Predicting the Need for Tracheostomy in Infants with Severe Bronchopulmonary Dysplasia

Kerry Ye

2023-11-06

Introduction

In collaboration with Dr. Chris Schmid, this project aims to develop statistical models for predicting the composite outcome of tracheostomy or death, considering the indication criteria and timing of tracheostomy placement. Bronchopulmonary Dysplasia (sBPD), a common sequela of prematurity, severely impacts the health of infants. It affects 10,000-15,000 infants annually and is largely influenced by individual susceptibility. Current research suggests that early tracheostomy placement may benefit those with severe Bronchopulmonary Dysplasia (sBPD). However, the indication criteria and optimal timing for tracheostomy placement in neonates with sBPD remain unclear. Tracheostomy provides a stable airway, improves ventilator synchrony, and is associated with improved sBPD outcomes when performed within the first 4 months of age. Conversely, tracheostomy also carries significant risks, including an increased likelihood of death and higher infection rates. Therefore, accurately predicting the eventual need for tracheostomy placement before discharge is essential.

Data Preprocessing

The underlying recruitment population were drawn from the BPD Collaborative Registry, a multi-center consortium of BPD programs located in the United States and Sweden in order to enhance the care of children with severe forms of BPD. The registry includes infants whose gestational age is less than 32 weeks and who have severe sBPD. In the registry, standard demographic and clinical data are collected at four time points: birth, 36 weeks PMA, 44 weeks PMA and discharge. For this study, we collected the data from patients with BPD and complete growth data between January 1 and July 19, 2021.

The raw dataset initially contained 999 observations and 30 variables. After removing 3 duplicate entries, 996 individuals remained. We first converted various categorical variables into factors, reflecting their different levels. Notably, 10 individuals lacked corresponding center information; however, their **record_id** prefix (1xxxxxx) indicated they were from **center** 1, so we manually imputed their center values as 1. One individual, with **record_id:2000824**, had four duplicate rows; we retained only one row and removed the duplicates. The **mat_race** column was removed from the data due to coding inconsistencies with the codebook. Additionally, one individual's **record_id** was coded as 21000001, and their **center** code as 21. We corrected their **record_id** to 1000001 and **center** to 1, as their original record_id and center were incorrectly recorded for center 21 instead of center 1. Finally, we created the composite outcome, **outcome**, where individuals who had either undergone a tracheostomy or died, as well as those who had undergone both, were coded as 1, while those who had neither undergone tracheostomy nor died were coded as 0. This composite outcome can provide a more comprehensive understanding of the overall burden and risk associated with tracheostomy and mortality, thereby reflecting the overall severity or progression of Bronchopulmonary Dysplasia.

Exploratory Data Analysis

A summary table is important because it provides crucial insights into the data's structure, helps identify potential issues, and informs the choice of appropriate statistical models and methods for analysis. In this

section, we first created the following summary statistics table for the dataset, focusing on the two composite outcomes: 1 and 0.

Characteristic	**N**	**0**, N = 811[1]	**1**, N = 183[1]
center	994		
1		31 (3.8%)	35 (19%)
2		545 (67%)	84 (46%)
3		55 (6.8%)	2 (1.1%)
4		47 (5.8%)	12 (6.6%)
5		33 (4.1%)	7 (3.8%)
7		31 (3.8%)	1 (0.5%)
12		28 (3.5%)	41 (22%)
16		37 (4.6%)	1 (0.5%)
20		4 (0.5%)	0 (0%)
mat_ethn	937		
1		64 (8.3%)	10 (5.9%)
2		703 (92%)	160 (94%)
bw	994		
Mean (SD)		817 (285)	756 (340)
ga	994		
Mean (SD)		26 (2)	26 (2)
blength	916		
Mean (SD)		33 (4)	32 (4)
birth_hc	917		
Mean (SD)		23.25 (2.65)	22.88 (3.29)
del_method	991		
1		245 (30%)	39 (21%)
2		564 (70%)	143 (79%)
prenat_ster	959	679 (86%)	154 (92%)
com_prenat_ster	801	499 (76%)	109 (76%)
mat_chorio	932	132 (17%)	28 (17%)
gender	990		
Female		334 (41%)	73 (40%)
Male		473 (59%)	110 (60%)
sga	979		
Not SGA		658 (82%)	118 (66%)
SGA		142 (18%)	61 (34%)
any_surf	562	374 (81%)	87 (88%)
weight_today.36	902		
Mean (SD)		2,142 (393)	1,981 (507)
ventilation_support_level.36	964		
0		109 (14%)	7 (4.3%)
1		553 (69%)	36 (22%)
2		140 (17%)	119 (73%)
inspired_oxygen.36	902		
Mean (SD)		0.31 (0.12)	0.49 (0.21)
p_delta.36	866		
Mean (SD)		4 (8)	16 (12)
peep_cm_h2o_modified.36	877		
Mean (SD)		6 (3)	7 (3)
med_ph.36	964		
0		770 (96%)	129 (80%)
1		32 (4.0%)	33 (20%)
weight_today.44	550		

Mean (SD)		3,695 (643)	3,473 (782)
ventilation_support_level_modified.44	572		
0		261 (60%)	8 (6.0%)
1		124 (28%)	22 (16%)
2		53 (12%)	104 (78%)
inspired_oxygen.44	548		
Mean (SD)		0.31 (0.11)	0.45 (0.20)
p_delta.44	548		
Mean (SD)		4 (11)	22 (16)
peep_cm_h2o_modified.44	550		
Mean (SD)		3 (4)	9 (3)
med_ph.44	572		
0		405 (92%)	68 (51%)
1		33 (7.5%)	66 (49%)
hosp_dc_ga	871		
Mean (SD)		49 (24)	73 (30)
Trach	994		
0		811 (100%)	37 (20%)
1		0 (0%)	146 (80%)
Death	994	0 (0%)	54 (30%)

[1] n (%)

From the summary table, we observe that there are a total of 811 infant observations where neither tracheotomy nor death occurred, while 183 observations involve either tracheotomy, death, or both. The study records data from nine centers, with Center 2 contributing the most observations, accounting for 630 individuals. Some centers have notably fewer observations; for example, Center 20 has only 4 recorded observations. Regarding birth variables, it is observed that mothers of non-Hispanic or Latino ethnicity have a higher proportion of infants undergoing tracheotomy or experiencing death. Infants who underwent tracheotomy or died tend to have lower birth weights, weights at 36 weeks, and weights at 44 weeks compared to those who did not. Additionally, males are more likely to undergo tracheotomy or experience death compared to females. Obstetrical gestational age, birth length, and head circumference appear comparable between groups with composite outcomes of 1 and 0. It is also noted that more individuals in invasive positive pressure ventilation at 36 and 44 weeks underwent tracheotomy or died compared to those in non-invasive positive pressure or without respiratory support. This observation aligns with the expectation that more severe symptoms require more invasive respiratory support. Furthermore, more cases of Cesarean section deliveries, as opposed to vaginal births, are associated with tracheotomy or death. This is consistent with the understanding that earlier deliveries, often necessitating Cesarean sections, are more likely in severe cases of Bronchopulmonary Dysplasia, a common complication in premature births.

Missing data can diminish the statistical power of analyses and may potentially yield less precise and robust estimates in subsequent modeling. To address this, an initial examination of the number and percentage of missing values across the entire dataset is presented in the table below.

Table 2: Missingness in Tracheostomy Dataset

variable	n_miss	pct_miss
inspired_oxygen.44	448	44.9799197
p_delta.44	448	44.9799197
weight_today.44	446	44.7791165
peep_cm_h2o_modified.44	446	44.7791165
any_surf	433	43.4738956
ventilation_support_level_modified.44	424	42.5702811
med_ph.44	424	42.5702811

variable	n_miss	pct_miss
com_prenat_ster	193	19.3775100
p_delta.36	128	12.8514056
hosp_dc_ga	124	12.4497992
peep_cm_h2o_modified.36	117	11.7469880
weight_today.36	92	9.2369478
inspired_oxygen.36	92	9.2369478
blength	78	7.8313253
birth_hc	77	7.7309237
mat_chorio	62	6.2248996
mat_ethn	57	5.7228916
prenat_ster	35	3.5140562
ventilation_support_level.36	30	3.0120482
med_ph.36	30	3.0120482
sga	15	1.5060241
gender	4	0.4016064
del_method	3	0.3012048
Death	2	0.2008032
outcome	2	0.2008032

Examining the 'Missingness in Tracheostomy Dataset' table reveals that the five variables with the highest percentages of missing data are **inspired_oxygen.44**, **p_delta.44**, **weight_today.44**, **peep_cm_h2o_modified.44**, and **any_surf**, with missingness percentages ranging from 44.98% to 43.47%. Relying on these columns for analysis could skew results due to their substantial amount of missing data. Notably, **inspired_oxygen.44** and **p_delta.44** have the highest missing data percentage, approximately 44.98%, each with 424 missing values. This table also indicates that variables recorded at 44 weeks exhibit a much higher percentage of missingness compared to those recorded at 36 weeks and at birth.

We further examine the missing values in the data by recording the missingness percentage across nine different centers in order to account for the missingness difference across different centers.

Table 3: Missing Percentage % Across Different Centers

	center1	center2	center3	center4	center7	center12	center16	center20
mat_ethn	37.88	0.00	3.51	3.33	84.38	0.00	0.00	25
bw	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
ga	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
blength	13.64	3.81	0.00	1.67	18.75	53.62	0.00	0
birth_hc	13.64	4.60	0.00	3.33	18.75	44.93	0.00	0
del_method	1.52	0.00	0.00	0.00	0.00	2.90	0.00	0
prenat_ster	6.06	0.16	5.26	1.67	6.25	33.33	2.63	0
com_prenat_ster	24.24	16.67	17.54	28.33	28.12	37.68	13.16	50
mat_chorio	45.45	0.00	40.35	1.67	3.12	0.00	13.16	25
gender	1.52	0.32	1.75	0.00	0.00	0.00	0.00	0
sga	1.52	1.59	5.26	1.67	0.00	0.00	0.00	0
any_surf	39.39	46.83	1.75	73.33	81.25	17.39	57.89	50
weight_today.36	25.76	5.71	5.26	10.00	3.12	42.03	0.00	0
ventilation_support_level.36	1.52	1.43	1.75	0.00	0.00	27.54	0.00	0
inspired_oxygen.36	28.79	5.71	3.51	5.00	3.12	42.03	0.00	50
p_delta.36	30.30	6.19	12.28	25.00	3.12	46.38	0.00	25
peep_cm_h2o_modified.36	36.36	6.51	19.30	10.00	3.12	49.28	0.00	0
med_ph.36	1.52	1.43	1.75	0.00	0.00	27.54	0.00	0
weight_today.44	10.61	40.00	66.67	100.00	65.62	37.68	86.84	0
ventilation_support_level_modified.44	6.06	37.94	64.91	100.00	62.50	31.88	86.84	0

	center1	center2	center3	center4	center7	center12	center16	center20
inspired_oxygen.44	13.64	40.16	66.67	100.00	65.62	36.23	86.84	0
p_delta.44	10.61	39.37	66.67	100.00	68.75	37.68	86.84	0
peep_cm_h2o_modified.44	10.61	39.68	68.42	100.00	62.50	40.58	86.84	0
med_ph.44	6.06	37.94	64.91	100.00	62.50	31.88	86.84	0
hosp_dc_ga	96.97	0.00	0.00	100.00	0.00	0.00	0.00	0
Trach	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0
Death	0.00	0.16	0.00	1.67	0.00	0.00	0.00	0
outcome	0.00	0.16	0.00	1.67	0.00	0.00	0.00	0

From the table 'Missing Percentage Across Different Centers,' it is evident that overall, all centers have relatively low missingness in the mother and child information recorded at birth. Notably, no center has missing values for birth weight and obstetrical gestational age. However, Center 4 and Center 1 exhibit a significant proportion of missingness in recording maternal ethnicity, and Center 12 has a considerable proportion of missingness in infants' birth length and head circumference. Examining the variables associated with 36 weeks, Center 16 has no missingness, while Center 12 accounts for half of the missingness. In terms of the variables associated with 44 weeks, almost all the centers exhibit around or more than half of the data missing, except for Center 1 and Center 20, which have a smaller amount of missingness. However, given that Center 20 has only 4 observations, its data at 44 weeks may not be reliable. Center 4 shows the highest percentage of missingness for variables associated with 44 weeks, with 100% missingness, indicating that this center has never recorded information for infants at 44 weeks. Center 16 has over 86% missingness in recording 44-week data. Researchers should be cautious of this variation in missingness across different centers when developing related prediction models.

From the patient's perspective, the patient with the highest percentage of missing data is identified by record_id:2000081 from Center 2, with 48.27% missingness and 14 missing values. The patients with the second and third highest missingness, having record_ids 1000049 and 2000586, both exhibit 44.83% in missing values, each with 13 missing values.

Model Development

To address missing data, we use the mice() function in the 'mice' package to create five imputed datasets with filled-in missing values. We then designate 70% of each imputed dataset as the training data and 30% as the testing set. We use the training data to train the two proposed models and evaluate the model performance using the testing sets.

We employ two variable selection methods to identify the appropriate variables for inclusion in the proposed prediction models. The first method is **Lasso**. Lasso regression is an analysis method that performs both variable selection and regularization, enhancing the prediction accuracy and interpretability of the statistical model. It incorporates a penalty term, $\lambda \sum_{j=1}^{p} |\beta_j|$, applied to different parameters of the model, effectively shrinking some coefficients and setting others to zero. Another variable selection method we use is **Best Subset**. This approach involves an exhaustive search through all possible combinations of predictors. For each possible subset, a separate logistic regression model is fitted.

First Step

The first step in model development involves constructing the lasso model and the best subset logistic model for each imputed dataset. We then employ cross-validation to select the optimal model for each dataset, followed by averaging the results across all five datasets

The Lasso Model: The first predictive model we propose is the lasso regression model, used for both feature selection and modeling purposes. For each imputed dataset, we fit the lasso regression model. A 10-fold cross-validation procedure is then implemented on the lasso regression to determine the best lambda value that minimizes both the sum of squared likelihoods and the penalty term. The coefficients obtained from fitting the lasso model with the best lambda value for each imputed dataset are then averaged to find the final lasso coefficients

The best subset Logistic Model: The second predictive model we propose is the logistic regression model, selected by the best subset selection method. For each imputed dataset, we fit a logistic regression model using the best subset selection method. A 10-fold cross-validation procedure is then implemented on the logistic regression models during the best subset selection process to determine the best k subset variables to be included in the optimal model. The coefficients obtained from fitting the logistic model with the best subsets for each imputed dataset are then averaged to find the final logistic model coefficients.

Table 4: Regression Coefficient Comparisons

	coefs_lasso	coefs_subset_logistic
(Intercept)	-4.558	-3.904
center2	-1.161	-1.622
center3	-1.976	-4.876
center4	-0.213	-0.649
center5	-0.121	-0.340
center7	-1.417	-1.833
center12	0.958	0.902
center16	-1.641	-2.265
center20	-0.923	-2.826
mat_ethn2	0.312	0.428
bw	0.000	0.000
ga	-0.101	-0.176
blength	0.064	0.109
birth_hc	0.020	0.000
del_method2	0.721	1.026
prenat_sterYes	0.388	0.310
com_prenat_sterYes	0.369	0.500
mat_chorioYes	0.000	0.000
genderMale	0.017	0.000
sgaSGA	0.093	0.000
any_surfYes	0.054	0.000
weight_today.36	0.000	0.000
ventilation_support_level.361	-0.246	0.000
ventilation_support_level.362	0.890	1.367
inspired_oxygen.36	1.056	0.849
p_delta.36	0.001	0.000
peep_cm_h2o_modified.36	0.000	-0.060
med_ph.361	-0.024	-0.151
weight_today.44	0.000	-0.001
ventilation_support_level_modified.441	-0.752	-1.448
ventilation_support_level_modified.442	0.002	-0.691
inspired_oxygen.44	1.840	2.020
p_delta.44	0.000	0.000
peep_cm_h2o_modified.44	0.262	0.404
med_ph.441	1.029	1.177
hosp_dc_ga	0.025	0.034

From the above table, we observe that when comparing the coefficients between lasso regression and Best Subset Logistic Regression, after averaging all five imputed sets, the magnitudes of the coefficients in logistic regression are generally larger than those in lasso regression. This discrepancy arises because the regularization term in lasso penalizes large coefficients. Lasso regression shrinks some of the coefficients to zero, and the logistic regression, selected by best subset, only shows the coefficients that are included at least once during the best subset selection process. We can see that some variables, such as **bw**, **mat_chorioYes**,

weight_today.36, and **p_delta.44**, have zero coefficients and are not selected across the five imputed sets in both the lasso and best subset processes. Overall, it is observable that the best subset method is more aggressive than lasso, as it results in more variables not being selected compared to lasso.

Second Step

The second step of the variable selection process involves examining the frequency table of zero coefficients across the five imputed datasets and removing the variables that are frequently not selected by the variable selection method.

The Lasso Model: We removed variables that were shrunk to zero coefficients by the lasso for all five imputed datasets. Additionally, we removed variables that had coefficients nearly zero. Although these variables did not shrink to exactly zero, their impact on the outcome is minimal. After the exclusion of these variables, we established our Final Lasso Model for the five imputed sets.

The best subset Logistic Model: We removed variables that were not selected by the best subset procedure in all five imputed datasets. Similarly, we removed variables if their coefficients were nearly zero. Despite these variables not being exactly zero, their minimal effect on the outcome led to their exclusion. Following the removal of these variables, we established our Final Best Subset Logistic Model for the five imputed sets.

Table 5: Frequency of Zero Coefficients Across Five Imputed Sets

	lasso	subset
(Intercept)	0	0
center2	0	1
center3	0	1
center4	3	3
center5	3	3
center7	0	2
center12	0	2
center16	0	2
center20	0	3
mat_ethn2	2	3
bw	5	5
ga	2	3
blength	2	3
birth_hc	4	5
del_method2	0	1
prenat_sterYes	0	4
com_prenat_sterYes	0	2
mat_chorioYes	5	5
genderMale	2	5
sgaSGA	3	5
any_surfYes	4	5
weight_today.36	3	5
ventilation_support_level.361	2	5
ventilation_support_level.362	0	0
inspired_oxygen.36	0	3
p_delta.36	4	5
peep_cm_h2o_modified.36	4	3
med_ph.361	3	4
weight_today.44	0	1
ventilation_support_level_modified.441	1	2
ventilation_support_level_modified.442	2	3
inspired_oxygen.44	0	1

	lasso	subset
p_delta.44	4	5
peep_cm_h2o_modified.44	0	0
med_ph.441	0	1
hosp_dc_ga	0	0

Model Evaluation

We evaluate the performance of the lasso and best subset logistic models based on both discrimination and calibration. Discrimination refers to a model's ability to correctly differentiate between positive and negative outcomes. In this context, we use **Sensitivity**, **Specificity**, and the **Area Under the Curve (AUC)** from the ROC-AUC plot to assess the model's ability to distinguish between composite outcomes of 1 and 0. Calibration, meanwhile, measures how well the predicted probabilities of an event match the observed outcomes. We assess the performance of these two models using the **Brier Score**.

For each model, we first manually calculated the predicted outcome values for each imputed testing set by using the coefficients calculated from the trained models, multiplied by the predictor values from the imputed testing set. Then, we used the formula $\frac{1}{1+e^{-(\beta_0+\beta_1+\ldots)}}$ to calculate the predicted probabilities. Finally, we transformed these probabilities into binary outcomes using the optimal threshold.

Table 6: Model Evaluation Measures

Measures	Lasso	SubsetLogistic
AUC	0.900	0.884
Sensitivity	0.855	0.766
Specificity	0.796	0.852
Threshold	0.532	0.087
BrierScore	0.151	0.099

we calculated evaluation metrics for each imputed testing set and average them across the five imputed testing sets.

To assess discrimination, we first use the sensitivity, defined as:

$$\text{Sensitivity} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

and specificity, defined as:

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positive}}$$

The Area Under the Curve (AUC) measures the model's ability to distinguish between positive and negative outcomes. An AUC score of 0.5 implies no discriminative power, equivalent to random guessing, while a score of 1 denotes perfect discrimination.

Among the two models, they share similar AUC scores. The lasso model has a slightly higher AUC score compared to the best subset logistic model. With an AUC score of 0.900, there is a 90.0% chance that this model will be able to distinguish between patients with and without tracheotomy and death. Meanwhile, the best subset logistic model has an AUC score of 0.884, indicating an 88.4% chance of correctly distinguishing between these patient outcomes.

The lasso model exhibits higher sensitivity compared to the best subset logistic model, meaning that the proportion of patients with active tracheotomy and death is correctly classified more often in the lasso model than in the best subset logistic model. Conversely, the best subset logistic model demonstrates higher specificity than the lasso model, indicating a higher proportion of correct classifications for patients without tracheotomy and death.

The best subset logistic model has a much lower threshold compared to the lasso model, with a threshold of 0.087. This suggests that the model is set to be more sensitive to detecting positive cases, potentially at the expense of increased false positives. Meanwhile, the threshold in the lasso model is relatively balanced between detecting positive and negative cases.

The Brier score is a metric used to evaluate the accuracy of probabilistic predictions. It represents the average squared difference between the predicted probabilities and the actual outcomes.The mathematical formula for the Brier score is:

$$\text{Brier score} = \frac{1}{N}\sum_{i=1}^{N}(f_i - o_i)^2,$$

where f_i is the predicted probability for the i-th event, o_i represents the observed outcome of the i-th event (0, 1), and N is the total number of prediction. The smaller the Brier score, the better the model's performance. A score of 0 means perfect predictions, while a score of 1 is the worst. The best subset logistic model has a smaller Brier score, suggesting that it provides more accurate predictions on patients' tracheotomy and death than the lasso model.

Conclusion

Overall, these two models share similarly high performance in predicting the need for tracheostomy in infants with severe bronchopulmonary dysplasia. The lasso model has a higher AUC score, indicating better performance in distinguishing between the two outcomes. Conversely, the best subset logistic model has a lower Brier score, suggesting more accurate probabilistic predictions. The lasso model is more precise in predicting positive cases, while the best subset logistic model excels in predicting negative cases. The lasso model may be more appropriate in healthcare, correctly identifying true positives is often more crucial due to the potential consequences of missing a serious condition. Early detection can lead to timely interventions, reducing the risk of complications and improving patient outcomes. However, the choice of the best predictive model should depend on several factors related to the specific needs of researchers and practitioners.

This project has several limitations. First, the data contains a substantial amount of missing values, which were filled using a multiple imputation process. While this approach helps in handling missing data, it may introduce biases or inaccuracies. The imputed values are estimates and may not perfectly represent the true missing values, potentially affecting the robustness of the findings. Second, we did not consider any interaction terms in our models. Interaction terms are important because they reveal how the effect of one variable depends on the level of another variable. By omitting these, we may have overlooked complex relationships between variables, leading to a potentially oversimplified understanding of the factors influencing the outcomes. Finally, we did not use mixed effects models, which allow for random effects to account for variability between different centers. This is important because individuals' measurements might be influenced by center-specific characteristics. Future work can be done by considering the interaction and mixed effects for model improvement.

Code Appendix:

```
knitr::opts_chunk$set(echo = TRUE)
knitr::opts_chunk$set(message = F)
knitr::opts_chunk$set(warning = F)
library(mice)
library(gtsummary)
library(naniar)
library(tidyverse)
library(knitr)
library(glmnet)
library(L0Learn)
library(nestfs)
library(pROC)
library(DescTools)
library(gt)
tracheo <- read.csv("/Users/kerryqwq/Downloads/project2.csv")
summary(tracheo)
tracheo$mat_ethn <- as.factor(tracheo$mat_ethn)
tracheo$del_method <- as.factor(tracheo$del_method)
tracheo$prenat_ster <- as.factor(tracheo$prenat_ster)
tracheo$com_prenat_ster <- as.factor(tracheo$com_prenat_ster)
tracheo$mat_chorio <- as.factor(tracheo$mat_chorio)
tracheo$gender <- as.factor(tracheo$gender)
tracheo$sga <- as.factor(tracheo$sga)
tracheo$any_surf <- as.factor(tracheo$any_surf)
tracheo$ventilation_support_level.36 <- as.factor(tracheo$ventilation_support_level.36)
tracheo$ventilation_support_level_modified.44 <- as.factor(tracheo$ventilation_support_level_modified.4
tracheo$Trach <- as.factor(tracheo$Trach)
tracheo$Death <- as.factor(tracheo$Death)
tracheo$med_ph.36 <- as.factor(tracheo$med_ph.36)
tracheo$med_ph.44 <- as.factor(tracheo$med_ph.44)
tracheo <- tracheo %>%
  mutate(center = ifelse(is.na(center), 1, center)) %>%
  mutate(outcome = case_when(tracheo$Trach ==1 & tracheo$Death == "Yes" ~ 1,
  tracheo$Trach ==1 | tracheo$Death == "Yes" ~ 1,
  tracheo$Trach ==0 & tracheo$Death == "No" ~ 0))
tracheo <- tracheo[!duplicated(tracheo$record_id),]
tracheo <- dplyr::select(tracheo, -c("mat_race"))
tracheo[which(tracheo$center==21),]$record_id <- 1000001
tracheo[which(tracheo$center==21),]$center <- 1

tracheo <- tracheo %>%
  arrange(record_id)

tracheo$outcome <- as.factor(tracheo$outcome)
tracheo$center <- as.factor(tracheo$center)
tracheo %>%
  dplyr::select(!record_id) %>%
  tbl_summary(
    by = outcome, # stratify by center
    type = all_continuous() ~ "continuous2", # ensure all continuous variables are treated as such
    missing = "no",
    statistic = list(
```

```
      all_continuous() ~ "{mean} ({sd})",  # for continuous variables, display mean (SD), missing value
      all_categorical() ~ "{n} ({p}%)" )) %>%
  add_n(statistic = "{n}") %>%
  as_gt() %>%
  tab_options(
    table.font.size = "x-small"  # Options include "xx-small", "x-small", "small", etc.
  ) %>%
  tab_caption("Summary Statistics across Different Variables")
missing_names <- colnames(tracheo)[colSums(is.na(tracheo)) > 0]
tracheo_missing <- tracheo[,c(missing_names)]
tracheo_miss <- miss_var_summary(tracheo_missing)
kable(tracheo_miss, caption = "Missingness in Tracheostomy Dataset")
center1 <- apply(tracheo[tracheo$center==1, !(names(tracheo) %in% c("record_id", "center"))], 2, functi
center2 <- apply(tracheo[tracheo$center==2, !(names(tracheo) %in% c("record_id", "center"))], 2, functi
center3 <- apply(tracheo[tracheo$center==3, !(names(tracheo) %in% c("record_id", "center"))], 2, functi
center4 <- apply(tracheo[tracheo$center==4, !(names(tracheo) %in% c("record_id", "center"))], 2, functi
center7 <- apply(tracheo[tracheo$center==7, !(names(tracheo) %in% c("record_id", "center"))], 2, functi
center12 <- apply(tracheo[tracheo$center==12, !(names(tracheo) %in% c("record_id", "center"))], 2, funct
center16 <- apply(tracheo[tracheo$center==16, !(names(tracheo) %in% c("record_id", "center"))], 2, funct
center20 <- apply(tracheo[tracheo$center==20, !(names(tracheo) %in% c("record_id", "center"))], 2, funct
kable(data.frame(center1, center2, center3, center4, center7, center12, center16, center20), caption =
# Count NA values for each patient
na_count_per_patient <- rowSums(is.na(tracheo))

# Calculate the percentage of NAs for each patient
total_columns <- ncol(tracheo) - 1  # Subtract 1 to exclude the 'record_id' column from the total
percentage_na_per_patient <- (na_count_per_patient / total_columns) * 100

# Create a new data frame with the 'record_id', 'na_count', and 'percentage_na'
na_data <- data.frame(record_id = tracheo$record_id,
                      center = tracheo$center,
                      na_count = na_count_per_patient,
                      percentage_na = percentage_na_per_patient)

na_data %>%
  arrange(desc(na_count)) %>%
  head(5)
# # Order by 'na_count' in descending order
# na_data_ordered <- na_data[order(-na_data$na_count), ]
#
# # Print the ordered data
# print(na_data_ordered)

tracheo %>%
  group_by(center) %>%  # Group data by the 'center' column
  summarise(num_patients_with_na = sum(if_any(everything(), is.na)),
            per_patients_with_na = round(num_patients_with_na/n() * 100, 2)) %>%  # Count the number of
  arrange(desc(per_patients_with_na))

tracheo %>%
  select(!record_id) %>%
  tbl_summary(
    by = center, # stratify by center
```

```
    type = all_continuous() ~ "continuous2", # ensure all continuous variables are treated as such
    missing = "no",
    statistic = list(
      all_continuous() ~ "{mean} ({sd})",  # for continuous variables, display mean (SD), missing value
      all_categorical() ~ "{n} ({p}%)" )) %>%
  add_n(statistic = "{n}")
set.seed(1)
tracheo_sub <- subset(tracheo, select = -c(record_id, Death, Trach))
ignore <- sample(c(TRUE, FALSE), size = 996, replace = TRUE, prob = c(0.3, 0.7))
tracheo_mice_out <- mice(tracheo_sub, m=5, seed = 1, ignore = ignore, print=F)

imp.train <- filter(tracheo_mice_out, !ignore)
imp.test <- filter(tracheo_mice_out, ignore)
tracheo_imp_train <- vector("list",5)
tracheo_imp_test <- vector("list",5)
for (i in 1:5) {
  tracheo_imp_train[[i]] <- mice::complete(imp.train,i)
  tracheo_imp_test[[i]] <- mice::complete(imp.test,i)
}
#########################################################
#### Lasso Model####
#########################################################

lasso <- function(df) {
  #' Runs 10-fold CV for lasso and returns corresponding coefficients
  #' @param df, data set
  #' @return coef, coefficients for minimum cv error

  # Matrix form for ordered variables
  x.ord <- model.matrix(outcome~., data = df)[,-1]
  y.ord <- df$outcome

  # Generate folds
  k <- 10
  set.seed(1) # consistent seeds between imputed data sets
  folds <- sample(1:k, nrow(df), replace=TRUE)

  # Lasso model
  lasso_mod <- cv.glmnet(x.ord, y.ord, nfolds = 10, foldid = folds,
                         alpha = 1, family = "binomial")

  lasso_ml <- glmnet(x.ord, y.ord, lambda = lasso_mod$lambda.min, alpha = 1, family = "binomial")
  # Get coefficients
  coef <- coef(lasso_ml, lambda = lasso_ml$lambda.min)
  return(coef)
}

  # Find average lasso coefficients over imputed datasets
lasso_coef1 <- lasso(tracheo_imp_train[[1]])
lasso_coef2 <- lasso(tracheo_imp_train[[2]])
lasso_coef3 <- lasso(tracheo_imp_train[[3]])
lasso_coef4 <- lasso(tracheo_imp_train[[4]])
lasso_coef5 <- lasso(tracheo_imp_train[[5]])
```

```
lasso_coef <- cbind(lasso_coef1, lasso_coef2, lasso_coef3,
                    lasso_coef4, lasso_coef5)
avg_coefs_lasso <- apply(lasso_coef, 1, mean)
lasso1_zero_names <- rownames(lasso_coef1)[as.numeric(lasso_coef1)== 0]
lasso2_zero_names <- rownames(lasso_coef2)[as.numeric(lasso_coef2)== 0]
lasso3_zero_names <- rownames(lasso_coef3)[as.numeric(lasso_coef3)== 0]
lasso4_zero_names <- rownames(lasso_coef4)[as.numeric(lasso_coef4)== 0]
lasso5_zero_names <- rownames(lasso_coef5)[as.numeric(lasso_coef5)== 0]

all_zero_names <- c(lasso1_zero_names, lasso2_zero_names, lasso3_zero_names, lasso4_zero_names, lasso5_z
allzerostable <- table(all_zero_names)
lasso_model <- avg_coefs_lasso[!names(avg_coefs_lasso) %in% c("bw", "mat_chorioYes", "p_delta.36", "p_de
bestsubset <- function(df) {
  x.ord <- model.matrix(outcome~., data = df)[,-1]
  y.ord <- df$outcome
  cv_subset <- L0Learn.cvfit(
  x.ord,
  y.ord,
  seed = 1,
  loss = "Logistic",
  penalty = "L0",
  nFolds = 10,
  intercept = TRUE)
  best_cvmeans <- which.min(cv_subset$cvMeans[[1]])
  coef <- c(cv_subset$fit$a0[[1]][best_cvmeans], cv_subset$fit$beta[[1]][,best_cvmeans])
  return(coef)
}
subset_coef1 <-bestsubset(tracheo_imp_train[[1]])
subset_coef2 <-bestsubset(tracheo_imp_train[[2]])
subset_coef3 <-bestsubset(tracheo_imp_train[[3]])
subset_coef4 <-bestsubset(tracheo_imp_train[[4]])
subset_coef5 <-bestsubset(tracheo_imp_train[[5]])

subset_coef <- cbind(subset_coef1, subset_coef2, subset_coef3, subset_coef4, subset_coef5)
avg_coefs_subset <- apply(subset_coef, 1, mean)

subset1_zero_names <- names(avg_coefs_lasso)[which(subset_coef1 == 0)]
subset2_zero_names <- names(avg_coefs_lasso)[which(subset_coef2 == 0)]
subset3_zero_names <- names(avg_coefs_lasso)[which(subset_coef2 == 0)]
subset4_zero_names <- names(avg_coefs_lasso)[which(subset_coef3 == 0)]
subset5_zero_names <- names(avg_coefs_lasso)[which(subset_coef4 == 0)]

all_zero_names <- c(subset1_zero_names, subset2_zero_names, subset3_zero_names, subset4_zero_names, subs
names(avg_coefs_subset) <- names(avg_coefs_lasso)
allzerostable <- table(all_zero_names)
subsetlogistic <- avg_coefs_subset[!names(avg_coefs_subset) %in% c("any_surfYes", "birth_hc", "bw", "ge
# cvforward <- function(df) {
#   #' Runs 10-fold CV for lasso and returns corresponding coefficients
#   #' @param df, data set
#   #' @return coef, coefficients for minimum cv error
#
# variable <- c(names(df)[-27])
# fs.res<- fs(outcome ~ 1, data=df, family=binomial(),
```

```r
# choose.from=variable, num.inner.folds=10)
# logistic <- glm(paste("outcome ~", paste(fs.res$final.model, collapse = " + ")), data = df, family =
# coef <- coef(logistic)
# return(coef)
# }
#
# forward_coef1 <-cvforward(tracheo_imp_train[[1]])
# forward_coef2 <-cvforward(tracheo_imp_train[[2]])
# forward_coef3 <-cvforward(tracheo_imp_train[[3]])
# forward_coef4 <-cvforward(tracheo_imp_train[[4]])
# forward_coef5 <-cvforward(tracheo_imp_train[[5]])
#
# coef_forward1 <- numeric(length(avg_coefs_lasso))
# index <- which(names(avg_coefs_lasso) %in% names(forward_coef1))
# coef_forward1[index] <- as.numeric(forward_coef1)
#
# coef_forward2 <- numeric(length(avg_coefs_lasso))
# index <- which(names(avg_coefs_lasso) %in% names(forward_coef2))
# coef_forward2[index] <- as.numeric(forward_coef2)
#
# coef_forward3 <- numeric(length(avg_coefs_lasso))
# index <- which(names(avg_coefs_lasso) %in% names(forward_coef3))
# coef_forward3[index] <- as.numeric(forward_coef3)
#
# coef_forward4 <- numeric(length(avg_coefs_lasso))
# index <- which(names(avg_coefs_lasso) %in% names(forward_coef4))
# coef_forward4[index] <- as.numeric(forward_coef4)
#
# coef_forward5 <- numeric(length(avg_coefs_lasso))
# index <- which(names(avg_coefs_lasso) %in% names(forward_coef5))
# coef_forward5[index] <- as.numeric(forward_coef5)
#
# names(coef_forward1)<- names(avg_coefs_lasso)
# names(coef_forward2)<- names(avg_coefs_lasso)
# names(coef_forward3)<- names(avg_coefs_lasso)
# names(coef_forward4)<- names(avg_coefs_lasso)
# names(coef_forward5)<- names(avg_coefs_lasso)
#
# forward_coef <- cbind(coef_forward1, coef_forward2, coef_forward3, coef_forward4, coef_forward5)
# avg_coefs_forward <- apply(forward_coef, 1, mean)
# forward1_selected_names <- names(coef_forward1)[as.numeric(coef_forward1)!= 0]
# forward2_selected_names <- names(coef_forward2)[as.numeric(coef_forward2)!= 0]
# forward3_selected_names <- names(coef_forward3)[as.numeric(coef_forward3)!= 0]
# forward4_selected_names <- names(coef_forward4)[as.numeric(coef_forward4)!= 0]
# forward5_selected_names <- names(coef_forward5)[as.numeric(coef_forward5)!= 0]
# #
# all_selected_names <- c(forward1_selected_names, forward2_selected_names, forward3_selected_names, fo
# allselectedtable <- table(all_selected_names)
# allselectedtable
# forwardmodel <- avg_coefs_forward[names(avg_coefs_forward) %in% c("(Intercept)", "med_ph.441", "peep_
# forwardmodel
coeff_df <- data.frame(coefs_lasso = avg_coefs_lasso, coefs_subset_logistic = avg_coefs_subset)
kable(coeff_df, caption = "Regression Coefficient Comparisons", align = "l",digits = 3)
```

```r
zero_freq <- data.frame(lasso = rowSums(as.matrix(lasso_coef)==0), subset = rowSums(as.matrix(subset_co
kable(zero_freq, caption = "Frequency of Zero Coefficients Across Five Imputed Sets", align = "l", digi
auc_lasso <- numeric(5)
sensitivity_lasso <- numeric(5)
specificity_lasso <- numeric(5)
threshold_lasso <- numeric(5)
BrierScore_lasso <- numeric(5)

for (i in 1:5) {
x_test <- model.matrix(outcome~., data=tracheo_imp_test[[i]])
y_test <- tracheo_imp_test[[i]]$outcome

qq <- x_test[,colnames(x_test) %in% names(lasso_model)]
qq <- as.matrix(qq)
dimnames(qq) <- NULL
pred <- qq %*% c(as.numeric(lasso_model))
predprob_lasso <- 1 / (1 + exp(-pred))

roc_info <- roc(predictor=as.numeric(predprob_lasso),
                response=y_test)
roc_coord <- coords(roc = roc_info, x="best")
auc_lasso[i] <- auc(roc_info)
sensitivity_lasso[i] <- roc_coord$sensitivity
specificity_lasso[i] <- roc_coord$specificity
threshold_lasso[i] <- roc_coord$threshold
BrierScore_lasso[i] <- BrierScore(as.numeric(as.character(y_test)), predprob_lasso)
}

auc_lasso <-  mean(auc_lasso)
sensitivity_lasso <- mean(sensitivity_lasso)
specificity_lasso <- mean(specificity_lasso)
threshold_lasso <- mean(threshold_lasso)
BrierScore_lasso <- mean(BrierScore_lasso)
auc_subsetlogistic <- numeric(5)
sensitivity_subsetlogistic <- numeric(5)
specificity_subsetlogistic <- numeric(5)
threshold_subsetlogistic <- numeric(5)
BrierScore_subsetlogistic <- numeric(5)

for (i in 1:5) {
x_test <- model.matrix(outcome~., data=tracheo_imp_test[[i]])
y_test <- tracheo_imp_test[[i]]$outcome

qq <- x_test[,colnames(x_test) %in% names(subsetlogistic)]
qq <- as.matrix(qq)
dimnames(qq) <- NULL
pred <- qq %*% c(as.numeric(subsetlogistic))
predprob_subsetlogistic <- 1 / (1 + exp(-pred))

roc_info <- roc(predictor=as.numeric(predprob_subsetlogistic),
                response=y_test)
roc_coord <- coords(roc = roc_info, x="best")
auc_subsetlogistic[i] <- auc(roc_info)
```

```
sensitivity_subsetlogistic[i] <- roc_coord$sensitivity
specificity_subsetlogistic[i] <- roc_coord$specificity
threshold_subsetlogistic[i] <- roc_coord$threshold
BrierScore_subsetlogistic[i] <- BrierScore(as.numeric(as.character(y_test)), predprob_subsetlogistic)
}

auc_subsetlogistic <-  mean(auc_subsetlogistic)
sensitivity_subsetlogistic <- mean(sensitivity_subsetlogistic)
specificity_subsetlogistic <- mean(specificity_subsetlogistic)
threshold_subsetlogistic <- mean(threshold_subsetlogistic)
BrierScore_subsetlogistic <- mean(BrierScore_subsetlogistic)
metrics_table <- data.frame(Measures = c("AUC", "Sensitivity", "Specificity","Threshold", "BrierScore")
          Lasso = round(c(auc_lasso, sensitivity_lasso, specificity_lasso, threshold_lasso, BrierScore_l
          SubsetLogistic = round(c(auc_subsetlogistic, sensitivity_subsetlogistic, specificity_subsetlog
kable(metrics_table, caption = "Model Evaluation Measures", align = "l",digits = 3)
```