Effects of Behavioral Treatment on EOT Smoking Abstinence

Jiahao Cui

Abstract

This study investigates the effects of behavioral treatment on end-of-treatment (EOT) smoking abstinence among adults, with a focus on identifying baseline variables as predictors and potential moderators. Utilizing data from a randomized trial, we explore various variable selection methods—Lasso, Elastic Net, Random Forest, and SVM-RFE—across four distinct treatment configurations to identify key predictors within each context. For each group, variable selection procedures were applied using leave-one-out cross-validation (LOOCV) to ensure robust predictor identification. Following variable selection, logistic regression models incorporating interaction terms were employed to examine the moderating effects of baseline characteristics on treatment outcomes. By examining causal effects through odds ratios and interaction terms, we gain insights into the influences of baseline characteristics, such as FTCD score and Exclusive Mentholated Cigarette User on smoking cessation outcomes.

Introduction

The goal of this project is to use data from a randomized trial to examine baseline variables as potential moderators of the effects of behavioral treatment on end-of-treatment (EOT) abstinence. Additionally, we will evaluate baseline variables as predictors of abstinence, controlling for both behavioral treatment and pharmacotherapy.

This dataset is derived from a randomized, placebo-controlled trial aimed at evaluating the efficacy and safety of behavioral activation (BA) and varenicline (Var) for smoking cessation among individuals with a history of major depressive disorder (MDD). It includes data from 300 participants, with detailed demographic, clinical, and treatment-related variables. Key features include participant characteristics such as age, sex, race/ethnicity (e.g., Non-Hispanic White, Black, Hispanic), income levels, and clinical measures like nicotine metabolism ratio (NMR), craving scores, and hedonic capacity (measured using the Snaith-Hamilton Pleasure Scale). Treatment indicators specify whether participants were assigned to BA or varenicline groups, and outcome variables measure smoking abstinence at various time points. The dataset also includes information on antidepressant medication use, current major depressive episode status, and readiness to quit smoking. This rich dataset enables the analysis of interactions between behavioral and pharmacological interventions, identification of predictors of smoking cessation, and exploration of moderators influencing treatment efficacy, making it a valuable resource for understanding smoking cessation in a population with co-occurring mental health challenges.

In this project, we will begin by using several variable selection methods to identify the most important baseline variables for predicting abstinence, conditional on behavioral treatment and pharmacotherapy. Instead of using the entire dataset, we will split the data into four groups based on combinations of behavioral treatment and pharmacotherapy. This approach accounts for the complex interaction effects between baseline variables and treatment variables, particularly those caused by major depressive disorder (MDD)[1]. Although analyzing each group separately may reduce statistical power, it simplifies the model by removing the need to consider interaction terms with treatment, potentially increasing accuracy and interpretability.

We will then combine the selected variables across groups, using the union of each variable selection subset as the final set of predictors. Next, we will assess baseline variables as potential moderators of the effects of behavioral treatment on EOT abstinence. In this stage, we treat pharmacotherapy as a treatment variable rather than a baseline variable, as it is assigned post-randomization. Therefore, we will split the data into two groups based on the pharmacotherapy variable and calculate the potential moderating effects within each group.

Data Collection

We make a summary table of the data below Table 1 which provides a comprehensive summary of the some baseline characteristics for participants, grouped by 4 different treatment types. Variables such as age, sex, income, and education provide demographic context, while smoking-related metrics like the FTCD score, cigarettes per day, and exclusive menthol use describe participants' smoking habits and dependencies. Additionally, mental health indicators, including the BDI score for depressive symptoms and Anhedonia, help identify baseline psychological factors that may influence treatment outcomes.

From the summary table, we observe that the distribution of variables differs across treatment groups. For instance, the mean age in the ST + placebo group is 50 years (with a standard deviation of 11), while the BASC + placebo group has a slightly lower mean age of 51 years (with a standard deviation of 14), but not significant through ANOVA. These differences highlight potential variations in baseline characteristics among groups.

To further explore these differences, we will plot histograms of continuous variables to visually assess their distributions across groups. These plots, displayed in the appendix, provide

Table 1: Summary table

Variable	**BASC + placebo** N = 68	**BASC + varenicline** N = 83	**ST + placebo** N = 68	**ST + varenicline** N = 81
Sex				
Male	30 / 68 (44%)	39 / 83 (47%)	29 / 68 (43%)	37 / 81 (46%)
Female	38 / 68 (56%)	44 / 83 (53%)	39 / 68 (57%)	44 / 81 (54%)
Age	51 (14)	50 (13)	50 (11)	49 (13)
Income				
Less than $20,000	25 / 67 (37%)	30 / 82 (37%)	26 / 68 (38%)	29 / 80 (36%)
$20,000–$35,000	16 / 67 (24%)	17 / 82 (21%)	14 / 68 (21%)	21 / 80 (26%)
$35,000–$50,000	8 / 67 (12%)	13 / 82 (16%)	14 / 68 (21%)	11 / 80 (14%)
$50,000–$75,000	12 / 67 (18%)	12 / 82 (15%)	8 / 68 (12%)	6 / 80 (7.5%)
More than $75,000	6 / 67 (9.0%)	10 / 82 (12%)	6 / 68 (8.8%)	13 / 80 (16%)
Unknown	1	1	0	1
Education				
Grade school	1 / 68 (1.5%)	0 / 83 (0%)	0 / 68 (0%)	0 / 81 (0%)
Some high school	3 / 68 (4.4%)	7 / 83 (8.4%)	2 / 68 (2.9%)	4 / 81 (4.9%)
High school graduate or GED	23 / 68 (34%)	15 / 83 (18%)	11 / 68 (16%)	27 / 81 (33%)
Some college/technical school	22 / 68 (32%)	32 / 83 (39%)	38 / 68 (56%)	24 / 81 (30%)
College graduate	19 / 68 (28%)	29 / 83 (35%)	17 / 68 (25%)	26 / 81 (32%)
Cigarettes per day at baseline phone survey	16 (9)	16 (9)	15 (7)	14 (7)
BDI score at baseline	19 (12)	18 (11)	18 (11)	20 (12)
Anhedonia	2 (3)	2 (3)	3 (3)	2 (3)
Unknown	2	0	1	0
Exclusive Mentholated Cigarette User	40 / 68 (59%)	48 / 82 (59%)	43 / 67 (64%)	47 / 81 (58%)
Unknown	0	1	1	0
FTCD Score	5 (2)	5 (2)	5 (2)	5 (2)
Unknown	0	0	1	0

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

additional insights into the spread and shape of variables like age, FTCD score, and cigarettes per day, which may be important for understanding treatment effects.

The dataset contains a mix of categorical and numeric variables. To create a correlation matrix heatmap Figure 1 for this type of data, we will use a combination of Pearson correlation for numeric variables and Cramér's V for categorical variables. For mixed variable pairs, we'll use point-biserial correlation.

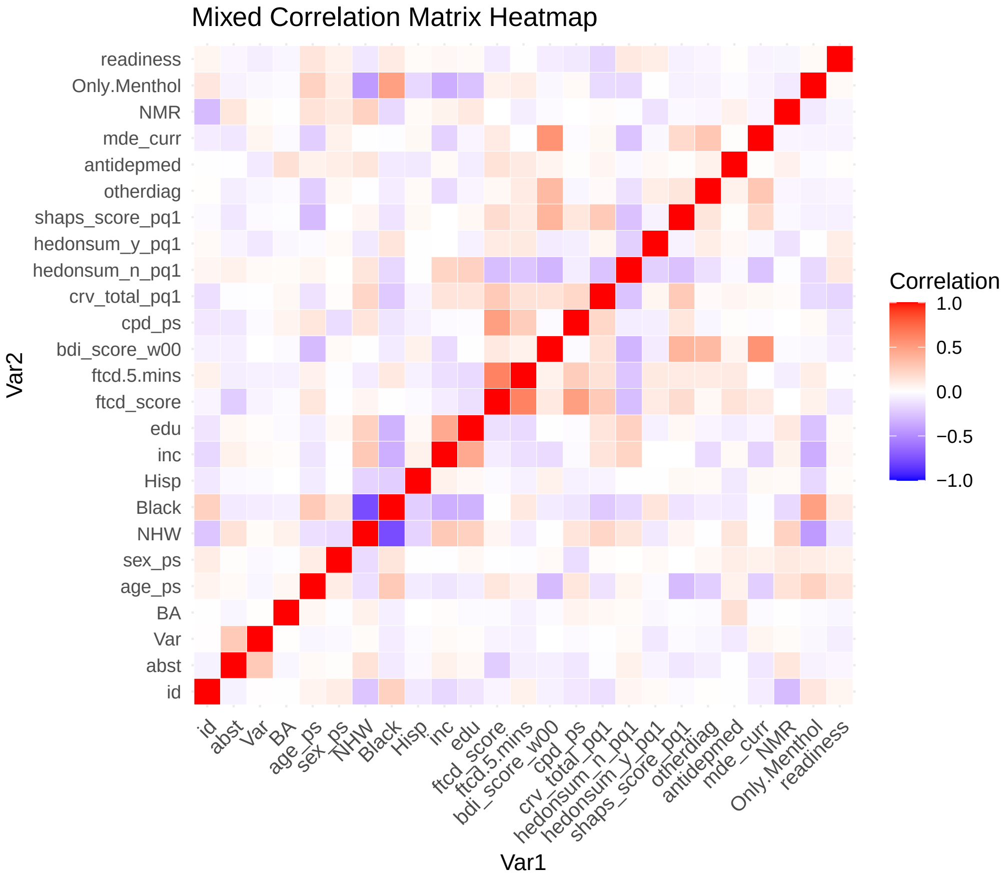

Figure 1

The missingness pattern for our data is illustrated in the appendix (see Figure 6). This visualization provides insights into the extent and distribution of missing values across variables. To maintain unbiasedness and keep the sample size, we will employ multiple imputation as our strategy for handling missing data. Multiple imputation allows us to create a series of complete datasets, each with plausible values filled in for the missing data, thereby reducing bias and preserving variability in the dataset for robust analysis. When using mice() function to do the multiple imputation, we have to run the model on each imputed dataset and then pool the results to obtain valid estimates. But also it is very computational expensive, so we will not show the code here.

Evaluate baseline variables as predictors of abstinence, controlling for behavioral treatment and pharmacotherapy.

We recognize that interaction effects involving the treatment variable are likely to influence outcomes, particularly given the impact of major depressive disorder (MDD) and other relevant domain knowledge.

In this project, we will address these interactions by splitting the data into four groups based on the treatment variable. By analyzing each group separately, we can perform variable selection without needing to account for complex interaction terms involving treatment. This approach enhances both the accuracy and interpretability of our results, allowing us to isolate the effect of baseline variables within each treatment context while maintaining a focus on clinically meaningful associations.

We assume that the treatment may have interaction effects with other variables for the outcomes. But covariates without treatment doesn't have interaction effects for the outcome. Otherwise, the Lasso Regression is used for only main effects but, finally, fitting a model with interaction doesn't make a lot of sense.

There will be four groups of data shown in the Table 1. We will first consider the group_st_placebo which is behavioral activation therapy and placebo treatment. We will consider four variable selection methods: Lasso[2], Elastic Net[3], Random Forest[4], and SVM-RFE[5] with a leave one out cross-validation (LOOCV) strategy. Finally, we will compare the performance of each method of variable selection on the validation set.

The data has been divided into four groups, as presented in Table 1. Our initial focus will be on the group_st_placebo group, which consists of participants receiving behavioral activation therapy with a placebo treatment. For this group, we will apply four different variable selection methods: Lasso, Elastic Net, Random Forest, and SVM-RFE. Each method will be implemented using a leave-one-out cross-validation (LOOCV) strategy(since the sample size is too small after splitting the data) to ensure robust performance assessment. Ultimately, we will compare the effectiveness of these variable selection methods based on their performance on the validation set, allowing us to identify the most suitable approach for selecting relevant predictors in this treatment context.

Lasso is the most widely used model, and has the ability of spar-silty than L2 regression, which is better to do the variable selection. Elastic Net is a combination of L1 and L2 regularization, which can handle the correlated variables. Random Forest is a non-parametric model, which can capture the complex relationship between variables. SVM-RFE is a feature selection method based on SVM, which can handle the high-dimensional data effectively.

Table 2: Selected Variables with Non-Zero Coefficients for Lasso

Variable	Coefficient
(Intercept)	0.1076792
NHW	0.0805860
Black	-1.3507672
edu	0.1827921
ftcd_score	-0.4164033
hedonsum_n_pq1	0.0053137
mde_curr	-1.1308452

To begin our analysis, we will apply Lasso regression for logistic regression to the dataset, focusing on selecting relevant predictors without incorporating interaction terms. The Lasso approach, which uses L1 regularization, is particularly useful here as it shrinks some coefficients to zero, effectively excluding less important variables from the model.

We will evaluate the performance of this Lasso logistic regression model using the ROC metric. The ROC curve provides a visual assessment of the model's ability to distinguish between the outcome classes across various threshold values, while the AUC (Area Under the Curve) value quantifies this discriminative power. This metric will allow us to gauge the effectiveness of the Lasso-selected variables in predicting the outcome accurately.

The Lasso logistic regression model achieved an overall mean accuracy of 0.897 on the validation set, indicating strong predictive performance. The optimal tuning parameter, lambda = 0.0327, was identified as the best choice for minimizing prediction error while effectively regularizing the model.

After applying Lasso with this lambda value, a subset of variables with non-zero coefficients was selected, as displayed in Table 2. These selected variables are the most influential predictors in the model, providing insight into key factors associated with the outcome.

To further evaluate predictive performance, we will apply an Elastic Net model to the same dataset using leave-one-out cross-validation (LOOCV). Elastic Net combines both L1 (Lasso) and L2 (Ridge) regularization, which allows it to handle correlated predictors more effectively than Lasso alone.

In this Elastic Net model, we will tune the alpha parameter to find the optimal balance between L1 and L2 penalties, along with the lambda parameter for regularization strength. The LOOCV approach will again ensure a robust assessment of model performance, minimizing the risk of overfitting and allowing for a direct comparison with the Lasso model's performance.

After identifying the best parameters, we will evaluate the Elastic Net model's accuracy and examine the selected variables with non-zero coefficients to understand the most influential predictors in the context of Elastic Net regularization.

The Elastic Net model, applied with leave-one-out cross-validation (LOOCV), yielded an overall mean accuracy of 0.897 on the validation set, identical to the Lasso model's performance. The best tuning parameters for Elastic Net were found to be lambda = 0.0327 and alpha = 1. An alpha value of 1 effectively makes Elastic Net identical to Lasso, as it relies entirely on the L1 penalty with no contribution from the L2 component. This outcome suggests that a pure L1 regularization approach is optimal for this dataset. The selected variables with non-zero coefficients thus mirror those identified in the Lasso model, highlighting consistent predictors that are influential for the outcome.

Next, we apply non-parametric methods for variable selection, beginning with a Random Forest model. This approach does not assume a specific underlying distribution for the data, allowing it to capture complex relationships and interactions among variables.

The Random Forest model achieved an overall mean accuracy of 0.926 on the validation set, indicating an improvement over the Lasso and Elastic Net models. This higher accuracy reflects Random Forest's ability to leverage ensemble learning to boost predictive performance.

The importance of each variable in the Random Forest model is presented in Table 3. The Variable Importance metric highlights the most influential predictors identified by the model, providing the variables that contribute most to the outcome.

Finally, we applied Support Vector Machine (SVM) for variable selection, achieving an overall mean accuracy of 0.9118 on the validation set. This result highlights SVM's capability to handle high-dimensional data effectively, providing competitive accuracy in comparison to both parametric and non-parametric methods. The variables selected by SVM with non-zero coefficients are "edu", "ftcd_score", "cpd_ps", "Black", and "mde_curr".

We will now replicate the same variable selection procedures—using Lasso, Elastic Net, Random Forest, and SVM—on the remaining three treatment groups. For each group, we will assess the model's predictive performance by calculating the overall mean accuracy on the validation set. The results of these analyses are summarized in Table 4.

Table 4: Model Performance Results

	Lasso	Elastic Net	Random Forest	SVM-RFE
ST + Placebo	0.897	0.897	0.926	0.912
BASC + Placebo	0.926	0.926	0.897	0.912
ST + Varenicline	0.679	0.679	0.642	0.741
BASC + Varenicline	0.711	0.723	0.747	0.687

Based on the findings in Table 4, we selected the "best" variable selection method for each treatment group. The "best" method is defined as the one that achieved the highest mean accuracy on the validation set within each group. Following this approach, we identified the most influential variables in each group as follows:

Table 3: Variable Importance in Random Forest Model

	Variable	Importance
ftcd_score	100.000000	100.000000
cpd_ps	56.514403	56.514403
Black	31.739441	31.739441
NHW	25.516361	25.516361
mde_curr	25.148696	25.148696
hedonsum_n_pq1	22.469457	22.469457
Only.Menthol	22.434986	22.434986
inc	21.699249	21.699249
sex_ps	16.579944	16.579944
Hisp	16.540525	16.540525
NMR	14.087911	14.087911
crv_total_pq1	13.390671	13.390671
antidepmed	12.624490	12.624490
bdi_score_w00	10.888064	10.888064
edu	8.657955	8.657955
hedonsum_y_pq1	8.089363	8.089363
ftcd.5.mins	8.041056	8.041056
readiness	7.719720	7.719720
age_ps	6.191098	6.191098
otherdiag	3.082171	3.082171
shaps_score_pq1	0.000000	0.000000

BASC + Placebo: The selected variables are ftcd_score, cpd_ps, Black, and mde_curr.

ST + Placebo: The selected variables include NHW, Black, edu, ftcd_score, hedonsum_n_pq1, and mde_curr.

ST + Varenicline: The selected variables are NMR, readiness, otherdiag, hedonsum_n_pq1, and Only.Menthol.

BASC + Varenicline: The selected variables are ftcd_score, hedonsum_n_pq1, readiness, and Only.Menthol.

These selected variables represent the most predictive baseline characteristics for each treatment group. By focusing on these key variables, we aim to gain insights into the factors that are most influential within each treatment context.

Examine baseline variables as potential moderators of the effects of behavioral treatment on end-of-treatment (EOT) abstinence.

To further explore the impact of baseline variables, we will assess them as potential moderators of the effects of behavioral treatment on end-of-treatment (EOT) abstinence. Moderators help us understand how the relationship between treatment and outcome may vary depending on certain baseline characteristics.

For this analysis, we will split the data into two groups based on the pharmacotherapy variable (i.e., whether participants received varenicline or placebo). This separation allows us to examine potential moderating effects within each pharmacotherapy condition, ensuring that any observed effects are not confounded by differences in pharmacotherapy status. Within each pharmacotherapy group, we will calculate the moderation effects of baseline variables on EOT abstinence.

For the group receiving varenicline, without considering baseline variables as potential moderators, the causal effects, defined as odds ratios, are as follows : "Odds Ratio (OR): 0.9652" and "Log Odds Ratio (OR): -0.0351". These values provide a baseline estimate of the treatment effect without accounting for potential moderators.

Following the variable selection procedure, we considered baseline variables as potential moderators of the treatment effects. For the varenicline configuration, we selected the union of key variables: NMR, readiness, otherdiag, hedonsum_n_pq1, Only.Menthol, and ftcd_score. We employed a logistic regression model to assess these potential moderators by including interaction terms between behavioral treatment and each selected variable. To capture higher-order interactions without overcomplicating the model, we included three-way interaction terms involving behavioral treatment and the intersection of two selected variable subset: hedonsum_n_pq1, readiness, Only.Menthol. Then, we do a stepwise selection based on AIC to select the best model.

Table 5: Logistic model for varenicline group

Variable	**OR**[1]	**95% CI**[1]	**p-value**
NMR	8.54	1.74, 46.4	0.010
BA	11.2	1.59, 90.0	0.018
readiness	0.76	0.55, 1.03	0.080
otherdiag	0.52	0.23, 1.10	0.093
Only.Menthol	2.30	0.83, 6.85	0.12
ftcd_score	1.03	0.81, 1.31	0.8
BA * Only.Menthol	0.23	0.05, 0.98	0.049
BA * ftcd_score	0.70	0.50, 0.98	0.042

[1] OR = Odds Ratio, CI = Confidence Interval

From the Table 5, the final selected logistic model are as follows:

$$logit(E[abst = 1|var = 1, X]) = -0.05 + 2.14NMR + 2.42BA - 0.28readiness-$$

$$0.66otherdiag + 0.83Only.Menthol + 0.03FTCD - 1.48BA \times Only.Menthol - 0.35BA \times FTCD$$

The coefficient of the treatment variable (BA) is estimated to be 2.42, which differs from the odds ratio calculated previously. This discrepancy highlights that the logistic regression model, accounting for interaction effects, may yield different estimates from a simple odds ratio calculation. Our primary interest lies in identifying moderators of the causal effect, rather than interpreting each individual coefficient. Focusing on the interaction terms, the coefficient for the interaction between BA and Only.Menthol is -1.48, suggesting that the treatment effect (in terms of odds ratio) is 1.48 lower for participants who exclusively use mentholated cigarettes. Similarly, the interaction between BA and ftcd_score has a coefficient of -0.35, indicating that the treatment effect decreases by 0.35 for each unit increase in FTCD score at baseline.

For the placebo configuration, the causal effects calculated without accounting for baseline variables are as follows. “Odds Ratio (OR): 0.4797” and “Log Odds Ratio (OR): -0.7347”.

```
[1] "Odds Ratio (OR): 0.479651503690653"

[1] "Log Odds Ratio (OR): -0.734695472748316"
```

Next, we considered both the union and intersection sets of selected variables for the placebo configuration. The union set includes ftcd_score, cpd_ps, Black, mde_curr, NHW, edu, and hedonsum_n_pq1, while the intersection set comprises ftcd_score, Black, and mde_curr. We then fit a logistic regression model, applying stepwise selection based on AIC criteria to identify the optimal model as before.

Table 6: Logistic model for placebo group

Variable	**OR**[1]	**95% CI**[1]	**p-value**
ftcd_score	0.82	0.69, 0.97	0.024
BA	0.34	0.11, 1.02	0.058
Black	2.41	0.76, 9.42	0.2
NHW	3.88	1.23, 15.2	0.031
hedonsum_n_pq1	0.97	0.94, 1.00	0.077
BA * hedonsum_n_pq1	1.04	1.00, 1.09	0.033

[1] OR = Odds Ratio, CI = Confidence Interval

From the Table 6, the final selected logistic model are as follows:

$$logit(E[abst = 1|var = 0, X]) = -0.07 - 0.19FTCD - 1.09BA + 0.88Black+$$
$$1.36otherdiag - 0.03PleasurableEventsScale + 0.04BA \times PleasurableEventsScale$$

The coefficient of the treatment variable (BA) is estimated to be -1.09, differing from the previously calculated odds ratio. This variation reflects how accounting for additional variables and interaction terms in the logistic regression model can influence the treatment effect estimate. Our primary focus is on the interaction term, specifically the interaction between BA and the Pleasurable Events Scale. This interaction coefficient is 0.04, indicating that the treatment effect (in terms of odds ratio) increases by 0.04 for each unit increase in the Pleasurable Events Scale score.

Conclusion:

This study underscores the complexity of treatment effects in smoking cessation interventions and the importance of considering baseline variables as moderators. By analyzing four distinct treatment configurations and applying multiple variable selection methods, we identified key predictors of EOT abstinence for each group. The use of interaction terms in logistic regression models further revealed significant moderating effects, with variables like FTCD score and Pleasurable Events Scale score. These findings highlight the need for personalized treatment approaches that account for individual baseline characteristics, particularly in populations with co-occurring conditions such as major depressive disorder.

References

1. Hitsman B, Papandonatos G D, Gollan J K, et al. Efficacy and safety of combination behavioral activation for smoking cessation and varenicline for treating tobacco dependence among individuals with current or past major depressive disorder: A 2× 2 factorial, randomized, placebo-controlled trial[J]. Addiction, 2023, 118(9): 1710-1725.
2. Tibshirani R. Regression shrinkage and selection via the lasso[J]. Journal of the Royal Statistical Society Series B: Statistical Methodology, 1996, 58(1): 267-288.
3. Zou H, Hastie T. Regularization and variable selection via the elastic net[J]. Journal of the Royal Statistical Society Series B: Statistical Methodology, 2005, 67(2): 301-320.
4. Breiman L. Random forests[J]. Machine learning, 2001, 45: 5-32.
5. Cortes C. Support-Vector Networks[J]. Machine Learning, 1995.

Appendix

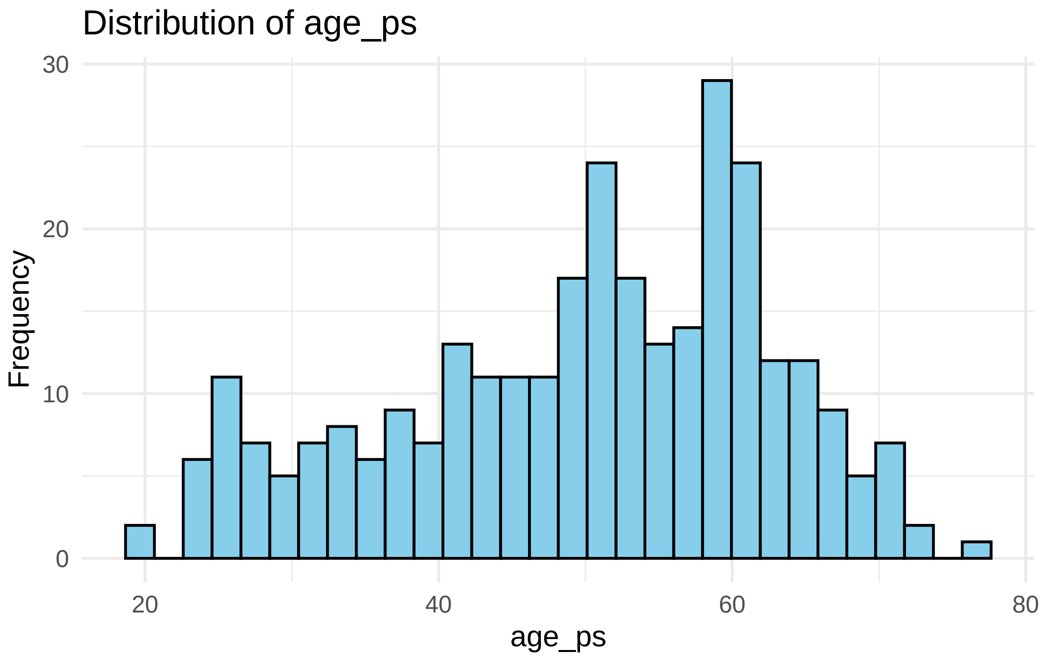

Figure 2

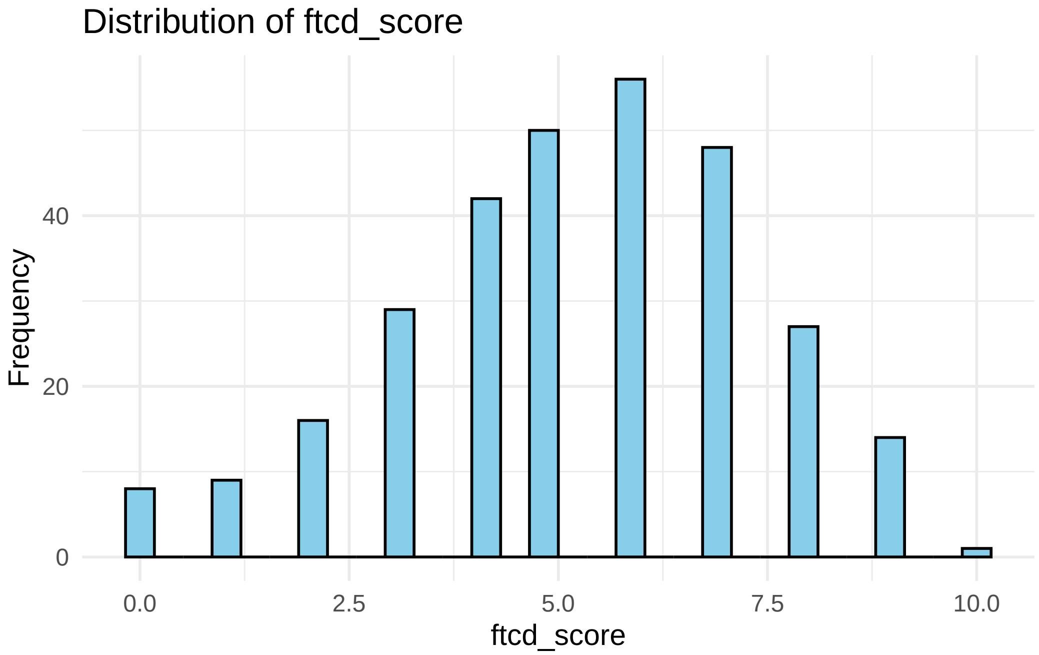

Figure 3

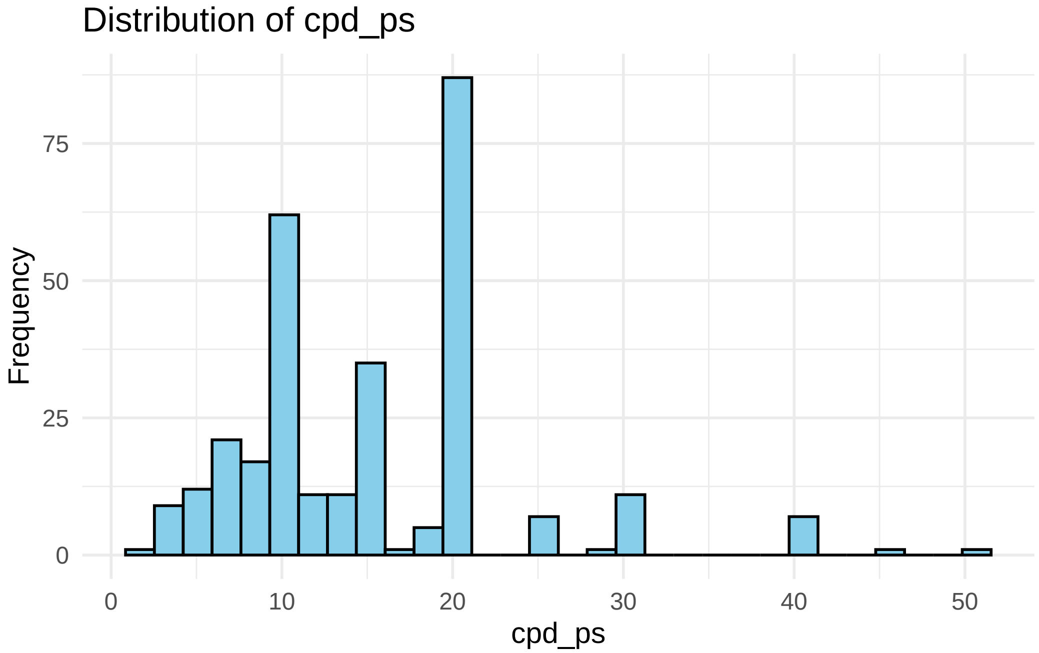

Figure 4

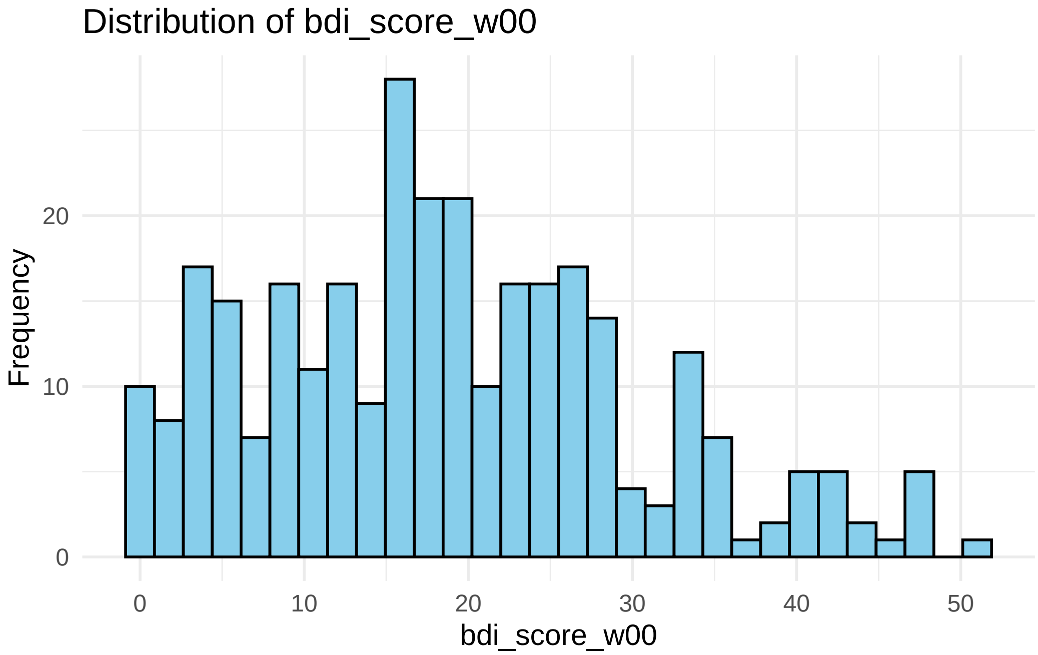

Figure 5

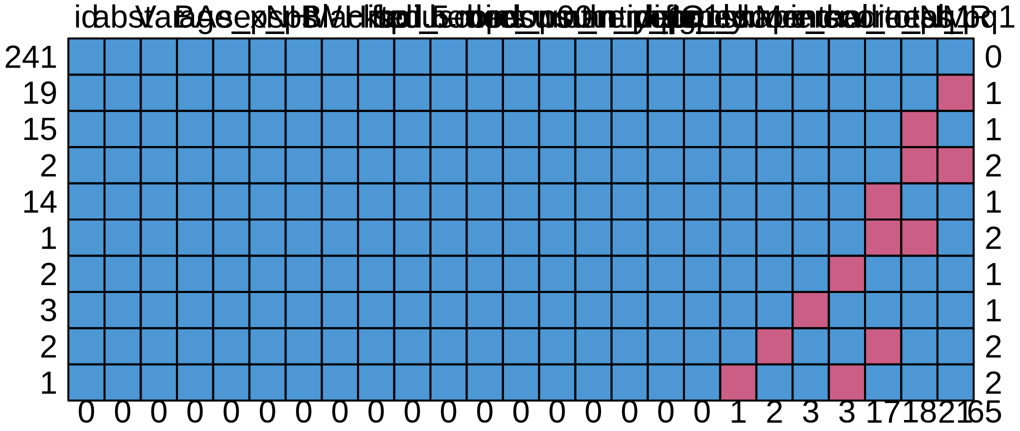

Figure 6

Code Appendix:

```
library(dplyr)
library(janitor)
library(ggplot2)
library(tidyr)
library(reshape2)
library(mice)
library(VIM)
library(MatchIt)
library(survey)
library(glmnet)
library(caret)
library(gtsummary)
library(kableExtra)
library(e1071)
library(MASS)
library(DescTools)
library(ltm)
library(epitools)

project2 <- read.csv("project2.csv")
# Summarize the data by treatment group

# Step 3: Create the treatment group variable based on Var and BA combinations
data_tbl <- project2 %>%
  mutate(
    "Treatment Group" = case_when(
      BA == 0 & Var == 0 ~ "ST + placebo",
      BA == 1 & Var == 0 ~ "BASC + placebo",
      BA == 0 & Var == 1 ~ "ST + varenicline",
      BA == 1 & Var == 1 ~ "BASC + varenicline"
    ),
    Sex = factor(sex_ps, labels = c("Male", "Female")),
    Age = age_ps,
    Income = factor(inc, labels = c("Less than $20,000", "$20,000-$35,000", "$35,000-$50,000
    Education = factor(edu, labels = c("Grade school", "Some high school", "High school gradu
    "Cigarettes per day at baseline phone survey" = cpd_ps,
    "BDI score at baseline" = bdi_score_w00,
    Anhedonia = shaps_score_pq1,
    "Exclusive Mentholated Cigarette User" = factor(Only.Menthol, labels = c("No", "Yes")),
    "FTCD Score" = ftcd_score
  )

# Use gtsummary to create the table
table_summary <- data_tbl[26:35] %>%
  tbl_summary(by = "Treatment Group",
              statistic = list(all_continuous() ~ "{mean} ({sd})",
                               all_categorical() ~ "{n} / {N} ({p}%)")) %>% # missing = "no"
  modify_header(label = "**Variable**") %>%
  modify_spanning_header(treatment_group = "**Treatment Group**") %>%
  as_kable_extra(booktabs = TRUE) %>%
  kable_styling(latex_options = "scale_down")

# Display the table
table_summary
# Load the dataset
data <- project2

# Separate numeric and categorical variables
numeric_vars <- sapply(data, is.numeric)
categorical_vars <- !numeric_vars

# Function to calculate mixed correlation matrix
calculate_mixed_correlation <- function(df, numeric_vars, categorical_vars) {
  n <- ncol(df)
  corr_matrix <- matrix(NA, n, n)
  colnames(corr_matrix) <- names(df)
  rownames(corr_matrix) <- names(df)

  for (i in 1:n) {
    for (j in 1:n) {
      if (i == j) {
        corr_matrix[i, j] <- 1
      } else if (numeric_vars[i] && numeric_vars[j]) {
        # Pearson correlation
        corr_matrix[i, j] <- cor(df[[i]], df[[j]], use = "complete.obs", method = "pearson")
      } else if (categorical_vars[i] && categorical_vars[j]) {
        # Cramér's V for categorical variables
        corr_matrix[i, j] <- CramerV(as.factor(df[[i]]), as.factor(df[[j]]), bias.correct = T
      } else {
        # Point-biserial correlation for mixed types
        num_col <- ifelse(numeric_vars[i], df[[i]], df[[j]])
        cat_col <- as.factor(ifelse(numeric_vars[i], df[[j]], df[[i]]))
        if (length(unique(cat_col)) == 2) {
          corr_matrix[i, j] <- biserial.cor(num_col, cat_col, use = "complete.obs")
        } else {
          corr_matrix[i, j] <- NA  # Not defined for non-binary categorical
        }
      }
    }
  }
  return(corr_matrix)
}

# Calculate the mixed correlation matrix
mixed_corr_matrix <- calculate_mixed_correlation(data, numeric_vars, categorical_vars)

# Convert matrix to long format for ggplot
melted_corr_matrix <- melt(mixed_corr_matrix, na.rm = TRUE)

# Plotting the heatmap
ggplot(data = melted_corr_matrix, aes(Var1, Var2, fill = value)) +
  geom_tile(color = "white") +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0, limit = c(-1
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1, size = 10, hjust = 1)) +
  coord_fixed() +
  labs(title = "Mixed Correlation Matrix Heatmap")


# Step 2: Set up and perform multiple imputation
# Specify the number of imputations (e.g., m = 5) and imputation method
project2_impute <- mice(project2, m = 5, method = "pmm", maxit = 50, seed = 500, printFlag=F
project2 <- complete(project2_impute)
# Step 1: Create a new variable that indicates each combination of BA and Var
project2 <- project2 %>%
  mutate(treatment_group = case_when(
    BA == 0 & Var == 0 ~ "ST + Placebo",
    BA == 1 & Var == 0 ~ "BASC + Placebo",
    BA == 0 & Var == 1 ~ "ST + Varenicline",
    BA == 1 & Var == 1 ~ "BASC + Varenicline"
  ))

# Step 2: Split the data into four groups based on the treatment_group variable
```

```r
group_st_placebo <- project2 %>% filter(treatment_group == "ST + Placebo")
group_basc_placebo <- project2 %>% filter(treatment_group == "BASC + Placebo")
group_st_varenicline <- project2 %>% filter(treatment_group == "ST + Varenicline")
group_basc_varenicline <- project2 %>% filter(treatment_group == "BASC + Varenicline")
# Define outcome and predictor variables
y <- group_st_placebo$abst  # Outcome variable
X <- group_st_placebo[,5:25] # Predictor variables

# Convert `y` to a factor if it isn't already
y <- as.factor(y)
# Sanitize factor levels to be valid R variable names
levels(y) <- make.names(levels(y))

# Convert predictors to model matrix format for glmnet compatibility
X <- model.matrix(~ . - 1, data = X)  # "-1" removes the intercept for compatibility with gl
# Step 1: Set up LOOCV in caret
train_control <- trainControl(
  method = "LOOCV",
  # method = "cv", number = 10,  # 10-fold cross-validation
  savePredictions = "final",   # Save predictions for each fold
  classProbs = TRUE,           # For binary classification, to get probabilities
  summaryFunction = twoClassSummary  # Use AUC, sensitivity, and specificity
)

# Step 2: Define the tuning grid for alpha and lambda
tune_grid <- expand.grid(
  alpha = 1,              # Lasso
  lambda = 10^seq(-3, 3, length = 100)  # Range for lambda tuning
)

# Step 3: Train the model using caret with LOOCV and glmnet
Lasso_model <- train(
  x = X,
  y = y,
  method = "glmnet",
  trControl = train_control,
  tuneGrid = tune_grid,
  family = "binomial",
  metric = "ROC"
)
# Step 4: Get validation performance metrics for each fold
predictions <- Lasso_model$pred
```

```
# View the validation results for each LOOCV fold
# print("Performance metrics for each LOOCV fold:")
# print(predictions)

# Step 5: Calculate overall validation performance
# For example, calculating mean accuracy if available
mean_accuracy_Lasso <- mean(predictions$pred == predictions$obs)
# print(paste("Overall mean accuracy on validation set:", mean_accuracy))

# Step 4: Output the selected model results
# print("Best tuning parameters:")
# print(Lasso_model$bestTune)

# Display coefficients of the final model with selected variables
final_coefficients <- coef(Lasso_model$finalModel, s = Lasso_model$bestTune$lambda)

selected_coefficients <- final_coefficients[final_coefficients != 0]

# Convert coefficients to a data frame for table display
coef_df <- data.frame(
  Variable = rownames(final_coefficients)[as.vector(final_coefficients != 0)],
  Coefficient = as.vector(selected_coefficients)
)


# Display a nicely formatted table
kable(coef_df,
      # caption = "Selected Variables with Non-Zero Coefficients",
      col.names = c("Variable", "Coefficient"),
      format = "latex",       # Set to "latex" for PDF compatibility
      booktabs = TRUE) %>%
  kable_styling(full_width = FALSE, bootstrap_options = c("striped", "hover"))
# Define grid for tuning both alpha and lambda
tune_grid <- expand.grid(alpha = seq(0, 1, by = 0.1), lambda = 10^seq(-3, 3, length = 100))

# Train Elastic Net model
elastic_net_model <- train(
  x = X,
  y = y,
  method = "glmnet",
  trControl = train_control,
  tuneGrid = tune_grid,
  family = "binomial",
  metric = "ROC"
)

# Check selected variables
# elastic_net_selected <- coef(elastic_net_model$finalModel, s = elastic_net_model$bestTune$l
# print(elastic_net_selected[elastic_net_selected != 0])
# Step 4: Get validation performance metrics for each fold
predictions <- elastic_net_model$pred

# View the validation results for each LOOCV fold
# print("Performance metrics for each LOOCV fold:")
# print(predictions)

# Step 5: Calculate overall validation performance
# For example, calculating mean accuracy if available
# mean_accuracy <- mean(predictions$pred == predictions$obs)
# print(paste("Overall mean accuracy on validation set:", mean_accuracy))

# Step 4: Output the selected model results
# print("Best tuning parameters:")
# print(elastic_net_model$bestTune)

# Display coefficients of the final model with selected variables
final_coefficients <- coef(elastic_net_model$finalModel, s = elastic_net_model$bestTune$lamb

selected_coefficients <- final_coefficients[final_coefficients != 0]

# Convert coefficients to a data frame for table display
coef_df <- data.frame(
  Variable = rownames(final_coefficients)[as.vector(final_coefficients != 0)],
  Coefficient = as.vector(selected_coefficients)
)

# Display a nicely formatted table
# kable(coef_df,
#       caption = "Selected Variables with Non-Zero Coefficients",
#       col.names = c("Variable", "Coefficient"),
#       format = "html") %>%
#   kable_styling(full_width = FALSE, bootstrap_options = c("striped", "hover"))
# Step 2: Define the tuning grid for Random Forest
tune_grid <- expand.grid(
  mtry = sqrt(ncol(X))  # Use the square root of predictors as starting point for mtry
)

# Step 3: Train the Random Forest model using caret with LOOCV
rf_model <- train(
  x = X,
  y = y,
  method = "rf",
  trControl = train_control,
  tuneGrid = tune_grid,
  metric = "ROC",  # Optimize for AUC in binary classification
  importance = TRUE
)

# Step 4: Get validation performance metrics for each fold
predictions <- rf_model$pred

# Calculate overall validation performance
# mean_accuracy <- mean(predictions$pred == predictions$obs)
# print(paste("Overall mean accuracy on validation set:", mean_accuracy))

# Step 5: Output the best tuning parameter (mtry)
# print("Best tuning parameters:")
# print(rf_model$bestTune)

# Step 6: Variable Importance
var_importance <- varImp(rf_model, scale = TRUE)
# print(var_importance)

# Step 7: Display a nicely formatted table for variable importance
importance_df <- as.data.frame(var_importance$importance)
colname <- colnames(importance_df)[1]
importance_df <- importance_df[order(-as.numeric(importance_df[[colname]])), ]

kable(importance_df,
      # caption = "Variable Importance in Random Forest Model",
      col.names = c("Variable", "Importance"),
      format = "latex",       # Set to "latex" for PDF compatibility
      booktabs = TRUE) %>%
  kable_styling(full_width = FALSE, bootstrap_options = c("striped", "hover"))
# Step 1: Set up RFE control with SVM
rfe_control <- rfeControl(
  functions = caretFuncs,     # Use generic caret functions as svmFuncs is not built-in
  method = "LOOCV"              # Print progress during the process
)

# Step 2: Define subset sizes to evaluate
# Specify the number of features in each subset that you want to evaluate
subset_sizes <- c(5,6,7,8,9)

# Step 3: Run RFE with SVM model

svm_rfe_model <- rfe(
  x = X,
  y = y,
  sizes = subset_sizes,       # Sizes of predictor subsets to evaluate
  rfeControl = rfe_control,
  method = "svmLinear"        # Use linear SVM model for ranking features
)

# Print the results and selected features
# print(svm_rfe_model)

# Display the selected features in the best model
# print("Selected features:")
# print(predictors(svm_rfe_model))

# Step 4: Plot the results to visualize accuracy vs. subset size
# plot(svm_rfe_model, main = "SVM-RFE: Accuracy vs. Number of Features")
results_value <- c(0.8970588, 0.8970588, 0.926470588235294, 0.9118,
                   0.926470588235294,0.926470588235294, 0.8970588, 0.9118,
                   0.6790123, 0.6790123, 0.6419753, 0.7407,
                   0.7108434,0.7228916,   0.746988, 0.6867)
matrix_data <- matrix(results_value, nrow = 4, ncol = 4, byrow = TRUE)

# Add row and column names
rownames(matrix_data) <- c("ST + Placebo", "BASC + Placebo" , "ST + Varenicline", "BASC + Va
colnames(matrix_data) <- c("Lasso", "Elastic Net", "Random Forest", "SVM-RFE")

knitr::kable(matrix_data,  digits = 3)
# Split data into two groups based on the Var variable
group_varenicline <- project2 %>% filter(Var == 1)
group_placebo <- project2 %>% filter(Var == 0)
# Create a contingency table for the treatment and outcome
```

```
contingency_table <- table(group_varenicline$BA, group_varenicline$abst)
odds_ratio <- oddsratio(contingency_table)$measure[2, 1]  # OR for treatment vs. control
# print(paste("Odds Ratio (OR):", odds_ratio))
# print(paste("Log Odds Ratio (OR):", log(odds_ratio)))

# Fit the initial logistic regression model with interactions
initial_model <- glm(
  abst ~ NMR * BA + readiness * BA + otherdiag * BA + hedonsum_n_pq1 * BA +
    Only.Menthol * BA + ftcd_score * BA,
  data = group_varenicline,   # Replace with `data` if using a specific subset
  family = binomial
)

# Perform stepwise selection based on AIC
stepwise_model <- suppressMessages(suppressWarnings(stepAIC(initial_model, direction = "both"

# Display the summary of the final selected model

summary_table <- tbl_regression(stepwise_model, exponentiate = TRUE) %>%
  modify_header(label = "**Variable**") %>%
  bold_labels()

summary_table
# Create a contingency table for the treatment and outcome
contingency_table <- table(group_placebo$BA, group_placebo$abst)
odds_ratio <- oddsratio(contingency_table)$measure[2, 1]  # OR for treatment vs. control
print(paste("Odds Ratio (OR):", odds_ratio))
print(paste("Log Odds Ratio (OR):", log(odds_ratio)))

# Fit the initial logistic regression model with interactions
initial_model <- glm(
  abst ~ ftcd_score * BA + cpd_ps * BA + Black + mde_curr * BA + NHW * BA + edu * BA + hedons
  data = group_varenicline,   # Replace with `data` if using a specific subset
  family = binomial
)

# Perform stepwise selection based on AIC
stepwise_model <- stepAIC(initial_model, direction = "both")
# Display the summary of the final selected model
summary_table <- tbl_regression(stepwise_model, exponentiate = TRUE) %>%
  modify_header(label = "**Variable**") %>%
  bold_labels()
```

```r
summary_table
# Continuous Variables: Histograms for Age, FTCD Score, CPD, BDI Score
continuous_vars <- c("age_ps", "ftcd_score", "cpd_ps", "bdi_score_w00")

for (var in continuous_vars) {
  p <- ggplot(project2, aes(x = !!sym(var))) +
    geom_histogram(bins = 30, color = "black", fill = "skyblue") +
    labs(title = paste("Distribution of", var), x = var, y = "Frequency") +
    theme_minimal()
  print(p)  # Explicitly print each plot
}
project2_mice <- read.csv("project2.csv")
missing_pattern <- md.pattern(project2_mice)


# Convert missing_pattern to a kable table for cleaner display

#change
# missing_pattern %>%
#   kable("html", caption = "Missing Data Pattern in Project2 Dataset") %>%
#   kable_styling(full_width = FALSE, bootstrap_options = c("striped", "hover"))
```