk = as.numeric(as.character(mom_smoke_pp12wk))
        , mom_smoke_pp6mo = as.numeric(as.character(mom_smoke_pp6mo))
        # Make indicator whether child use any substance at all (cig/drug/alc)
        , substance_at_all = case_when(cig == 1 | alc == 1 | drug == 1 ~ 1
                                       , cig == 0 & alc == 0 & drug == 0 ~ 0
                                       , TRUE ~ NA
                                       )
        # Make indicator whether parent use any substance at all (cig/drug/alc)
        , psubstance_at_all = case_when(pcig == 1 | palc == 1 | pdrug == 1 ~ 1
                                       , pcig == 0 & palc == 0 & pdrug == 0 ~ 0
                                       , TRUE ~ NA
                                       )
        # Track how many types of substance used by child (0-3)
        , num_substance_used = rowSums(dplyr::select(., cig:alc), na.rm=TRUE)
        # Track how many types of substance used by parent (0-3)
        , pnum_substance_used = rowSums(dplyr::select(., pcig:palc), na.rm=TRUE)
        # Total child's parental monitoring score
        , pmq_total = pmq_parental_knowledge + pmq_child_disclosure
                      + pmq_parental_solicitation + pmq_parental_control
        # Total parent's parental monitoring score
        , ppmq_total = ppmq_parental_knowledge + ppmq_child_disclosure
                      + ppmq_parental_solicitation + ppmq_parental_control
        # Average of brief problem monitor score (child)
        , bpm_summary = ifelse(is.na(bpm_int)
                               , (bpm_att+bpm_ext)/2
```

```
                            , (bpm_att+bpm_ext+bpm_int)/3)
        # Average of brief problem monitor score (parent)
        , pbpm_summary = ifelse(is.na(bpm_att_a)
                            , (bpm_int_a+bpm_ext_a)/2
                            , ifelse(is.na(bpm_ext_a)
                                    ,(bpm_int_a+bpm_att_a)/2
                                    ,(bpm_att_a+bpm_ext_a+bpm_int_a)/3))
        # Average of emotion regulation score (child)
        , erq_summary = ifelse(is.na(erq_cog)
                            , erq_exp
                            , ifelse(is.na(erq_exp)
                                    , erq_cog
                                    , (erq_exp+erq_cog)/2))
        # Average of emotion regulation score (child)
        , perq_summary = ifelse(is.na(erq_cog_a)
                            , erq_exp_a
                            , ifelse(is.na(erq_exp_a)
                                    , erq_cog_a
                                    , (erq_exp_a+erq_cog_a)/2))
        # Average of ADHD score (child)
        , swan_summary = (swan_hyperactive + swan_inattentive)/2
        # Age gap between parent and child
        , age_gap = abs(tage - page)
        # Simplify income based on this range (https://money.usnews.com/money/
        #    +personal-finance/family-finance/articles/
        #    +where-do-i-fall-in-the-american-economic-class-system)
        , income = case_when(income < 38133 ~ 1
                            , (income >= 38133 & income <57200) ~ 2
                            , (income >= 57200 & income <114000) ~ 3
                            , income >= 114000 ~ 4
                            , TRUE ~ NA)
        ) %>%
        # Indicator whether mom smoked at all during pregnancy or not
mutate(mom_prenatal_smoke = case_when((mom_smoke_16wk == 0 & mom_smoke_22wk == 0
                                    & mom_smoke_32wk == 0) ~ 0
                                    , (mom_smoke_16wk == 1 | mom_smoke_22wk == 1
                                    | mom_smoke_32wk == 1)  ~ 1
                                    , TRUE ~ NA)
        # Indicator whether mom smoked at all post pregnancy or not
        , mom_postnatal_smoke = case_when((mom_smoke_pp12wk == 0
                                        & mom_smoke_pp6mo == 0) ~ 0
```

```
                                    , (mom_smoke_pp12wk == 1
                                       | mom_smoke_pp6mo == 1) ~ 1
                                    , TRUE ~ NA)
        # Indicator whether mom consistently smoked during pregnancy or not
        #    number get bigger if mom consistently reported smoking every followup
        , mom_prenatal_smoke_consistency = mom_smoke_16wk + mom_smoke_22wk + mom_smoke_32
        # Indicator whether mom consistently smoked post pregnancy or not
        #    number get bigger if mom consistently reported smoking every followup
        , mom_postnatal_smoke_consistency = mom_smoke_pp12wk + mom_smoke_pp6mo
        # Indicator whether child exposed by smoked at all during pregnancy or not
        , prenatal_exposure = case_when(smoke_exposure_6mo == 0
                                      & smoke_exposure_12mo == 0 ~ 0
                                      , smoke_exposure_6mo == 1
                                      | smoke_exposure_12mo == 1 ~ 1
                                      , TRUE ~ NA
                                      )
        # Indicator whether child exposed by smoked at all post pregnancy or not
        , postnatal_exposure = case_when(smoke_exposure_2yr == 0
                                       & smoke_exposure_3yr == 0
                                       & smoke_exposure_4yr == 0
                                       & smoke_exposure_5yr == 0 ~ 0
                                       , smoke_exposure_2yr == 1
                                       | smoke_exposure_3yr == 1
                                       | smoke_exposure_4yr == 1
                                       | smoke_exposure_5yr == 1 ~ 1
                                       , TRUE ~ NA)
        # Indicator whether child consistently exposed by smoked during pregnancy or not
        #    number get bigger if child consistently exposed every followup
        , prenatal_exposure_consistency =  smoke_exposure_6mo + smoke_exposure_12mo
        # Indicator whether child consistently exposed by smoked post pregnancy or not
        #    number get bigger if child consistently exposed every followup
        , postnatal_exposure_consistency = smoke_exposure_2yr + smoke_exposure_3yr
                                         + smoke_exposure_4yr + smoke_exposure_5yr
        # Difference between child & parent parental monitoring score
        , pc_pmq_disharmony = abs(pmq_total - ppmq_total)
        ) %>%
        # impute some missing value in prenatal_exposure if mom_prenatal_smoke == 1
    mutate(prenatal_exposure = ifelse(is.na(prenatal_exposure) & mom_prenatal_smoke == 1
                                    ,1
                                    ,prenatal_exposure)
        # impute some missing value in postnatal_exposure if mom_postnatal_smoke == 1
```

```
        , postnatal_exposure = ifelse(is.na(postnatal_exposure)
                                      & mom_postnatal_smoke == 1
                                      ,1
                                      ,postnatal_exposure)
        ) %>%
        # Get prenatal & postnatal smoke exposure combination
        #   (e.g., 01 meaning no exposure during pregnancy but exposure after pregnancy)
  mutate(exposure_pattern = case_when(is.na(prenatal_exposure) ~ NA
                                      , is.na(postnatal_exposure) ~ NA
                                      , TRUE ~ paste(prenatal_exposure,postnatal_exp
                                      )
        # Get prenatal & postnatal mom smoke combination
        #   (e.g., 01 meaning mom didn't smoke during pregnancy but smoke after pregnancy
        , mom_smoke_pattern = case_when(is.na(mom_prenatal_smoke) ~ NA
                                      , is.na(mom_postnatal_smoke) ~ NA
                                      , TRUE ~ paste(mom_prenatal_smoke,mom_postnata
                                      )
  )

####################
### FILTER DATA ###
####################

# Only take necessary variables for analysis
new_df3 = new_df2 %>%
  dplyr::select(-c(plang:prace_other,childasd:cotimean_34wk,bpm_att_p:smoke_exposure_5yr
                   ,language:trace_other,num_cigs_30,num_e_cigs_30,num_mj_30,num_alc_30))
categorical_var = c('psex','employ','pedu','income','tsex','cig_ever'
                    ,'e_cig_ever','mj_ever','alc_ever','cig','drug','alc'
                    ,'pcig','pdrug','palc','race','prace','phigheredu'
                    ,'substance_at_all','psubstance_at_all','mom_prenatal_smoke'
                    ,'mom_postnatal_smoke','prenatal_exposure','postnatal_exposure'
                    ,'exposure_pattern','mom_smoke_pattern')

# Make tableone to get the population characteristic
pop_characteristic = CreateTableOne(data = new_df3, factorVars = categorical_var)
pop_characteristic_tb = print(pop_characteristic)
save(pop_characteristic_tb,file='pop_characteristic_tb.Rda')

# Statistic summary for continuous variables
summary(new_df3 %>% dplyr::select(-categorical_var))
```

```
####################################
### INTERRELATEDNESS BETWEEN EXT ###
####################################

# Make pair plot for all EXT variables
externalizing_behaviors = c('bpm_att','bpm_int','bpm_ext','erq_cog','erq_exp'
                            ,'swan_inattentive','swan_hyperactive'
                            ,'num_substance_used')

ext_behavior = new_df3[,externalizing_behaviors] %>%
  rename('Attention\nProblem' = 'bpm_att'
         ,'Internalizing\nProblem' = 'bpm_int'
         ,'Externalizing\nProblem' = 'bpm_ext'
         ,'Cognitive\nReappraisal' = 'erq_cog'
         ,'Expressive\nSuppression' = 'erq_exp'
         ,'ADHD\nInattentive' = 'swan_inattentive'
         ,'ADHD\nHyperactive' = 'swan_hyperactive'
         ,'Substance\nVariety' = 'num_substance_used')

# Make pair plot for all EXT variables
ggpairs(ext_behavior
        ,lower = list(continuous=wrap("smooth", colour="grey"))
        ,diag = list(continuous = wrap("densityDiag", colour = "blue"))
        ,upper = list(continuous = wrap("cor", size = 3))
        ,progress = NULL) +
  theme_bw() +
  labs(title = 'Pair Plot of EXT Variables') +
  theme(legend.position = 'bottom'
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5)
        , text=element_text(size=5))

#####################################################################
### COMPARE EXT CHARACTERISTICS BETWEEN DIFFERENT GROUPS (SDP) ###
#####################################################################

# Mom Prenatal Smoke
# Filter dataset
included = c('bpm_att','bpm_int','bpm_ext','erq_cog','erq_exp','swan_inattentive'
             ,'swan_hyperactive','num_substance_used','mom_prenatal_smoke')
prenatalComparison = new_df3[,included]
```

```r
# Stratify summary by group
prenatalComparison %>%
  tbl_summary(by = mom_prenatal_smoke
              , type = list(c('bpm_att','bpm_int','bpm_ext','erq_cog','erq_exp','swan_inat
                              ,'swan_hyperactive','num_substance_used') ~ 'continuous')
              , statistic = list(all_continuous() ~ "{median} ({p25}, {p75})")
              , missing = 'no'
              , label = list(
                'bpm_att' = 'Attention Problem'
                ,'bpm_int' = 'Internalizing Problem'
                ,'bpm_ext' = 'Externalizing Problem'
                ,'erq_cog' = 'Cognitive Reappraisal'
                ,'erq_exp' = 'Expressive Suppression'
                ,'swan_inattentive' = 'ADHD Inattentive'
                ,'swan_hyperactive' = 'ADHD Hyperactive'
                ,'num_substance_used' = 'Substance Variety'
              )) %>%
  add_p(pvalue_fun = ~ style_pvalue(.x, digits = 2)) %>%
  add_overall() %>%
  add_n() %>%
  modify_header(label ~ "**Variable**") %>%
  modify_spanning_header(c("stat_1", "stat_2") ~ "**Maternal Prenatal Smoking**") %>%
  bold_labels() %>%
  as_kable_extra(booktabs = TRUE) %>%
  kableExtra::kable_classic(full_width = F
                            , html_font = 'Cambria'
                            , latex_options = 'scale_down')

# Prenatal Exposure
# Filter dataset
included = c('bpm_att','bpm_int','bpm_ext','erq_cog','erq_exp','swan_inattentive'
             ,'swan_hyperactive','num_substance_used','prenatal_exposure')
prenatalComparison = new_df3[,included]

# Stratify summary by group
prenatalComparison %>%
  tbl_summary(by = prenatal_exposure
              , type = list(c('bpm_att','bpm_int','bpm_ext','erq_cog','erq_exp','swan_inat
                              ,'swan_hyperactive','num_substance_used') ~ 'continuous')
              , statistic = list(all_continuous() ~ "{median} ({p25}, {p75})")
              , missing = 'no'
```

```
              , label = list(
                'bpm_att' = 'Attention Problem'
                ,'bpm_int' = 'Internalizing Problem'
                ,'bpm_ext' = 'Externalizing Problem'
                ,'erq_cog' = 'Cognitive Reappraisal'
                ,'erq_exp' = 'Expressive Suppression'
                ,'swan_inattentive' = 'ADHD Inattentive'
                ,'swan_hyperactive' = 'ADHD Hyperactive'
                ,'num_substance_used' = 'Substance Variety'
              )) %>%
  add_p(pvalue_fun = ~ style_pvalue(.x, digits = 2)) %>%
  add_overall() %>%
  add_n() %>%
  modify_header(label ~ "**Variable**") %>%
  modify_spanning_header(c("stat_1", "stat_2") ~ "**Prenatal Environmental Smoking Exposur
  bold_labels() %>%
  as_kable_extra(booktabs = TRUE) %>%
  kableExtra::kable_classic(full_width = F
                            , html_font = 'Cambria'
                            , latex_options = 'scale_down')

# Postnatal Exposure
# Filter dataset
included = c('bpm_att','bpm_int','bpm_ext','erq_cog','erq_exp','swan_inattentive'
             ,'swan_hyperactive','num_substance_used','postnatal_exposure')
# prenatalComparison = new_df3[,c(9:10,26:30,46,56)]
prenatalComparison = new_df3[,included]

# Stratify summary by group
prenatalComparison %>%
  tbl_summary(by = postnatal_exposure
              , type = list(c('bpm_att','bpm_int','bpm_ext','erq_cog','erq_exp','swan_inat
                              ,'swan_hyperactive','num_substance_used') ~ 'continuous')
              , statistic = list(all_continuous() ~ "{median} ({p25}, {p75})")
              , missing = 'no'
              , label = list(
                'bpm_att' = 'Attention Problem'
                ,'bpm_int' = 'Internalizing Problem'
                ,'bpm_ext' = 'Externalizing Problem'
                ,'erq_cog' = 'Cognitive Reappraisal'
                ,'erq_exp' = 'Expressive Suppression'
```

```
                ,'swan_inattentive' = 'ADHD Inattentive'
                ,'swan_hyperactive' = 'ADHD Hyperactive'
                ,'num_substance_used' = 'Substance Variety'
              )) %>%
  add_p(pvalue_fun = ~ style_pvalue(.x, digits = 2)) %>%
  add_overall() %>%
  add_n() %>%
  modify_header(label ~ "**Variable**") %>%
  modify_spanning_header(c("stat_1", "stat_2") ~ "**Postnatal Environmental Smoking Exposu
  bold_labels() %>%
  as_kable_extra(booktabs = TRUE) %>%
  kableExtra::kable_classic(full_width = F
                            , html_font = 'Cambria'
                            , latex_options = 'scale_down')

### CHILD INTERNALIZING PROBLEM SCORE ###
bpm_int_p1 = ggplot(subset(new_df3, !is.na(prenatal_exposure))) +
  geom_boxplot(aes(x=as.factor(prenatal_exposure), y=bpm_int)
               , alpha = 0.5, fill = 'springgreen3', color = 'springgreen3') +
  theme_bw() +
  labs(x = 'Prenatal Smoke Exposure'
       ,y = 'Child Internalizing Problem Score'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5))
# bpm_int_p1

bpm_int_p2 = ggplot(subset(new_df3, !is.na(race))) +
  geom_boxplot(aes(x=as.factor(race), y=bpm_int)
               , alpha = 0.5, fill = 'springgreen3', color = 'springgreen3') +
  theme_bw() +
  labs(x = 'Race'
       ,y = 'Child Internalizing Problem Score'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5))
# bpm_int_p2
```

```r
bpm_int_p3 = ggplot(subset(new_df3, !is.na(employ))) +
  geom_boxplot(aes(x=as.factor(employ), y=bpm_int)
               , alpha = 0.5, fill = 'springgreen3', color = 'springgreen3') +
  scale_x_discrete(labels=c('No', 'Part-Time', 'Full-Time')) +
  theme_bw() +
  labs(x = 'Parent Employment Status'
       ,y = 'Child Internalizing Problem Score'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5))
# bpm_int_p3

bpm_int_p4 = ggplot(new_df3) +
  geom_jitter(aes(x=prenatal_exposure_consistency, y=bpm_int), color = 'grey') +
  geom_smooth(aes(x=prenatal_exposure_consistency, y=bpm_int), method = 'lm'
               , alpha = 0.5, fill = 'springgreen3', color = 'springgreen3') +
  theme_bw() +
  labs(x = 'Prenatal Smoke Exposure Consistency'
       ,y = 'Child Internalizing Problem Score'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5))
# bpm_int_p4

# Combine EXT variable plots
patchwork4 = (bpm_int_p1 + bpm_int_p2 + bpm_int_p3 + bpm_int_p4)

options(repr.plot.width=6, repr.plot.height=4)
patchwork4 +
  plot_annotation(tag_levels = 'A'
                  ,title = 'Child Internalizing Problem Score vs SDP & Other Potential Con
                  ,caption = 'Figure 2. Child Internalizing Problem Score vs SDP & Other P
                  ,theme = theme(plot.title = element_text(size = 12)
                                 ,plot.tag = element_text(size = 10)))
############################################################
### CORRELATION OF EXT WITH OTHER CONTINUOUS VARIABLE ###
############################################################
```

```r
# Variables to be compared
EXT = c('bpm_att','bpm_int','bpm_ext','erq_cog','erq_exp','swan_inattentive'
        ,'swan_hyperactive','num_substance_used')
non_EXT = c('bpm_att_a','bpm_int_a','bpm_ext_a','erq_cog_a','erq_exp_a'
            ,'pnum_substance_used','mom_prenatal_smoke_consistency'
            ,'mom_postnatal_smoke_consistency','prenatal_exposure_consistency'
            ,'postnatal_exposure_consistency','pmq_parental_knowledge'
            ,'pmq_child_disclosure','pmq_parental_solicitation'
            ,'pmq_parental_control','tage','age_gap')

# Make correlation matrix
cor_mat = cor(new_df3[,c(EXT,non_EXT)]
              , use="pairwise.complete.obs")

# Remove diagonal, redundant values and rename the values
cor_mat[!lower.tri(cor_mat)] = NA
cor_df = data.frame(cor_mat) %>%
  rownames_to_column() %>%
  gather(key="variable", value="correlation", -rowname) %>%
  filter(abs(correlation) > 0.35)
# Only take non EXT and EXT pair with abs(correlation) > 0.35
cor_df %>%
  rename('variable1' = 'rowname', 'variable2' = 'variable') %>%
  filter(!(variable1 %in% EXT & variable2 %in% EXT)
         , !(variable1 %in% non_EXT & variable2 %in% non_EXT)) %>%
  mutate(variable2 = case_when(variable2=='bpm_att' ~ 'Attention Problem'
                               ,variable2=='bpm_int' ~ 'Internalizing Problem'
                               ,variable2=='bpm_ext' ~ 'Externalizing Problem'
                               ,variable2=='erq_cog' ~ 'Cognitive Reappraisal'
                               ,variable2=='erq_exp' ~ 'Expressive Suppression'
                               ,variable2=='swan_inattentive' ~ 'ADHD Inattentive'
                               ,variable2=='swan_hyperactive' ~ 'ADHD Hyperactive'
                               ,variable2=='num_substance_used' ~ 'Substance Variety')
         ,variable1 = case_when(variable1=='bpm_att_a' ~ 'Parent Attention Problem'
                                ,variable1=='bpm_int_a' ~ 'Parent Internalizing Problem'
                                ,variable1=='bpm_ext_a' ~ 'Parent Externalizing Problem'
                                ,variable1=='erq_cog_a' ~ 'Parent Cognitive Reappraisal'
                                ,variable1=='mom_prenatal_smoke_consistency'
                                ~ 'Maternal Prenatal Smoking Consistency'
                                ,variable1=='mom_postnatal_smoke_consistency'
                                ~ 'Maternal Postnatal Smoking Consistency'
```

```
                                ,variable1=='prenatal_exposure_consistency'
                                ~ 'Prenatal Environmental Smoking Exposure Consistency'
                                ,variable1=='postnatal_exposure_consistency'
                                ~ 'Postnatal Environmental Smoking Exposure Consistency'
                                ,variable1=='pnum_substance_used' ~ 'Parental Substance Va
                                ,variable1=='pmq_parental_knowledge' ~ 'Parental Knowledge
                                ,variable1=='pmq_child_disclosure' ~ 'Child Disclosure'
                                ,variable1=='pmq_parental_solicitation' ~ 'Parental Solici
                                ,variable1=='pmq_parental_control' ~ 'Parental Control'
                                ,variable1=='cotimean_34wk' ~ 'Prenatal Maternal Cotinine'
                                ,variable1=='cotimean_pp6mo_baby' ~ 'Baby Cotinine 6mo'
                                ,variable1=='cotimean_pp6mo' ~ 'Postnatal Maternal Cotinin
                                ,variable1=='age_gap' ~ 'Parent Child Age Gap')) %>%
  arrange(desc(variable2)) %>%
  rename('Other Variables' = 'variable1'
         , 'Externalizing Behavior Variables' = 'variable2'
         , 'Correlation' = 'correlation') %>%
  mutate(Correlation = round(Correlation, 2)) %>%
  kableExtra::kbl(caption = 'Correlation Between EXT and Non EXT Variables'
                  , booktabs = T
                  , escape = F
                  , align = 'c') %>%
  kableExtra::kable_classic(full_width = F
                            ,font_size = 7
                            , html_font = 'Cambria'
                            , latex_options = 'HOLD_position')


### CHILD SUBSTANCE TYPES USED ###
subs_used_p1 = ggplot(subset(new_df3, !is.na(mom_postnatal_smoke_consistency))) +
  geom_jitter(aes(x=mom_postnatal_smoke_consistency, y=num_substance_used), color = 'grey'
  geom_smooth(aes(x=mom_postnatal_smoke_consistency, y=num_substance_used), method = 'lm'
              , alpha = 0.5, fill = 'mediumpurple4', color = 'mediumpurple4') +
  theme_bw() +
  labs(x = 'Mom Postnatal Smoke Consistency'
       ,y = 'Child Substance Types Used'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5))
```

```
# subs_used_p1

subs_used_p2 = ggplot(subset(new_df3, !is.na(pmq_parental_knowledge))) +
  geom_jitter(aes(x=pmq_parental_knowledge, y=num_substance_used), color = 'grey') +
  geom_smooth(aes(x=pmq_parental_knowledge, y=num_substance_used), method = 'lm'
              , alpha = 0.5, fill = 'mediumpurple4', color = 'mediumpurple4') +
  theme_bw() +
  labs(x = 'Parental Knowledge Score (Child POV)'
       ,y = 'Child Substance Types Used'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5))
# subs_used_p2

subs_used_p3 = ggplot(subset(new_df3, !is.na(pmq_child_disclosure))) +
  geom_jitter(aes(x=pmq_child_disclosure, y=num_substance_used), color = 'grey') +
  geom_smooth(aes(x=pmq_child_disclosure, y=num_substance_used), method = 'lm'
              , alpha = 0.5, fill = 'mediumpurple4', color = 'mediumpurple4') +
  theme_bw() +
  labs(x = 'Child Disclosure Score (Child POV)'
       ,y = 'Child Substance Types Used'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5))
# subs_used_p3

subs_used_p4 = ggplot(subset(new_df3, !is.na(pmq_parental_control))) +
  geom_jitter(aes(x=pmq_parental_control, y=num_substance_used), color = 'grey') +
  geom_smooth(aes(x=pmq_parental_control, y=num_substance_used), method = 'lm'
              , alpha = 0.5, fill = 'mediumpurple4', color = 'mediumpurple4') +
  theme_bw() +
  labs(x = 'Parental Control Score (Child POV)'
       ,y = 'Child Substance Types Used'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
```

```
        , plot.caption = element_text(hjust = 0.5))
# subs_used_p4

subs_used_p5 = ggplot(subset(new_df3, !is.na(bpm_att_a))) +
  geom_jitter(aes(x=bpm_att_a, y=num_substance_used), color = 'grey') +
  geom_smooth(aes(x=bpm_att_a, y=num_substance_used), method = 'lm'
              , alpha = 0.5, fill = 'mediumpurple4', color = 'mediumpurple4') +
  theme_bw() +
  labs(x = 'Parent Attention Problem Score'
       ,y = 'Child Substance Types Used'
       ,) +
  theme(legend.position = 'bottom'
        ,text = element_text(size=7)
        , plot.title = element_text(hjust = 0.5)
        , plot.caption = element_text(hjust = 0.5))
# subs_used_p5

# Combine EXT variable plots
patchwork7 = (subs_used_p1 + subs_used_p2 + subs_used_p3 + subs_used_p4
              +subs_used_p5)

options(repr.plot.width=6, repr.plot.height=4)
patchwork7 +
  plot_annotation(tag_levels = 'A'
                  ,title = 'Total Substance Types Used by Children vs SDP & Other Potentia
                  ,caption = 'Figure 3. Total Substance Types Used by Children vs SDP & Ot
                  ,theme = theme(plot.title = element_text(size = 12)
                                 ,plot.tag = element_text(size = 10)))
```