# Identifying Key Moderators and Predictors of Smoking Cessation in Adults with Major Depressive Disorder

Practical Data Analysis - Project 2: Regression Analysis

Aristofanis Rontogiannis

2024-11-04

**Abstract**

**Purpose**: This project evaluates the efficacy of smoking cessation treatments among adults with Major Depressive Disorder (MDD), focusing on identifying baseline factors that moderate and predict treatment success. This analysis aims to clarify the role of both behavioral and pharmacological interventions, examining interactions between participant characteristics and treatment components.

**Methods**: Using data from a randomized, placebo-controlled trial, we assessed the effectiveness of Behavioral Activation (BA) and standard treatment (ST), each paired with either varenicline or a placebo, in supporting smoking cessation. Three statistical approaches– Lasso regression, stepwise logistic regression, and best subset selection– were employed to identify significant moderators and predictors of abstinence at the end of treatment, based on a sample of 300 adult smokers with current or past MDD.

**Results**: The findings indicate that specific baseline characteristics, including FTCD score (a measure of nicotine dependence) and Nicotine Metabolism Ratio (NMR), significantly influence treatment outcomes. FTCD score was associated with lower abstinence rates, while a higher NMR was linked to greater likelihood of cessation. These predictors were consistently identified across models, suggesting a robust relationship between these factors and smoking cessation outcomes.

**Conclusions**: This study supports the need for tailored smoking cessation strategies for individuals with MDD, highlighting the potential of combining behavioral and pharmacological treatments based on individual characteristics. By identifying critical predictors and moderators, the findings provide a foundation for developing more personalized and effective cessation programs for this high-risk population.

## Introduction

This project is a collaboration with Dr. George Papandonatos from the Department of Biostatistics at Brown University. This project examines the effectiveness of smoking cessation treatments for adults with major depressive disorder (MDD), a group that encounters unique challenges when trying to quit smoking. People with MDD often smoke more heavily, have a stronger dependence on nicotine, and face more intense withdrawal symptoms than those without MDD. While varenicline, a medication for quitting smoking, has shown promising results, psychological approaches that address depression-related issues, like behavioral activation (BA), might further enhance quit rates for this group.

This project uses data from a randomized, placebo-controlled study (Hitsman et al. (2023)) that compared behavioral activation with standard treatment (ST), both combined with either varenicline or a placebo. The study allows us to explore how baseline characteristics may affect treatment success at the end of treatment (EOT), including 300 adult smokers who have current or past MDD. Specifically, we aim to identify baseline factors that may influence the effectiveness of behavioral treatments and predict abstinence, considering both behavioral and medication-based therapies.

The findings from this research could lead to more effective, personalized smoking cessation strategies for people with MDD, addressing their specific challenges and improving treatment outcomes.

## Data Overview

As shown in Table 1 below, participant characteristics were stratified by four treatment combination– behavioral activation with placebo (BASC+placebo), standard treatment with placebo (ST+placebo), behavioral activation with varenicline (BASC+varenicline), and standard treatment with varenicline (ST+varenicline)– and summarized by treatment group in three primary areas: demographics, smoking characteristics, and psychiatric characteristics.

In the demographics section, participant variables such as age, sex, race, income, and education level were included. Age was summarized as a continuous variable using mean and standard deviation, while categorical variables like sex, race, income, and education level were presented with counts and percentages for each treatment group. This section provides an overview of the baseline demographic distribution across the four treatment conditions.

Smoking-related characteristics were summarized to include baseline measures of cigarette consumption, such as cigarettes per day, FTCD score, and readiness to quit smoking. Continuous variables like cigarettes per day and FTCD score were summarized with mean and standard deviation, while categorical variables, such as whether participants smoked within five minutes of waking up, were presented with counts and percentages. This section helps illustrate smoking behaviors across the different treatment groups.

Psychiatric characteristics were also summarized to capture information related to mental health, such as anhedonia (measured by the SHAPS score), presence of other DSM-5 diagnoses, use of antidepressant medication, and current versus past MDD status. Continuous variables, like the SHAPS score, were summarized by mean and standard deviation, while categorical variables, like the presence of other DSM-5 diagnoses, were displayed with counts and percentages. This section highlights the psychiatric characteristics of participants across treatment groups.

With those three sections, Table 1 provides an overall snapshot of participant characteristics by treatment group and by overall sample. This table is organized to present a clear and comprehensive view of the data, allowing for straightforward comparisons across treatment groups and providing context for further analyses on treatment outcomes.

Table 1: Participant characteristics by treatment and overall sample

| Group | Characteristic | Overall N = 300 | BASC+placebo N = 68 | BASC+varenicline N = 83 | ST+placebo N = 68 | ST+varenicline N = 81 |
|---|---|---|---|---|---|---|
| Demographics | Age (years) | 50.0 (12.6) | 50.7 (13.5) | 50.3 (13.2) | 50.3 (10.8) | 48.7 (12.7) |
| | Sex (Female) | 165 (55%) | 38 (56%) | 44 (53%) | 39 (57%) | 44 (54%) |
| | Race | | | | | |
| | Non-Hispanic White | 105 (35%) | 24 (35%) | 34 (41%) | 22 (32%) | 25 (31%) |
| | Black | 157 (52%) | 37 (54%) | 37 (45%) | 40 (59%) | 43 (53%) |
| | Hispanic | 16 (5.3%) | 4 (5.9%) | 3 (3.6%) | 4 (5.9%) | 5 (6.2%) |
| | Other | 22 (7.3%) | 3 (4.4%) | 9 (11%) | 2 (2.9%) | 8 (9.9%) |
| | Income | | | | | |
| | Less than $20,000 | 110 (37%) | 25 (37%) | 30 (37%) | 26 (38%) | 29 (36%) |
| | $20,000–$35,000 | 68 (23%) | 16 (24%) | 17 (21%) | 14 (21%) | 21 (26%) |
| | $35,001–$50,000 | 46 (15%) | 8 (12%) | 13 (16%) | 14 (21%) | 11 (14%) |
| | $50,001–$75,000 | 38 (13%) | 12 (18%) | 12 (15%) | 8 (12%) | 6 (7.5%) |
| | More than $75,000 | 35 (12%) | 6 (9.0%) | 10 (12%) | 6 (8.8%) | 13 (16%) |
| | Education | | | | | |
| | Grade school | 1 (0.3%) | 1 (1.5%) | 0 (0%) | 0 (0%) | 0 (0%) |
| | Some high school | 16 (5.3%) | 3 (4.4%) | 7 (8.4%) | 2 (2.9%) | 4 (4.9%) |
| | High school graduate or GED | 76 (25%) | 23 (34%) | 15 (18%) | 11 (16%) | 27 (33%) |
| | Some college/technical school | 116 (39%) | 22 (32%) | 32 (39%) | 38 (56%) | 24 (30%) |
| | College graduate | 91 (30%) | 19 (28%) | 29 (35%) | 17 (25%) | 26 (32%) |
| Smoking | Cigarettes per day at baseline phone survey | 15.1 (7.9) | 15.6 (9.1) | 15.5 (8.5) | 15.0 (7.2) | 14.4 (6.6) |
| | FTCD score at baseline | 5.2 (2.1) | 5.3 (2.0) | 5.1 (2.3) | 5.4 (2.1) | 5.2 (2.1) |
| | Smoking within 5 mins of waking up (Yes) | 138 (46%) | 32 (47%) | 33 (40%) | 35 (51%) | 38 (47%) |
| | BDI score at baseline | 18.7 (11.5) | 19.0 (12.3) | 18.0 (10.6) | 18.5 (10.8) | 19.5 (12.2) |
| | Cigarette reward value at baseline | 7.2 (3.7) | 7.4 (3.8) | 7.2 (3.9) | 7.0 (3.7) | 7.1 (3.5) |
| | Pleasurable Events Scale - substitute reinforcers | 22.6 (19.6) | 23.2 (20.3) | 22.9 (19.0) | 20.8 (20.1) | 23.4 (19.5) |
| | Pleasurable Events Scale - complementary reinforcers | 25.4 (19.4) | 27.7 (21.5) | 22.4 (17.0) | 27.4 (19.9) | 25.0 (19.4) |
| | Exclusive Mentholated Cigarette User (Yes) | 178 (60%) | 40 (59%) | 48 (59%) | 43 (64%) | 47 (58%) |
| | Readiness to quit smoking | 6.8 (1.2) | 6.8 (1.4) | 6.7 (1.2) | 7.0 (1.3) | 6.7 (1.1) |
| | Nicotine Metabolism Ratio | 0.4 (0.2) | 0.3 (0.2) | 0.4 (0.2) | 0.4 (0.3) | 0.4 (0.2) |
| Psychiatric | Anhedonia | 2.2 (3.2) | 2.2 (3.2) | 2.3 (3.1) | 2.5 (3.4) | 2.1 (3.0) |
| | Other lifetime DSM-5 diagnosis (Yes) | 133 (44%) | 35 (51%) | 30 (36%) | 28 (41%) | 40 (49%) |
| | Taking antidepressant medication at baseline (Yes) | 82 (27%) | 28 (41%) | 24 (29%) | 15 (22%) | 15 (19%) |
| | Current vs past MDD (Yes) | 147 (49%) | 32 (47%) | 40 (48%) | 31 (46%) | 44 (54%) |

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

Additionally, to facilitate certain analyses, a new variable was created to consolidate education levels. The three lower levels of education (Grade school, Some high school and high school graduate or GED) were combined into a single category representing lower education levels. This aggregation simplifies the education variable, as those three levels contained a small amount of participants, allowing for broader categorical comparisons while retaining essential information on educational background.

## Data Missingness and Imputation

The missing data analysis involves examining the extent of missingness across variables in the dataset. A summary table was generated (Table 2), listing variables with missing values, the count of missing entries for each variable, and the corresponding percentage relative to the total sample size. For example, the variable with the highest missingness is the variable NMR (7%), followed by the variables crv_total_pq1 (6%) and readiness (5.67%). Variables with no missing values were excluded from this table for clarity. This missingness analysis is a critical step in data preparation, as it helps to address data quality issues and ensure that the dataset is ready for further statistical modeling and interpretation. The total percentage of missingness was around 20%. If we have deleted all the rows with missing data, we might have lost a significant portion of our dataset, which could reduce the statistical power of our analyses and lead to biased estimates.

Table 2: Summary of Missing Values

| Variable | Number | Pct |
|---|---|---|
| missing_NMR | 21 | 7.00% |
| missing_crv_total_pq1 | 18 | 6.00% |
| missing_readiness | 17 | 5.67% |
| missing_inc | 3 | 1.00% |
| missing_shaps_score_pq1 | 3 | 1.00% |
| missing_Only.Menthol | 2 | 0.67% |
| missing_ftcd_score | 1 | 0.33% |

To handle missing data, the Multiple Imputation by Chained Equations (MICE) method was applied. MICE is an iterative process that generates multiple plausible datasets by imputing missing values based on the observed data structure. Here, we used predictive mean matching (PMM) as the imputation method. PMM is advantageous as it imputes realistic values by using observed values from other participants with similar predictive values.

## Exploration of potential interactions

As part of the Explanatory Data Analysis, we want to examine the relationship between menthol cigarette use and race. We created a contingency table to display the frequencies of menthol and non-menthol use across racial categories, and then performed a Chi-square test to assess whether there is a statistically significant association between race and menthol cigarette use.

The Chi-square test produced a very low p-value, indicating a statistically significant association between race and menthol cigarette use. This result suggests that menthol cigarette usage depends on race, as there appears to be a notable relationship between the two variables.

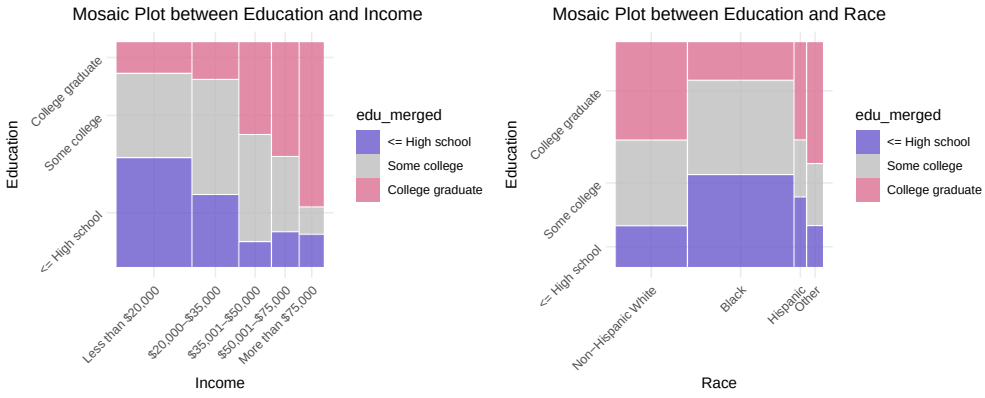
We can easily observe from Table 3 that Black individuals have a preference for menthol cigarettes over regular non-menthol ones (130 out of 157 people).

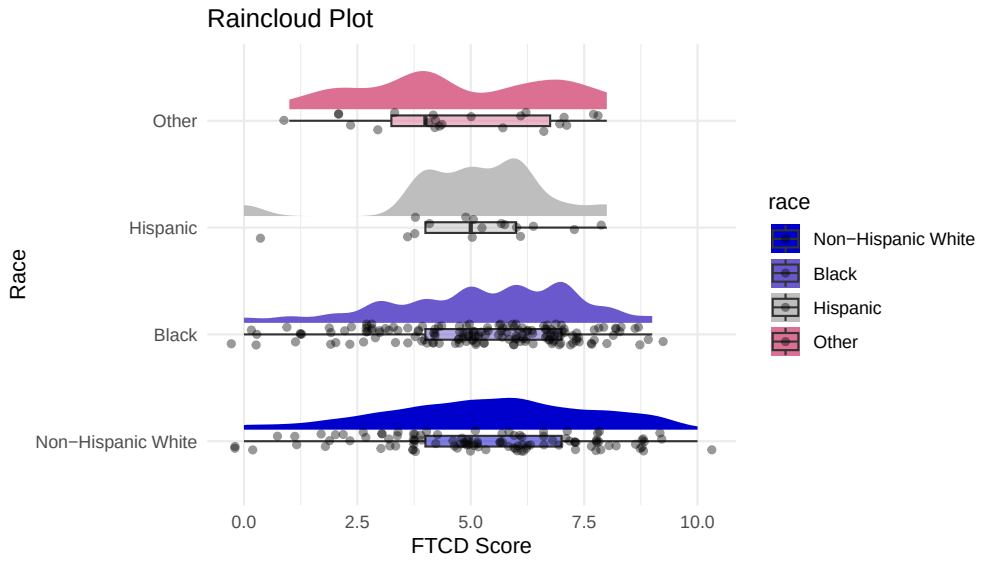
Table 3: Contingency Table of Only Menthol Use and Race

| | Non-Menthol | Menthol |
|---|---|---|
| Non-Hispanic White | 72 | 33 |
| Black | 27 | 130 |
| Hispanic | 12 | 4 |
| Other | 9 | 13 |

*Note:*
Chi-Square Statistic: 77.9023, p-value : Approaching 0.0000

Additionally, we examine the relationships between education, income, race, and FTCD scores through mosaic and raincloud plots.




**Mosaic Plot between Education and Income** This plot shows the distribution of education levels (edu_merged) across different income categories. We observe a gradient where higher education levels (coded in pink) tend to be associated with higher income levels, while lower education levels (in blue) were more concentrated in lower income brackets. This suggests a positive association between education and income.

**Mosaic Plot between Education and Race** This plot reveals the distribution of education levels across different racial groups. The distribution varies, with Non-Hispanic Whites showing a relatively higher proportion in the higher education categories, while other groups, such as Black and Hispanic, have a more balanced spread across the education levels. This pattern may indicate disparities in educational attainment across racial groups.

**Raincloud Plot for FTCD Score by Race** The raincloud plot shows the distribution and density of FTCD scores across racial groups. Non-Hispanic Whites have the highest density around higher FTCD scores, while other groups show a more varied distribution, with Hispanics and Black participants displaying lower scores overall. This visualization highlights potential differences in FTCD scores among racial groups, which may correlate with other socioeconomic factors.

# Methods

## Data separation and full model selection

To assess the model's performance and ensure its generalizability, we divided the data into training and test sets, with 70% of the observations assigned to the training set and 30% to the test set. This split allows us to train the model on one portion of the data and then evaluate its performance on a separate, unseen portion, reducing the risk of overfitting.

We ensured that the variables Var and BA are kept controlled in the model as those variables are crucial for investigating treatment interactions and their effect on abstinence. For objective one, we also added interaction terms between BA and baseline variables to explore potential moderator effects and other interaction terms, e.g., NMR:readiness and income:education, as reasonable covariates. For the second objective of this project, we considered baseline characteristics as potential predictors while keeping BA and Var controlled. Thus, for simplicity, no interaction terms were included in the full models.

## Lasso Regression

In this part of the analysis, we implemented a Lasso regression to identify potential moderators of the treatment effects on smoking abstinence. By using Lasso, we aim to reduce the model to a subset of variables and interaction terms that are most predictive of the outcome. Lasso is particularly useful here as it performs variable selection by penalizing the coefficients of less relevant variables, ultimately setting them to zero, which leads to a more interpretable model.

The training set was used to build the Lasso model, optimizing the selection of variables and interactions by identifying the subset that most effectively predicts smoking abstinence. Cross-validation on the training data determined the best penalty parameter (lambda), balancing the trade-off between model complexity and predictive accuracy.

After training the model, we evaluated it on the test set, providing an unbiased assessment of how well the selected variables and interactions generalize to new data. This approach ensures that the model is robust and capable of making reliable predictions beyond the original dataset.

The final model coefficients at the optimal lambda include only those variables and interactions with non-zero coefficients, which are presented in the resulting tables (Table 4 and Table 5). This selection helps identify significant moderators of smoking abstinence while maintaining a simpler model.

## Stepwise Logistic Regression

Furthermore, we performed stepwise logistic regression to determine the optimal combination of variables and interaction terms for predicting smoking abstinence. The process starts by defining both the outcome variable

and a set of moderator and predictor variables. Specifically, the outcome variable represents abstinence, and a set of relevant predictor variables is selected from our dataset based on their potential influence on the outcome. These predictors include demographic information, baseline characteristics, and other relevant factors. The model is built to include both main effects (the predictors' direct effects on abstinence) and potential interaction terms. The main effects include the variables Var and BA, that should be controlled in all models. We also used additional predictors like age, gender, income, education level, race, and other baseline measures. Additionally, a set of interaction terms is specified, particularly those involving BA and Var, to explore moderation effects. These interaction terms allow the model to capture relationships where the effect of one predictor on abstinence might depend on the level of another predictor, adding a layer of complexity to the model.

Using the step function, the model undergoes stepwise logistic regression, which involves both forward and backward selection to find the most effective model. This approach adds or removes predictors and interactions iteratively, aiming to optimize the model based on the Akaike Information Criterion (AIC).

To sum up, this stepwise approach optimizes the logistic regression model by keeping the most relevant predictors and discarding those that do not improve the model's performance, leading to an efficient and interpretable model for predicting smoking abstinence.

### Best Subset Selection

Lastly, we performed best subset selection to identify the most predictive set of variables for smoking abstinence. For the first objective, we used sequential replacement (it was not possible to use every possible combination of variables like we did in the second objective– exhaustive replacement– as the number of interactions could be over a million) to find the one that best explains the outcome with a balance between predictive power and model complexity.

The selection process started with defining the outcome variable and predictor variables in the training dataset. Then we examined subsets of the predictors, looking for the best combination based on specific criteria. Here, Var1 and BA1 are forced into the model, meaning these terms will always be included in the subset selection process due to their importance as main effects. After that, we identified the subset with the minimum Mallows' $Cp$ statistic, which is often used to select a model with good fit while avoiding overfitting.

In essence, this approach evaluates possible subsets of predictors and selects the combination with the lowest $Cp$, yielding a model that is both predictive and efficient for determining smoking abstinence.

## Results

Table 4 compares the coefficients of variables selected by the three different modeling approaches and their corresponding Odds Ratios: logistic regression with stepwise selection, Lasso regression, and best subset selection. This combined table allows for easy comparison of the variables who are considered the most predictive across these methods and their respective coefficients.

We can observe that except BA1 and Var1 (obviously - we forced them) there are no other common variables between all three model selection methods. However, there are two common variables between Lasso and Stepwise Regression (ftcd_score, NMR). FTSD score's OR is smaller than 1 in both Lasso and Stepwise Regression, meaning that participants with higher cigarette dependence would have lower success rate of smoking abstinence. On the opposite, the OR of NMR is greater than 1 in both methods, meaning that participants with higher nicotine metabolism ratio would have a higher possibility of quitting smoking. The opposite directionality between these two variables' impact on smoking cessation showcases the potential biological relation between baseline characteristic and the outcome. Except from that, every other variable is present to one method solely, indicating that the three methods used different selection criteria and thus lead to high variability in variables/interaction terms selected.

Table 4: Summary of Coefficients and Odds Ratios across 3 Sel ection Methods (for Moderators)

| Variables | Lasso | Stepwise | Best Subset | Lasso OR | Stepwise OR | Best Subset OR |
|---|---|---|---|---|---|---|
| Var1 | 1.4897 | -0.3142 | 0.2012 | 4.4359 | 0.7304 | 1.2229 |
| BA1 | -0.251 | -0.1942 | -0.0473 | 0.778 | 0.8235 | 0.9538 |
| ftcd_score | -0.1302 | -0.6058 | | 0.8779 | 0.5456 | |
| NMR | 0.2036 | 1.2302 | | 1.2258 | 3.4221 | |
| inc2:edu_merged3 | -0.0564 | | | 0.9452 | | |
| inc5:edu_merged3 | 0.218 | | | 1.2436 | | |
| mde_curr1:readiness | -0.035 | | | 0.9656 | | |
| edu_merged2:Only.Menthol1 | -0.2817 | | | 0.7545 | | |
| mde_curr1 | | -0.8037 | | | 0.4476 | |
| Var1:ftcd_score | | 0.4471 | | | 1.5637 | |
| edu_merged2 | | | -0.0373 | | | 0.9633 |
| BA1:otherdiag1 | | | -0.0076 | | | 0.9924 |
| Var1:sex_ps2 | | | 0.0139 | | | 1.014 |
| inc3:edu_merged2 | | | -0.0558 | | | 0.9457 |
| antidepmed1:readiness | | | -7e-04 | | | 0.9993 |
| ftcd.5.mins1:readiness | | | -0.0105 | | | 0.9896 |
| sex_ps2:Only.Menthol1 | | | 0.0131 | | | 1.0131 |
| raceBlack:Only.Menthol1 | | | -0.0896 | | | 0.9143 |
| raceOther:Only.Menthol1 | | | -0.23 | | | 0.7945 |
| edu_merged3:Only.Menthol1 | | | 0.0809 | | | 1.0842 |
| NMR:readiness | | | 0.0139 | | | 1.014 |

The plots below provide a comprehensive comparison of three model selection methods — Lasso, Stepwise, and Best Subset — in predicting smoking abstinence.

### Calibration Plots (Left Panel)

**Lasso Model** The calibration plot shows that the Lasso model's predicted probabilities generally align with the observed smoking abstinence rates, but there is noticeable variability, especially at higher probability estimates. The model tends to slightly underpredict abstinence at lower probabilities and demonstrates a wider spread of actual values as probabilities increase.

**Stepwise Model** This model's calibration plot shows a pretty good alignment with the ideal calibration line, suggesting that it produces probability estimates that can reflect actual abstinence outcomes across most probability ranges. The error bars indicate stable estimates with less variability, making the Stepwise model the most reliable in terms of calibration.

**Best Subset Model** The calibration plot for the Best Subset model reveals a greater discrepancy from the ideal line and wider error bars compared to the other models. This model demonstrates higher variability and less reliable probability estimates, particularly at the extreme ends of the probability range.

### LOESS-Smoothing Calibration Curves (Middle Panel)

**Lasso Model** The LOESS curve for the Lasso model shows underprediction of abstinence at lower probabilities, with the curve staying below the ideal line at the low end. The model's predictions improve slightly as probabilities increase but still demonstrate some deviation, indicating inconsistencies in predicted abstinence probability across ranges.

**Stepwise Model** The Stepwise model's LOESS curve stays close to the ideal line, suggesting a high degree of calibration accuracy. The curve remains close to the 45-degree line across the range of probabilities, indicating that predicted probabilities are well-calibrated and reflective of actual abstinence outcomes.

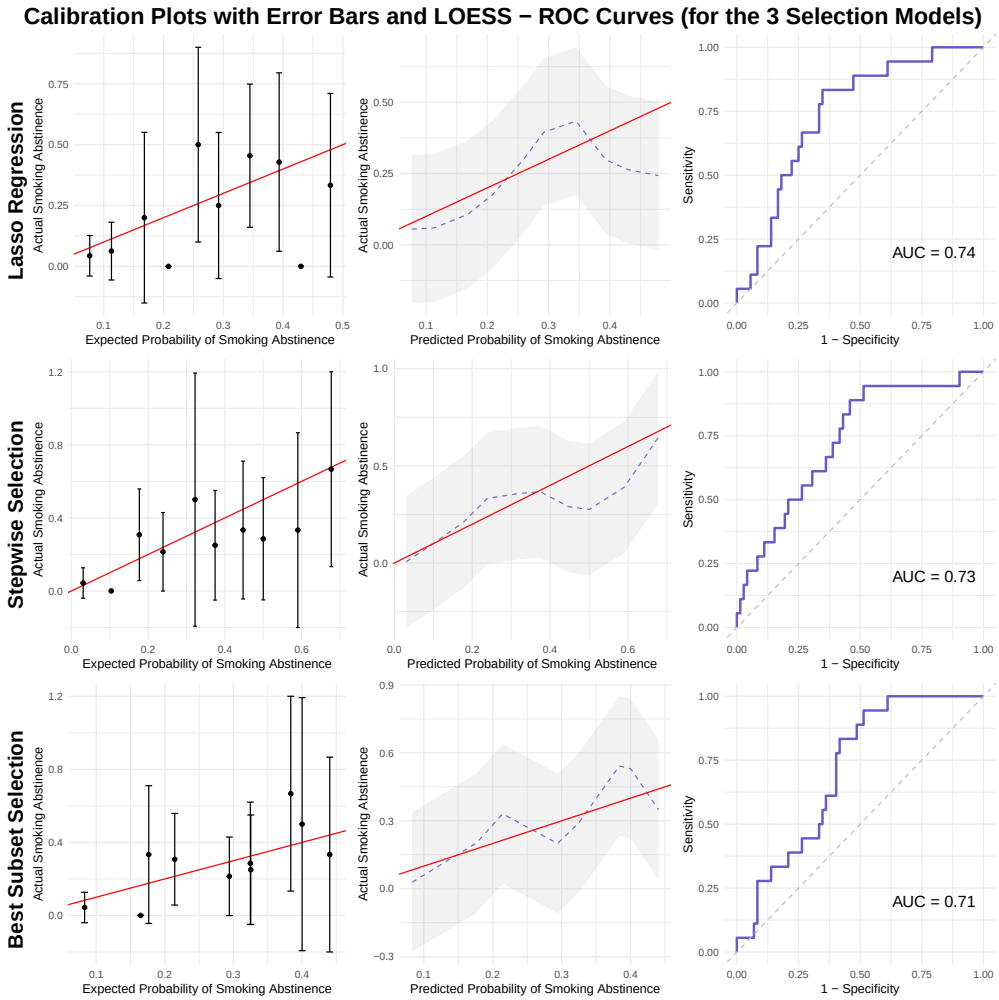
**Best Subset Model** The LOESS curve for the Best Subset model deviates from the ideal line, particularly at the lower and upper probability ranges, suggesting substantial miscalibration. The curve shows greater fluctuations, indicating that the model's predictions are not consistent with observed abstinence probabilities.

### ROC Curves (Right Panel)

**Lasso Model** The ROC curve for the Lasso model yields an AUC of 0.74, indicating that this model has the best discrimination ability among the three models. It suggests that the Lasso model can effectively distinguish between abstinent and non-abstinent individuals.

**Stepwise Model** The Stepwise model's ROC curve has an AUC of 0.73, very close to the Lasso model. While slightly lower, this AUC still represents good discrimination, indicating that the Stepwise model can differentiate between abstinent and non-abstinent cases with reasonable accuracy.

**Best Subset Model** The Best Subset model has an AUC of 0.71, the lowest among the three models. This suggests that it has the weakest discrimination capability, potentially leading to more misclassification of abstinence status.




For the second objective, we explored baseline characteristic as potential predictors. Overall, we followed similar analysis procedure as in objective one, but due to lower model complexity, we could employ exhaustive replacement in best subset selection. Table 5 compares the coefficients of variables selected by the three different modeling approaches and their corresponding ORs. On this approach, we observed that three

variables (ftcd_score, mde_curr and NMR) were present in all three models. As previously discussed, FTCD score and NMR could be potentional predictors for predicting the outcome. Also, the OR of mde_curr is less than 1 in all three methods, meaning mde_curr has a similar behavior with ftcd_score. This means that participants with current MDD would have a lower success rate of smoking abstinence than those with past MDD. Finally, Var has a consistently >1 OR in all three methods used, while BA did not show a positive impact on abstinence, which is consistent with the findings from the original study (Hitsman et al. (2023)).

Table 5: Summary of Coefficients and Odds Ratios across 3 Selection Methods (for Predictors)

| Variables | Lasso | Stepwise | Best Subset | Lasso OR | Stepwise OR | Best Subset OR |
|---|---|---|---|---|---|---|
| Var1 | 1.5131 | 1.6194 | 0.2164 | 4.5407 | 5.0499 | 1.2416 |
| BA1 | -0.2506 | -0.2686 | -0.0396 | 0.7783 | 0.7645 | 0.9611 |
| inc2 | -0.0123 | | | 0.9878 | | |
| ftcd_score | -0.1221 | -0.2717 | -0.0405 | 0.8850 | 0.7621 | 0.9604 |
| mde_curr1 | -0.1637 | -0.8116 | -0.1089 | 0.8490 | 0.4442 | 0.8969 |
| NMR | 0.1163 | 1.1982 | 0.1736 | 1.1233 | 3.3141 | 1.1896 |

## Conclusions and Limitations

This study underscores the potential benefits of tailored smoking cessation interventions for individuals with Major Depressive Disorder (MDD). Our analysis revealed that baseline characteristics, particularly FTCD score and Nicotine Metabolism Ratio (NMR), significantly impact abstinence outcomes. Higher nicotine dependence was associated with lower cessation success, while a higher NMR indicated a greater likelihood of quitting. These findings suggest that individualized treatments, incorporating both pharmacotherapy and behavioral strategies, may be more effective for this high-risk population.

The use of multiple modeling approaches— Lasso, Stepwise Logistic Regression, and Best Subset Selection— allowed for a robust evaluation of predictive factors, capturing both common and unique predictors across methods. While Behavioral Activation and varenicline showed mixed effects on abstinence rates, variability in predictor selection highlights the complexity of predicting cessation success in individuals with MDD. This suggests that integrating insights from different models can help identify a broader range of influential factors, enhancing the personalization of treatment approaches.

However, limitations of this study include a moderate sample size and reliance on self-reported smoking data, which may limit generalizability and introduce potential biases. Furthermore, the high number of interaction terms included in the full model for moderator effect exploration could introduce the issue of overfitting. Future research should expand on these findings by employing larger samples, objective measures of smoking, and long-term follow-up. Further exploration of additional predictors, such as genetic markers or psychiatric profiles, could provide deeper insights into treatment responses, ultimately guiding the development of optimized cessation strategies for those with MDD.

## Data Privacy and Code Availability

The analysis dataset was obtained by Dr. George Papandonatos from the Department of Biostatistics at Brown University and cannot be shared due to privacy. The replication code can be found at https://github.com/AristofanisR/Practical_Data_Analysis_Project2

# References

Hitsman, Brian, George D Papandonatos, Jacqueline K Gollan, Mark D Huffman, Raymond Niaura, David C Mohr, Anna K Veluz-Wilkins, et al. 2023. "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." *Addiction* 118 (9): 1710–25.

## Code Appendix

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

library(summarytools)
library(ggplot2)
library(knitr)
library(kableExtra)
library(GGally)
library(patchwork)
library(dplyr)
library(reshape2)
library(tidyr)
library(grid)
library(lubridate)
library(gtsummary)
library(gt)
library(ggcorrplot)
library(glmnet)
library(MASS)
library(pROC)
library(gridExtra)
library(VIM)
library(mice)
library(leaps)
library(ggplot2)
library(ggmosaic)
library(ggbeeswarm)
library(ggdist)
library(cowplot)
#Read main data set
data<-read.csv("project2.csv")

# Data processing
data = data  %>%
  # create race variable
  mutate(race = factor(case_when(
    NHW == 1 ~ "Non-Hispanic White",
    Black == 1 ~ "Black",
    Hisp == 1 ~ "Hispanic",
    TRUE ~ "Other"  # Handle cases where none of the above conditions are met
  ), levels = c("Non-Hispanic White", "Black", "Hispanic", "Other"))) %>%
  # create treatment categories
  mutate(treatment_cat = factor(case_when(BA == 1 & Var == 0 ~ "BASC+placebo",
                                  BA == 0 & Var == 0 ~ "ST+placebo",
                                  BA == 1 & Var == 1 ~ "BASC+varenicline",
                                  BA == 0 & Var == 1 ~ "ST+varenicline"))) %>%
  # Change variables attributes to be only Numeric or Factor at the end
  #Factor
  mutate(
```

```
    abst = factor(abst),
    Var = factor(Var),
    BA = factor(BA),
    sex_ps = factor(sex_ps),
    ftcd.5.mins = factor(ftcd.5.mins),
    otherdiag = factor(otherdiag),
    antidepmed = factor(antidepmed),
    mde_curr = factor(mde_curr),
    Only.Menthol = factor(Only.Menthol),
    edu = factor(edu, levels = c(1, 2, 3, 4, 5)),
    inc = factor(inc, levels = c(1, 2, 3, 4, 5))
  ) %>%
 #Numeric (except id)
  mutate(across(
    .cols = where(is.numeric) & !all_of("id"),
    .fns = as.numeric
  ))

# Summary Table
table1_data = data %>%
  mutate(edu = factor(edu, levels = c(1, 2, 3, 4, 5),
                  labels = c("     Grade school",
                             "     Some high school",
                             "     High school graduate or GED",
                             "     Some college/technical school",
                             "     College graduate")),
    inc = factor(inc, levels = c(1, 2, 3, 4, 5),
                  labels = c("     Less than $20,000",
                             "     $20,000-$35,000",
                             "     $35,001-$50,000",
                             "     $50,001-$75,000",
                             "     More than $75,000")),
    race = factor(race, labels = c("    Non-Hispanic White",
                                   "    Black",
                                   "    Hispanic",
                                   "    Other"))
    )

# Demographics table
demographics_table <- table1_data %>%
  dplyr::select(
    treatment_cat,
    age_ps,
    sex_ps,
    race,
    inc,
    edu
  ) %>%
  tbl_summary(
    by = treatment_cat,
    label = list(
      age_ps = "Age (years)",
      sex_ps = "Sex (Female)",
```

```
      race = "Race",
      inc = "Income",
      edu = "Education"
    ),
    type = list(
      age_ps ~ "continuous",
      sex_ps ~ "dichotomous",
      race ~ "categorical",
      inc ~ "categorical",
      edu ~ "categorical"),
    value = list(
      sex_ps ~ "2"),
    statistic = list(all_continuous() ~ "{mean} ({sd})",
                     all_categorical() ~ "{n} ({p}%)"),
    digits = all_continuous() ~ 1,
    missing = "no"
  ) %>% add_overall()


# Smoking table
smoking_table <- table1_data %>%
  dplyr::select(
    treatment_cat,
    cpd_ps,
    ftcd_score,
    ftcd.5.mins,
    bdi_score_w00,
    crv_total_pq1,
    hedonsum_n_pq1,
    hedonsum_y_pq1,
    Only.Menthol,
    readiness,
    NMR
  ) %>%
  tbl_summary(
    by = treatment_cat,
    label = list(
      cpd_ps = "Cigarettes per day at baseline phone survey",
      ftcd_score = "FTCD score at baseline",
      ftcd.5.mins = "Smoking within 5 mins of waking up (Yes)",
      bdi_score_w00 = "BDI score at baseline",
      crv_total_pq1 = "Cigarette reward value at baseline",
      hedonsum_n_pq1 = "Pleasurable Events Scale - substitute reinforcers",
      hedonsum_y_pq1 = "Pleasurable Events Scale - complementary reinforcers",
      Only.Menthol = "Exclusive Mentholated Cigarette User (Yes)",
      readiness = "Readiness to quit smoking",
      NMR = "Nicotine Metabolism Ratio"
    ), type = list(
      cpd_ps ~ "continuous",
      ftcd_score ~ "continuous",
      ftcd.5.mins ~ "dichotomous",
      bdi_score_w00 ~ "continuous",
      crv_total_pq1 ~ "continuous",
```

```
      hedonsum_n_pq1 ~ "continuous",
      hedonsum_y_pq1 ~ "continuous",
      NMR ~ "continuous",
      Only.Menthol ~ "dichotomous",
      readiness ~ "continuous"),
    value = list(
      Only.Menthol ~ "1",
      ftcd.5.mins ~ "1"),
    statistic = list(all_continuous() ~ "{mean} ({sd})", all_categorical() ~ "{n} ({p}%)"),
    digits = all_continuous() ~ 1,
    missing = "no"
  ) %>% add_overall()

# Psychiatric table
psychiatric_table <- table1_data %>%
  dplyr::select(
    treatment_cat,
    shaps_score_pq1,
    otherdiag,
    antidepmed,
    mde_curr
  ) %>%
  tbl_summary(
    by = treatment_cat,
    label = list(
      shaps_score_pq1 = "Anhedonia",
      otherdiag = "Other lifetime DSM-5 diagnosis (Yes)",
      antidepmed = "Taking antidepressant medication at baseline (Yes)",
      mde_curr = "Current vs past MDD (Yes)"
    ),
    type = list(shaps_score_pq1 ~ "continuous",
      otherdiag ~ "dichotomous",
      antidepmed ~ "dichotomous",
      mde_curr ~ "dichotomous"),
    value = list(otherdiag ~ "1",
      antidepmed ~ "1",
      mde_curr ~ "1"),
    statistic = list(all_continuous() ~ "{mean} ({sd})",
                     all_categorical() ~ "{n} ({p}%)"),
    digits = all_continuous() ~ 1,
    missing = "no"
  ) %>% add_overall()


# Merge tables
final_table <- tbl_stack(
  tbls = list(demographics_table, smoking_table, psychiatric_table),
  group_header = c("Demographics", "Smoking", "Psychiatric")
) %>%
  modify_caption("Participant characteristics by treatment and overall sample") %>%
  as_kable_extra(
    booktabs = TRUE,
    longtable = TRUE,
```

```
    linesep = "",
    format = "latex"
  ) %>%
  kable_styling(
    position = "center",
    latex_options = c("striped", "repeat_header"),
    stripe_color = "gray!15",
    font_size = 8
  )


final_table
#Missingness Table
# Calculate missing values for each variable
missing_summary <- data %>%
  summarise(across(everything(), ~ sum(is.na(.)), .names = "missing_{col}")) %>%
  pivot_longer(cols = everything(), names_to = "Variable", values_to = "Number") %>%
  mutate(Pct = (Number / nrow(data)) * 100) %>%
  filter(Number > 0)  # Exclude variables with 0 missingness

# Define a named vector with old and new names for variables
variable_names <- c(
  "inc" = "Income",
  "ftcd_score" = "FTCD Score",
  "crv_total_pqr" = "Cigarette reward at baseline",
  "shaps_score_pqi" = "Anhedonia",
  "NMR" = "Nicotine MEtabolism Ratio",
  "Only.Menthol" = "Exclusive Mentholated Cigarette User",
  "readiness" = "Baseline readiness to quit smoking"
)

# Rename variables in the summary table
missing_summary <- missing_summary %>%
  mutate(Variable = recode(Variable, !!!variable_names))

# Load knitr for table formatting (optional)
library(knitr)

# Create and display the table
missing_summary %>%
  arrange(desc(Pct)) %>%
  mutate(Pct = sprintf("%.2f%%", Pct)) %>%
  kable(col.names = c("Variable", "Number", "Pct"), caption = "Summary
        of Missing Values")
# Create a new variable that contains the 3 first levels of edu
data <- data %>%
  mutate(edu_merged = factor(case_when(
    edu %in% c("1", "2", "3") ~ "1",
    edu == "4" ~ "2",
    edu == "5" ~ "3"
  )))
```

```r
# MICE
# Perform MICE imputation
data_mice <- mice(data, m = 5
                  , method = "pmm", maxit = 50, seed = 58, printFlag= FALSE)

# Complete the data by extracting one of the imputed datasets
data_mice <- complete(data_mice, action = 4)
# Check the relationship between Menthol Cigarettes and Race
table_race_menthol <- table(data_mice$race, data_mice$Only.Menthol)

# Chi-square test
chi_square_test <- chisq.test(table_race_menthol) #p-value too small
#We reject the null hypothesis (which is there is no association)

chi_square_test_note <- paste0(
  sprintf("Chi-Square Statistic: %.4f", chi_square_test$statistic),
  sprintf(", p-value : Approaching %.4f", chi_square_test$p.value)
)

kable(table_race_menthol,
      caption = "Contingency Table of Only Menthol Use and Race",
      col.names = c("Non-Menthol", "Menthol"),
      row.names = TRUE,
      format = "markdown")  %>%
  footnote(chi_square_test_note, footnote_as_chunk = FALSE)

# Mosaic between education, income
# mosaic = vcd::mosaic(~ inc + edu_merged, data = data_mice, shade = TRUE,
#                      legend = TRUE, labeling = labeling_values)

data_viz = data_mice %>%
  mutate(
    edu_merged = recode(factor(edu_merged),
                        `1` = "<= High school",
                        `2` = "Some college",
                        `3` = "College graduate"),
    inc = recode(factor(inc),
                 `1` = "Less than $20,000",
                 `2` = "$20,000-$35,000",
                 `3` = "$35,001-$50,000",
                 `4` = "$50,001-$75,000",
                 `5` = "More than $75,000"),
    race = factor(race, levels = c("Non-Hispanic White",
                                   "Black",
                                   "Hispanic",
                                   "Other"))
    )

# Heatmap between education, income
heatmap1 = ggplot(data = data_viz) +
  geom_mosaic(aes(weight = 1, x = product(inc), fill = edu_merged)) +
  labs(
    title = "           Mosaic Plot between Education and Income",
```

```
    x = "Income",
    y = "Education"
  ) +
  scale_fill_manual(values = c("slateblue3", "grey", "palevioletred")) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    axis.text.y = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5)
  )

# Heatmap between race, education
heatmap2 = ggplot(data = data_viz) +
  geom_mosaic(aes(weight = 1, x = product(race), fill = edu_merged)) +
  labs(
    title = "Mosaic Plot between Education and Race",
    x = "Race",
    y = "Education"
  ) +
  scale_fill_manual(values = c("slateblue3", "grey", "palevioletred")) +
  theme_minimal() +
  theme(
    axis.text.x = element_text(angle = 45, hjust = 1),
    axis.text.y = element_text(angle = 45, hjust = 1),
    plot.title = element_text(hjust = 0.5)
  )

# Beeswarm plot between Sex, NMR
#beeswarm = ggplot(data_mice, aes(x = sex_ps, y = NMR, color = sex_ps)) +
 # geom_beeswarm(size = 2, alpha = 0.7) +
  #labs(x = "Sex", y = "NMR", title = "Beeswarm Plot") +
  #theme_minimal() +
  #theme(legend.position = "none") +
  #scale_color_manual(values = c("slateblue3", "palevioletred"))



# Raincloud plot between edu, income
raincloud = ggplot(data_mice, aes(x = race, y = ftcd_score, fill = race)) +
  ggdist::stat_halfeye(adjust = 0.5, width = 0.6, .width = 0,
                       justification = -0.2, point_colour = NA) +
  geom_boxplot(width = 0.1, outlier.shape = NA, alpha = 0.5) +
  geom_jitter(width = 0.1, alpha = 0.4) +
  coord_flip() +  # Flip coordinates for a horizontal layout
  labs(x = "Race", y = "FTCD Score", title = "Raincloud Plot") +
  theme_minimal() +
  scale_fill_manual(values = c("mediumblue","slateblue3", "grey", "palevioletred"))


combined_plot <- plot_grid(heatmap1, heatmap2, ncol = 2, align = "hv")
combined_plot

#combined_plot <- grid.arrange(mosaic, heatmap, beeswarm, raincloud, ncol = 2)
```

```r
#combined_plot
#par(mfrow= c(2,2))
#mosaic

raincloud
# Modeling Preprocessing
# Define the outcome and variables in the model
outcome <- data_mice$abst
variable_names <- c("Var", "BA", "age_ps", "sex_ps", "inc", "edu_merged", "race",
                    "ftcd_score", "ftcd.5.mins", "bdi_score_w00", "cpd_ps",
                    "crv_total_pq1", "hedonsum_n_pq1", "hedonsum_y_pq1",
                    "shaps_score_pq1", "otherdiag", "antidepmed", "mde_curr",
                    "NMR", "Only.Menthol", "readiness")
variables <- data_mice[, variable_names]
# for Lasso (to break down factors with >2 levels)
variables_dummy <- model.matrix(~ 0 + ., data = variables)
# remove the extra reference group
variables_dummy <- variables_dummy[, -which(colnames(variables_dummy) =="Var0")]

# Split into train and test
set.seed(58)
train_index <- sample(1:nrow(data_mice), 0.7 * nrow(data_mice))
train_data <- data_mice[train_index,]
test_data <- data_mice[-train_index,]
train_outcome <- outcome[train_index]
test_outcome <- outcome[-train_index]

# for Lasso
train_variables_dummy <- variables_dummy[train_index, ]
test_variables_dummy <- variables_dummy[-train_index, ]

# for best subset (we will do this next)
train_variables <- variables[train_index, ]
test_variables <- variables[-train_index, ]

train_data_glmnet = data.frame(abst = train_outcome, train_variables_dummy)
test_data_glmnet = data.frame(abst = test_outcome, test_variables_dummy)

### First objective - Moderators

# We have to keep BA and Var with 0 penalty
# Lasso Regression
# ^2 generates all pairwise interactions
train_variables_dummy_df <- as.data.frame(train_variables_dummy)
train_variables_dummy_full_interactions <- model.matrix(~ .^2,
                                                        data = train_variables_dummy_df)
test_variables_dummy_df <- as.data.frame(test_variables_dummy)
test_variables_dummy_full_interactions <- model.matrix(~ .^2,
                                                       data = test_variables_dummy_df)
train_variables_dummy_include_names <- c(
  "Var1", "BA1", "age_ps", "sex_ps2", "inc2", "inc3",
  "inc4", "inc5", "edu_merged2", "edu_merged3",
  "raceBlack", "raceHispanic", "raceOther",
```

```
  "ftcd_score", "ftcd.5.mins1", "bdi_score_w00", "cpd_ps",
  "crv_total_pq1", "hedonsum_n_pq1", "hedonsum_y_pq1",
  "shaps_score_pq1", "otherdiag1", "antidepmed1",
  "mde_curr1", "NMR", "Only.Menthol1",
  "readiness",

  "BA1:mde_curr1",
  "BA1:age_ps", "BA1:sex_ps2",
  "BA1:raceBlack", "BA1:raceHispanic",
  "BA1:raceOther", "BA1:ftcd_score",
  "BA1:shaps_score_pq1","BA1:bdi_score_w00",
  "BA1:otherdiag1", "BA1:antidepmed1",
  "BA1:mde_curr1","BA1:NMR",
  "BA1:Only.Menthol1", "BA1:readiness", "BA1:cpd_ps",


  "Var1:BA1",
  "Var1:age_ps", "Var1:sex_ps2",
  "Var1:raceBlack", "Var1:raceHispanic",
  "Var1:raceOther", "Var1:ftcd_score", "Var1:cpd_ps",

   "inc2:edu_merged2", "inc2:edu_merged3",
  "inc3:edu_merged2", "inc3:edu_merged3",
  "inc4:edu_merged2", "inc4:edu_merged3",
  "inc5:edu_merged2", "inc5:edu_merged3",
  "antidepmed1:readiness", "Only.Menthol1:readiness",
  "mde_curr1:readiness", "ftcd.5.mins1:readiness",
  "bdi_score_w00:readiness", "Var1:shaps_score_pq1", "shaps_score_pq1:mde_curr1",
  "sex_ps2:ftcd_score", "raceBlack:ftcd_score",
  "raceHispanic:ftcd_score", "raceOther:ftcd_score",
  "age_ps:ftcd_score", "sex_ps2:Only.Menthol1",
  "raceBlack:Only.Menthol1", "raceHispanic:Only.Menthol1",
  "raceOther:Only.Menthol1",
  "inc2:Only.Menthol1", "inc3:Only.Menthol1",
  "inc4:Only.Menthol1", "inc5:Only.Menthol1",
  "edu_merged2:Only.Menthol1", "edu_merged3:Only.Menthol1",
  "sex_ps2:NMR", "age_ps:NMR", "cpd_ps:NMR",
  "NMR:readiness", "ftcd_score:NMR"
)

train_variables_dummy_include = train_variables_dummy_full_interactions[,train_variables_dummy_include_n

test_variables_dummy_include = test_variables_dummy_full_interactions[,train_variables_dummy_include_nam

# Initialize penalty factors to 1 for all variables
penalty_factors <- rep(1, ncol(train_variables_dummy_include))

# Identify columns corresponding exactly to "Var1" and "BA1" (not their interactions)
var1_col <- grep("^Var1$", colnames(train_variables_dummy_include))
ba1_col <- grep("^BA1$", colnames(train_variables_dummy_include))

penalty_factors[c(var1_col, ba1_col)] <- 0
names(penalty_factors) <- colnames(train_variables_dummy_include)
```

```r
set.seed(58)
Lasso_model <- cv.glmnet(as.matrix(train_variables_dummy_include), train_outcome,
                         penalty.factor = penalty_factors,
                         alpha = 1, family = "binomial")

# Extract coefficients at the optimal lambda (best_lambda)
best_lambda_Lasso <- Lasso_model$lambda.min
#remove intercept
optimal_coefs_Lasso <- as.numeric(coef(Lasso_model, s = best_lambda_Lasso)[-1])
coef_names_Lasso <- rownames(coef(Lasso_model, s = best_lambda_Lasso))[-1]

result_table_Lasso <- data.frame(
  variable = coef_names_Lasso,
  Coefficient = optimal_coefs_Lasso
) %>%
  filter(Coefficient != 0)

# Logistic regression

# Define outcome and predictor variables
outcome <- data_mice$abst
variable_names <- c("Var", "BA", "age_ps", "sex_ps", "inc", "edu_merged",
                    "race",
                    "ftcd_score", "ftcd.5.mins", "bdi_score_w00", "cpd_ps",
                    "crv_total_pq1", "hedonsum_n_pq1", "hedonsum_y_pq1",
                    "shaps_score_pq1", "otherdiag", "antidepmed", "mde_curr",
                    "NMR", "Only.Menthol", "readiness")

# Prepare the dataset
variables <- data_mice[, variable_names]
variables$Var <- factor(variables$Var)
variables$BA <- factor(variables$BA)

# Main effects for `Var` and `BA` (to be controlled in all models)
main_effects <- paste(c("Var", "BA", "age_ps", "sex_ps", "inc",
                        "edu_merged", "race",
                        "ftcd_score", "ftcd.5.mins", "bdi_score_w00",
                        "cpd_ps",
                        "crv_total_pq1", "hedonsum_n_pq1",
                        "hedonsum_y_pq1",
                        "shaps_score_pq1", "otherdiag", "antidepmed",
                        "mde_curr",
                        "NMR", "Only.Menthol", "readiness"),
                      collapse = " + ")

# Define the interaction terms we want to consider
interaction_terms <- paste(

  "BA:mde_curr + BA:age_ps + BA:sex_ps + BA:race + BA:ftcd_score",
  "BA:shaps_score_pq1 + BA:bdi_score_w00",
  "BA:otherdiag + BA:antidepmed",
  "BA:mde_curr + BA:NMR",
  "BA:Only.Menthol + BA:readiness + BA:cpd_ps + Var:BA + BA:cpd_ps",
```

```
  "Var:age_ps + Var:sex_ps + Var:race + Var:ftcd_score + Var:cpd_ps",
  "inc:edu_merged + antidepmed:readiness + Only.Menthol:readiness",
  "mde_curr:readiness + ftcd.5.mins:readiness + bdi_score_w00:readiness",
  "Var:shaps_score_pq1 + ",
  "shaps_score_pq1:mde_curr + sex_ps:ftcd_score + race:ftcd_score +
  age_ps:ftcd_score",
  "sex_ps:Only.Menthol + race:Only.Menthol + inc:Only.Menthol +
  edu_merged:Only.Menthol",
  "sex_ps:NMR + age_ps:NMR + cpd_ps:NMR + NMR:readiness + ftcd_score:NMR",
  sep = " + "
)

# Full formula for main effects and interactions
full_formula <- as.formula(paste("abst ~", main_effects,
                                 "+", interaction_terms))

# Define scope with `Var` and `BA` as forced terms in the main model
scope_list <- list(
  lower = as.formula("abst ~ Var + BA"),  # Minimal model with controlled terms
  upper = full_formula         # Full model with all main and interaction terms
)

# Fit logistic regression model with stepwise selection
set.seed(58)
logistic_model <- step(
  glm(formula = abst ~ Var + BA, data = train_data, family = "binomial"),
  scope = scope_list,
  direction = "both",
  trace=0
)
# Best subset selection

# Run best subset selection with regsubsets
best_subset_model <- regsubsets(
  y = train_outcome, x = train_variables_dummy_include,
  force.in = c("Var1", "BA1"), nbest = 1,
  nvmax = 100,  # Adjust this based on how many variables you want to consider
  method = "seq", force.out = NULL,
  really.big=T
)

# Summarize results
best_subset_summary <- summary(best_subset_model)
best_subset_summary = summary(best_subset_model)
cp_min = which.min(best_subset_summary$cp) # min is 10
best_subset_coefs = coef(best_subset_model, cp_min)
best_subset_names = names(coef(best_subset_model, cp_min))
# Create a table with coefficients from Logistic, Lasso, Best Subset Selection

# Logistic coefficients
stepwise_coefs <- coef(logistic_model)
stepwise_df <- data.frame(
  variable = names(stepwise_coefs),
```

```
  `Stepwise` = as.numeric(stepwise_coefs)
)

# Lasso coefficients
Lasso_df <- result_table_Lasso %>%
  rename(`Lasso` = Coefficient)

# Best subset coefficients
best_subset_df <- data.frame(
  variable = names(best_subset_coefs),
  `Best Subset` = as.numeric(best_subset_coefs)
)

# Merge all into one table based on variable names
combined_df <- full_join(Lasso_df, stepwise_df, by = "variable") %>%
  full_join(best_subset_df, by = "variable") %>%
  mutate(across(where(is.numeric), ~ round(.x, 4)))

# Remove the intercept row
combined_df <- combined_df[combined_df$variable != "(Intercept)", ]

# Replace NA values with an empty space
combined_df[is.na(combined_df)] <- " "

colnames(combined_df) <- gsub("Best.Subset", "Best Subset", colnames(combined_df))

# Change "variable" to "Variables" in the combined_df
colnames(combined_df)[colnames(combined_df) == "variable"] <- "Variables"
# Create a table with ORs from Logistic, Lasso, and Best Subset Selection

# Logistic coefficients (converted to OR)
stepwise_or <- exp(stepwise_coefs)  # Convert coefficients to Odds Ratios
stepwise_df <- data.frame(
  variable = names(stepwise_or),
  `Stepwise OR` = as.numeric(stepwise_or)
)

# Lasso coefficients (converted to OR)
Lasso_or_df <- result_table_Lasso %>%
  mutate(`Lasso OR` = exp(Coefficient)) %>%  # Convert coefficients to Odds Ratios
  dplyr::select(variable, `Lasso OR`)

# Best subset coefficients (converted to OR)
best_subset_or <- exp(best_subset_coefs)  # Convert coefficients to Odds Ratios
best_subset_df <- data.frame(
  variable = names(best_subset_or),
  `Best Subset OR` = as.numeric(best_subset_or)
)

# Merge all into one table based on variable names
combined_or_df <- full_join(Lasso_or_df, stepwise_df, by = "variable") %>%
  full_join(best_subset_df, by = "variable") %>%
  mutate(across(where(is.numeric), ~ round(.x, 4)))
```

```r
# Remove the intercept row
combined_or_df <- combined_or_df[combined_or_df$variable != "(Intercept)", ]

# Replace NA values with an empty space
combined_or_df[is.na(combined_or_df)] <- " "

# Rename columns
colnames(combined_or_df) <- gsub("Best.Subset", "Best Subset", colnames(combined_or_df))
colnames(combined_or_df)[colnames(combined_or_df) == "variable"] <- "Variables"
# Combined table for both coef and ORs
full_df = left_join(combined_df, combined_or_df, by = "Variables")

kable(full_df, row.names = FALSE,
      col.names = c("Variables","Lasso", "Stepwise", "Best Subset",
                    "Lasso OR", "Stepwise OR", "Best Subset OR"),
      caption = "Summary of Coefficients and Odds Ratios across 3 Sel
ection Methods (for Moderators)") %>%
  kable_styling(font_size = 8)
# Roc/Auc
predicted_prob_Lasso <- predict(Lasso_model,
                                newx = as.matrix(test_variables_dummy_include),
                                s = "lambda.min", type = "response")

# Convert predictions to numeric if needed (glmnet returns a matrix)
predicted_prob_Lasso <- as.numeric(predicted_prob_Lasso)

# Plot ROC and calculate AUC
# Generate ROC object
roc_obj <- roc(test_outcome, predicted_prob_Lasso)

# Convert the ROC object to a data frame for ggplot2
roc_data <- data.frame(
  Specificity = rev(roc_obj$specificities),
  Sensitivity = rev(roc_obj$sensitivities)
)

# Calculate the AUC
auc_value <- auc(roc_obj)

# Plot ROC curve with ggplot2
ROC_Lasso = ggplot(roc_data, aes(x = 1-Specificity, y = Sensitivity)) +
  geom_line(color = "slateblue3", size = 1) +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", color = "grey") +
  annotate("text", x = 0.8, y = 0.2, label = paste("AUC =",
  round(auc_value, 2)),
  size = 5, color = "Black") +
  labs(
    x = "1 - Specificity",
    y = "Sensitivity"
  ) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
```

```r
# Calibration Plots
num_cuts <- 10  # Number of bins for calibration

calib_data <- data.frame(
  prob = predicted_prob_Lasso,  # predicted probabilities
  # binning into `num_cuts` groups
  bin = cut(predicted_prob_Lasso, breaks = num_cuts),
  # observed values (abst outcome in test data)
  class = as.numeric(test_outcome)-1
)

calib_data <- calib_data %>%
  group_by(bin) %>%
  summarise(
    observed = mean(class),
    predicted = mean(prob),
    se = sqrt(observed * (1 - observed) / n())  # Standard error
  )

# Add Loess Fit for Flexible Calibration Line
loess_fit <- loess(observed ~ predicted, data = calib_data, span = 0.75)
calib_data$loess_pred <- predict(loess_fit, calib_data$predicted)

# Plot Calibration Curve with Error Bars
calib_error_bar_Lasso = ggplot(calib_data) +
  geom_abline(intercept = 0, slope = 1, color = "red") +
  geom_errorbar(aes(x = predicted, ymin = observed - 1.96 * se,
                    ymax = observed + 1.96 * se),
                colour="black", width=.01)+
  geom_point(aes(x = predicted, y = observed)) +
  labs(x = "Expected Probability of Smoking Abstinence",
       y = "Actual Smoking Abstinence") +
  theme_minimal()


# Plot Calibration Curve with Loess
calib_data <- calib_data %>%
  mutate(loess_ci_lower = loess_pred - 1.96 * sd(loess_pred),
         loess_ci_upper = loess_pred + 1.96 * sd(loess_pred))

calib_loess_Lasso = ggplot(calib_data, aes(x = predicted, y = observed)) +
  # Flexible calibration (Loess)
  geom_line(aes(y = loess_pred), color = "slateblue3", linetype = "dashed") +
  geom_ribbon(aes(ymin = loess_ci_lower, ymax = loess_ci_upper),
              alpha = 0.2, fill = "grey") +
  geom_abline(intercept = 0, slope = 1, color = "red") + # Perfect calibration line
  labs(x = "Predicted Probability of Smoking Abstinence",
       y = "Actual Smoking Abstinence") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
# ROC/AUC
predicted_prob_logistic <- predict(logistic_model,
                                   newdata = test_data,
```

```
                                type = "response")

# Convert predictions to numeric if needed (glmnet returns a matrix)
predicted_prob_logistic <- as.numeric(predicted_prob_logistic)

# Plot ROC and calculate AUC
# Generate ROC object
roc_obj <- roc(test_outcome, predicted_prob_logistic)

# Convert the ROC object to a data frame for ggplot2
roc_data <- data.frame(
  Specificity = rev(roc_obj$specificities),
  Sensitivity = rev(roc_obj$sensitivities)
)

# Calculate the AUC
auc_value <- auc(roc_obj)

# Plot ROC curve with ggplot2
ROC_step = ggplot(roc_data, aes(x = 1-Specificity, y = Sensitivity)) +
  geom_line(color = "slateblue3", size = 1) +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", color = "grey") +
  annotate("text", x = 0.8, y = 0.2, label = paste("AUC =",
                                                   round(auc_value, 2)),
           size = 5, color = "Black") + labs(
    x = "1 - Specificity",
    y = "Sensitivity"
  ) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
# Calibration Plots
num_cuts <- 10  # Number of bins for calibration

calib_data <- data.frame(
  prob = predicted_prob_logistic,  # predicted probabilities
  # binning into `num_cuts` groups
  bin = cut(predicted_prob_logistic, breaks = num_cuts),
  # observed values (abst outcome in test data)
  class = as.numeric(test_outcome)-1
)

calib_data <- calib_data %>%
  group_by(bin) %>%
  summarise(
    observed = mean(class),
    predicted = mean(prob),
    se = sqrt(observed * (1 - observed) / n())  # Standard error
  )

# Add Loess Fit for Flexible Calibration Line
loess_fit <- loess(observed ~ predicted, data = calib_data, span = 0.75)
calib_data$loess_pred <- predict(loess_fit, calib_data$predicted)
```

```r
# Plot Calibration Curve with Error Bars
calib_error_bar_step = ggplot(calib_data) +
  geom_abline(intercept = 0, slope = 1, color = "red") +
  geom_errorbar(aes(x = predicted, ymin = observed - 1.96 * se,
                    ymax = observed + 1.96 * se),
                colour="black", width=.01)+
  geom_point(aes(x = predicted, y = observed)) +
  labs(x = "Expected Probability of Smoking Abstinence",
       y = "Actual Smoking Abstinence") +
  theme_minimal()


# Plot Calibration Curve with Loess
calib_data <- calib_data %>%
  mutate(loess_ci_lower = loess_pred - 1.96 * sd(loess_pred),
         loess_ci_upper = loess_pred + 1.96 * sd(loess_pred))

calib_loess_step = ggplot(calib_data, aes(x = predicted, y = observed)) +
  # Flexible calibration (Loess)
  geom_line(aes(y = loess_pred), color = "slateblue3", linetype = "dashed") +
  geom_ribbon(aes(ymin = loess_ci_lower, ymax = loess_ci_upper),
              alpha = 0.2,
              fill = "grey") +
  geom_abline(intercept = 0, slope = 1, color = "red") +
  # Perfect calibration line
  labs(x = "Predicted Probability of Smoking Abstinence",
       y = "Actual Smoking Abstinence") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
# ROC/AUC
predict_best_subset <- function(test_data = test_variables_dummy_include,
                                best_subset_coefs) {
  intercept <- best_subset_coefs[1]
  selected_vars <- names(best_subset_coefs)[-1]

  test_subset <- test_data[, selected_vars, drop = FALSE]

  # Calculate predictions by multiplying test data with coefficients
  # Matrix multiplication for predictors + intercept
  predictions <- intercept + as.matrix(test_subset) %*% best_subset_coefs[selected_vars]

  return(predictions)
}

predicted_prob_best_subset <- predict_best_subset(test_data = test_variables_dummy_include,
                                  best_subset_coefs)
# Convert predictions to numeric
predicted_prob_best_subset <- as.numeric(predicted_prob_best_subset) -1

# Plot ROC and calculate AUC
# Generate ROC object
roc_obj <- roc(test_outcome, predicted_prob_best_subset)
```

```r
# Convert the ROC object to a data frame for ggplot2
roc_data <- data.frame(
  Specificity = rev(roc_obj$specificities),
  Sensitivity = rev(roc_obj$sensitivities)
)

# Calculate the AUC
auc_value <- auc(roc_obj)

# Plot ROC curve with ggplot2
ROC_best_subset = ggplot(roc_data, aes(x = 1-Specificity, y = Sensitivity)) +
  geom_line(color = "slateblue3", size = 1) +
  geom_abline(slope = 1, intercept = 0, linetype = "dashed", color = "grey") +
  annotate("text", x = 0.8, y = 0.2,
           label = paste("AUC =", round(auc_value, 2)), size = 5, color = "Black") +
  labs(
    x = "1 - Specificity",
    y = "Sensitivity"
  ) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5))
# Calibration plots
num_cuts <- 10  # Number of bins for calibration

calib_data <- data.frame(
  prob = predicted_prob_best_subset,  # predicted probabilities
  # binning into `num_cuts` groups
  bin = cut(predicted_prob_logistic, breaks = num_cuts),
  # observed values (abst outcome in test data)
  class = as.numeric(test_outcome)-1
)

calib_data <- calib_data %>%
  group_by(bin) %>%
  summarise(
    observed = mean(class),
    predicted = mean(prob),
    se = sqrt(observed * (1 - observed) / n())  # Standard error
  )

# Add Loess Fit for Flexible Calibration Line
loess_fit <- loess(observed ~ predicted, data = calib_data, span = 0.75)
calib_data$loess_pred <- predict(loess_fit, calib_data$predicted)

# Plot Calibration Curve with Error Bars
calib_error_bar_bss = ggplot(calib_data) +
  geom_abline(intercept = 0, slope = 1, color = "red") +
  geom_errorbar(aes(x = predicted, ymin = observed - 1.96 * se,
                    ymax = observed + 1.96 * se),
                colour="black", width=.01)+
  geom_point(aes(x = predicted, y = observed)) +
  labs(x = "Expected Probability of Smoking Abstinence",
```

```
       y = "Actual Smoking Abstinence") +
  theme_minimal()


# Plot Calibration Curve with Loess
calib_data <- calib_data %>%
  mutate(loess_ci_lower = loess_pred - 1.96 * sd(loess_pred),
         loess_ci_upper = loess_pred + 1.96 * sd(loess_pred))

calib_loess_bss = ggplot(calib_data, aes(x = predicted, y = observed)) +
  # Flexible calibration (Loess)
  geom_line(aes(y = loess_pred), color = "slateblue3", linetype = "dashed") +
  geom_ribbon(aes(ymin = loess_ci_lower, ymax = loess_ci_upper), alpha = 0.2, fill = "grey") +
  geom_abline(intercept = 0, slope = 1, color = "red") +  # Perfect calibration line
  labs(x = "Predicted Probability of Smoking Abstinence",
       y = "Actual Smoking Abstinence") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))

# Create the 3x3 plot "matrix"
plots_step = arrangeGrob(
  calib_error_bar_step, calib_loess_step, ROC_step,
  ncol = 3,
  left = textGrob("Stepwise Selection", rot = 90,
                  gp = gpar(fontface = "bold", fontsize = 18)
))

plots_Lasso = arrangeGrob(
  calib_error_bar_Lasso, calib_loess_Lasso, ROC_Lasso,
  ncol = 3,
  left = textGrob("Lasso Regression", rot = 90,
                  gp = gpar(fontface = "bold", fontsize = 18)
))

plots_bss = arrangeGrob(
  calib_error_bar_bss, calib_loess_bss, ROC_best_subset,
  ncol = 3,
  left = textGrob("Best Subset Selection", rot = 90,
                  gp = gpar(fontface = "bold", fontsize = 18)
))

# Bold the main title
main_title <- textGrob(
  "Calibration Plots with Error Bars and LOESS - ROC Curves (for the 3 Selection Models)",
  gp = gpar(fontface = "bold", fontsize = 20)
)

# Arrange everything with the bold title
grid.arrange(
  plots_Lasso,
  plots_step,
  plots_bss,
  nrow = 3,
```

```
  top = main_title
)
### Second objective - Predictors

predictor_names <- c("Var", "BA", "age_ps", "sex_ps", "inc", "edu_merged", "race",
                     "ftcd_score", "ftcd.5.mins", "bdi_score_w00", "cpd_ps",
                     "crv_total_pq1", "hedonsum_n_pq1", "hedonsum_y_pq1",
                     "shaps_score_pq1", "otherdiag", "antidepmed", "mde_curr",
                     "NMR", "Only.Menthol", "readiness")

predictors <- data_mice[, predictor_names]
# for Lasso (to break down factors with >2 levels)
predictors_dummy <- model.matrix(~ 0 + ., data = predictors)
# remove the extra reference group
predictors_dummy <- predictors_dummy[, -which(colnames(predictors_dummy) =="Var0")]


# Lasso Regression
# To identify the potential interaction terms for moderator effects
train_predictors_dummy_include_names <- c(
  "Var1", "BA1", "age_ps", "sex_ps2", "inc2", "inc3",
  "inc4", "inc5", "edu_merged2", "edu_merged3",
  "raceBlack", "raceHispanic", "raceOther",
  "ftcd_score", "ftcd.5.mins1", "bdi_score_w00", "cpd_ps",
  "crv_total_pq1", "hedonsum_n_pq1", "hedonsum_y_pq1",
  "shaps_score_pq1", "otherdiag1", "antidepmed1",
  "mde_curr1", "NMR", "Only.Menthol1",
  "readiness"
)

train_predictors_dummy <- predictors_dummy[train_index, ]
test_predictors_dummy <- predictors_dummy[-train_index, ]

train_predictors_dummy_include =
  train_predictors_dummy[,train_predictors_dummy_include_names]
test_predictors_dummy_include =
  test_predictors_dummy[,train_predictors_dummy_include_names]

# Initialize penalty factors to 1 for all variables
penalty_factors <- rep(1, ncol(train_predictors_dummy_include))

# Identify columns corresponding exactly to "Var1" and "BA1" (not their interactions)
var1_col <- grep("^Var1$", colnames(train_predictors_dummy_include))
ba1_col <- grep("^BA1$", colnames(train_predictors_dummy_include))

penalty_factors[c(var1_col, ba1_col)] <- 0
names(penalty_factors) <- colnames(train_predictors_dummy_include)

set.seed(58)
Lasso_model <- cv.glmnet(as.matrix(train_predictors_dummy_include), train_outcome,
                         penalty.factor = penalty_factors,
                         alpha = 1, family = "binomial")
```

```r
# Extract coefficients at the optimal lambda (best_lambda)
best_lambda_Lasso <- Lasso_model$lambda.min
#remove intercept
optimal_coefs_Lasso <- as.numeric(coef(Lasso_model, s = best_lambda_Lasso)[-1])
coef_names_Lasso <- rownames(coef(Lasso_model, s = best_lambda_Lasso))[-1]

pred_result_table_Lasso <- data.frame(
  variable = coef_names_Lasso,
  Coefficient = optimal_coefs_Lasso
) %>%
  filter(Coefficient != 0)
# Logistic Regression

# Full formula for main effects and interactions
full_formula <- as.formula(paste("abst ~", main_effects))

# Define scope with `Var` and `BA` as forced terms in the main model
scope_list <- list(
  lower = as.formula("abst ~ Var + BA"),  # Minimal model with controlled terms
  upper = full_formula                    # Full model with all main terms
)

# Fit logistic regression model with stepwise selection
set.seed(58)
predictor_logistic_model <- step(
  glm(formula = abst ~ Var + BA, data = train_data, family = "binomial"),
  scope = scope_list,
  direction = "both",
  trace = 0
)
# Best subset selection

# Run best subset selection with regsubsets
best_subset_model_pred <- regsubsets(
  y = train_outcome, x = train_predictors_dummy_include,
  force.in = c("Var1", "BA1"), nbest = 1,
  nvmax = 100,  # Adjust this based on how many variables you want to consider
  method = "seq", force.out = NULL,
  really.big=T
)

# Summarize results
best_subset_summary <- summary(best_subset_model_pred)
best_subset_summary = summary(best_subset_model_pred)
cp_min = which.min(best_subset_summary$cp) # min is 10
pred_best_subset_coefs = coef(best_subset_model_pred, cp_min)
best_subset_names_pred = names(coef(best_subset_model_pred, cp_min))
# Create a table with coefficients from Logistic, Lasso, Best Subset Selection

# Stepwise coefficients
stepwise_coefs <- coef(predictor_logistic_model)
stepwise_df <- data.frame(
  variable = names(stepwise_coefs),
```

```
  `Stepwise` = as.numeric(stepwise_coefs)
)

# Lasso coefficients
Lasso_df <- pred_result_table_Lasso %>%
  rename(`Lasso` = Coefficient)

# Best subset coefficients
best_subset_df <- data.frame(
  variable = names(pred_best_subset_coefs),
  `Best Subset` = as.numeric(pred_best_subset_coefs)
)

# Merge all into one table based on variable names
combined_df <- full_join(Lasso_df, stepwise_df, by = "variable") %>%
  full_join(best_subset_df, by = "variable") %>%
  mutate(across(where(is.numeric), ~ round(.x, 4)))

# Remove the intercept row
combined_df <- combined_df[combined_df$variable != "(Intercept)", ]

# Replace NA values with an empty space
combined_df[is.na(combined_df)] <- " "

colnames(combined_df) <- gsub("Best.Subset", "Best Subset", colnames(combined_df))

# Change "variable" to "Variables" in the combined_df
colnames(combined_df)[colnames(combined_df) == "variable"] <- "Variables"
# Create a table with OR from the three methods
# Convert Stepwise coefficients to Odds Ratios
stepwise_coefs <- coef(predictor_logistic_model)
stepwise_or <- exp(stepwise_coefs)
stepwise_df <- data.frame(
  Variables = names(stepwise_or),
  `Stepwise OR` = as.numeric(stepwise_or)
)

# Convert Lasso coefficients to Odds Ratios
Lasso_or_df <- pred_result_table_Lasso %>%
  mutate(`Lasso OR` = exp(Coefficient)) %>%
  rename(Variables = variable)  %>%
  dplyr::select(Variables, `Lasso OR`)

# Convert Best subset coefficients to Odds Ratios
best_subset_or <- exp(pred_best_subset_coefs)
best_subset_df <- data.frame(
  Variables = names(best_subset_or),
  `Best Subset OR` = as.numeric(best_subset_or)
)

# Merge all into one table based on variable names
combined_or_df <- full_join(Lasso_or_df, stepwise_df, by = "Variables") %>%
  full_join(best_subset_df, by = "Variables") %>%
```

```r
  mutate(across(where(is.numeric), ~ round(.x, 4)))

# Remove the intercept row
combined_or_df <- combined_or_df[combined_or_df$Variables != "(Intercept)", ]

# Replace NA values with an empty space
combined_or_df[is.na(combined_or_df)] <- " "
# Combined table for both coef and ORs
full_df = left_join(combined_df, combined_or_df, by = "Variables")
kable(full_df, row.names = FALSE,
      col.names = c("Variables","Lasso", "Stepwise", "Best Subset",
                    "Lasso OR", "Stepwise OR", "Best Subset OR"),
      caption = "Summary of Coefficients and Odds Ratios across 3
Selection Methods (for Predictors)")
```