Predicting the Need for Tracheostomy in Infants with Severe Bronchopulmonary Dysplasia

Author: Wanyi Chen Date: Nov 12, 2023

Abstract

Background: Tracheostomy is a procedure to help oxygen reach the lungs for patients with severe Bronchopulmonary Dysplasia (BPD). But there are advantages and disadvantges for performing a tracheostomy and different severity levels of BPD require different treatments. Developing a statistical model using clinical data may be one way to predict the need for tracheostomy.

Methods: LASSO and Best Subset model selection methods were used. Models with and without interactions were fitted. Cross-validations with 10 folds were used during the model derivation process. After obtaining predictions from the models, predicted results and model performances were compared and evaluated using AUC, Brier Score, and F1 Score.

Results: Both LASSO and Best Subset models performed better with interactions included, and the AUC scores were 0.8964 and 0.8908 respectively. Brier scores and F1 scores were all higher for LASSO models which indicates its prediction strength over Best Subset models.

Conclusions: From the model coefficients, non-invasive positive pressure ventilation support at week 36 and invasive positive pressure ventildation support at week 36 are positively impacting the outcome (death associated with tracheostomy) which implying higher risk of death associated with tracheostomy. Models should also be selected carefully to match the research goal.

Introduction

Bronchopulmonary Dysplasia (BPD) is the most common complication of prematurity with the severe form affecting 10,000 - 15,000 infants each year (McKinney and Levin). There are 4 grades to BPD, Mild (Grade 1): room air at 36 weeks of Post Menstrual Age (PMA);

Moderate (Grade 3) < 30% oxygen at 36 weeks of PMA; Severe (Grade 4) >30% oxygen at 36 weeks of PMA or on Positive Pressure Ventilation (PPV); Very Severe (Grade 4): Death between 14 days and 36 weeks (NHLBI 2001). Within these levels of BPD severity, patients with Grade 3 BPD depends on a ventilator at 36 weeks corrected gestational age and, 75% of these patients will remain on a ventilator when they are discharged from the hospital while 25% will not need a ventilator. When discharged from the hospital on ventilator, a patient needs tracheostomy which is a surgical hole in the patient's neck that allows them to be hooked up with a tracheostomy tube or ventilator and this does not need to be permanent. However, performing a tracheostomy comes with benefits as well as associated risks. Some benefits of a tracheostomy includes providing a stable airway for the patient, improving patient's age-appropriate interactions, improving patient's participation in developmental care and more importantly, tracheostomy performed within 4 months of age is associated with improved outcomes (Mckinney and Levin). The associated risks can included: increased risk of death compared to no tracheostomy, accidental decannulation that can lead to death, and increased rates of infection from skin, trachea, and lungs.

Given the advantages and disadvantages of performing a tracheostomy, it is important to understand and identify which groups of patients really needs tracheostomy. It is also important to find out the ideal time frame to refer a patient for tracheostomy. The goal of this project, is to develop a statistical model using clinical data at 36 and 44 weeks PMA to predict the need for tracheostomy or death prior to discharge.

Study Population

Data Collection

The data were collected from the BPD Collaborative Registry, a multi-center consortium of interdisciplinary BPD programs located in the United States and Sweden. The BPD Collaborative Registry was formed to promote research to enhance the care of children with severe forms of BPD and 10 centers had contributed at the time of analysis. Study participants included infants whose gestational age is less than 32 weeks and who have severe BPD at 36-weeks PMA and a total of 996 patients were drawn in this study. Standard demographic and clinical data were collected at four time points: birth, 36 weeks PMA, 44 weeks PMA and discharge. Birth variables include weight, gestational age, prenatal steroids, maternal race, gender, and chorioamnionitis. Respiratory support variables include level of support (nothing, FIO2, non-invasive support, and invasive support), positive end-expiratory pressure (PEEP), Fraction of inspired oxygen, and peak respiratory pressure. Data related to pulmonary hypertension and tracheostomy variables at 36 and 44 weeks comprehensive geriatric assessment (CGA) were also collected.

Data Descriptive Summary

Table 1 gives some descriptive statistics of some selected variables in the data set. Center 2 has the highest number of patients participated (630 patients) while Center 20 and Center 21 only has 5 patients altogether. This unbalanced number of patients from each center will impact the model derivation process, which means that the predictions of the final model can be influenced the most by Center 2. The tracheostomy frequency is calculated for each center, and it looks like Center 12 and Center 1 have the highest frequency (0.5072 and 0.4154) even though the number of patients are only 65 and 69 respectively. The death frequency is also calculated by center, Center 12 and Center 1 are again having the highest two frequencies (0.2029 and 0.1077). The outcome variable (1 for both death and tracheostomy and 0 otherwise) in this analysis is constructed to develop a regression model to predict the composite outcome of tracheostomy/death to guide the indication criteria and timing of tracheostomy placement. The outcome frequency is the highest in Center 12 and Center 2 while other centers have a frequency of 0. It seems like had tracheostomy and died is a rare case across the 10 centers. Center 20 has the highest mean birth weight, Center 12 has the highest mean gestational age, Center 16 and Center 4 have the highest mean birth length, Center 20 and Center 16 have the highest mean head circumference in cm, and finally, Center 20 and Center 1 have the highest SGA (1 = small for gestational age, 0 = not small for gestational age) freqeuncy.

Table 1: Summary statistics for the selected variables grouped by center

Center	Number of Patients	Trach Freq	Death Freq	Outcome Freq	bw Mean	ga Mean	blength Mean	birth_hc Mean	sga Freq
2	630	0.1016	0.0461	0.0143	832.3508	25.8746	32.7475	23.3123	0.1887
12	69	0.5072	0.2029	0.1159	781.0000	26.0725	32.4062	23.2303	0.2464
1	65	0.4154	0.1077	0.0000	689.8769	25.6615	30.8393	22.5839	0.4062
4	60	0.1833	0.0169	0.0000	833.2500	25.7500	33.2034	23.7293	0.0847
3	57	0.0175	0.0175	0.0000	764.8070	25.7018	32.1754	23.4579	0.2407
5	40	0.1250	0.0500	0.0000	605.3500	24.0750	29.4615	21.0500	0.2000
16	38	0.0263	0.0000	0.0000	889.2895	26.2895	33.7105	23.7645	0.1842
7	32	0.0312	0.0000	0.0000	724.8750	25.0938	32.0769	22.2654	0.2500
20	4	0.0000	0.0000	0.0000	1088.7500	25.7500	32.5000	24.0125	0.5000
21	1	1.0000	0.0000	0.0000	590.0000	24.0000	29.0000	21.0000	0.0000

To understand the relationships of the variables in the data set, it is also important to obtain their correlations for better model derivation process. Figure 1 gives the correlations and their corresponding plots for the select variables. Weight at 36 weeks and weight at 44 weeks have the highest positive correlation of 0.735, and birth weight and gestational age have the second highest positive correlation of 0.696. Inspired_oxygen.36 and birth weight have the highest negative correlation of -0.126. Also note that fraction of inspired oxygen needed at 36 weeks

(inspired_oxygen.36) and outcome have a correlation of 0.217, and weight at 36 weeks and outcome have a correlation of -0.087, which are highest positive and negative values among the 5 outcome correlations. This suggests that these two variables can be potential predictors in model development.

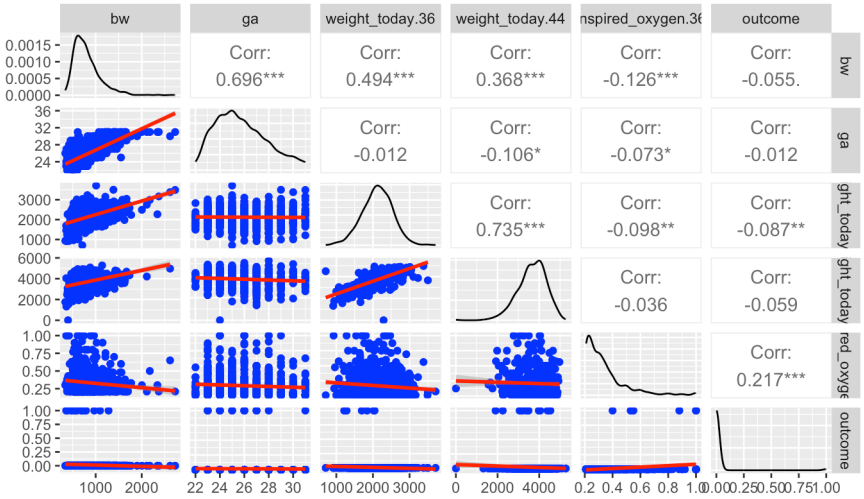

Figure 1: Correlations of some continous variables

In Figure 2, we can take a closer look at how death associated with tracheostomy is related to ventildation support and inspired oxygen. We observed that the no ventilation support groups have high outliers for inspired oxygen at both 36 weeks and 44 weeks. The non-invasive positive pressure ventilation support group shows no occurence or very small occurrences of the outcome. Within the invasive positive pressure ventilation support group, the outcome variable is also associated with higher fraction of inspired oxygen. Based on the observed patterns from Figure 2 and Figure 1, potential interactions for the models would be: `ventilation_support_level.36` and `inspired_oxygen.36` (interaction 1), `ventilation_support_level.36` and `med_ph.36` (interaction 2), `ventilation_support_level.44` and `inspired_oxygen.44` (interaction 3), and `ventilation_support_level.44` and `med_ph.44` (interaction 4).

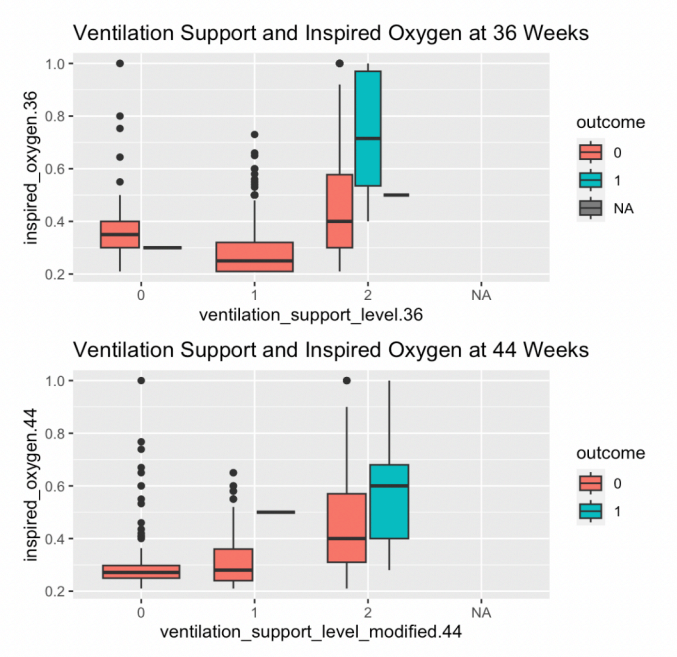

Figure 2: Ventilation support and fraction of inspired oxygen grouped by outcome of interest.

Missing Data

We do observed missingness in our data set, and Table 2 shows the variables that have an overall missingness percentage greater than 10%. There are only 14.91% of the observations in our data set are complete cases. No observations are completely empty but there are 3 duplicated observations in the data set, so there are a total of 996 observations/rows. Information related to standard demographic has low percentage of missingness across the 10 centers while the information related to ventilation support, inspired oxygen at 36 weeks and 44 weeks, and weight at 36 weeks and 44 weeks are ranging from about 0% to 25% of missingness within each center. The high overall missing percentages ranging from about 43% to 45%, are all variables related to clinical data at 36 and 44 weeks PMA. It is especially the case for the variables at week 44 and this could be due lost to follow up for patients who recovered or discharged. This large portion of missingness in week 44 would also influence the time effects in the final models.

Table 2. Missing percentages by centers for the variables that have an overall missingness percentage greater than 10% are included in this table. See Table 5 in appendix for the full table.

Vars	Overall Missingness	1	2	3	4	5	7	12	16	20	21
inspired_oxygen.44	44.98	0.90	25.40	3.82	6.02	0.90	2.11	2.51	3.31	0.0	0.0
p_delta.44	44.98	0.70	24.90	3.82	6.02	1.41	2.21	2.61	3.31	0.0	0.0
weight_today.44	44.78	0.70	25.30	3.82	6.02	0.90	2.11	2.61	3.31	0.0	0.0
peep_cm_h2o_modified.44	44.78	0.70	25.10	3.92	6.02	0.90	2.01	2.81	3.31	0.0	0.0
any_surf	43.47	2.51	29.62	0.10	4.42	0.50	2.61	1.20	2.21	0.2	0.1
ventilation_support_level_modified.44	42.57	0.40	24.00	3.71	6.02	0.90	2.01	2.21	3.31	0.0	0.0
med_ph.44	42.57	0.40	24.00	3.71	6.02	0.90	2.01	2.21	3.31	0.0	0.0
com_prenat_ster	19.38	1.61	10.54	1.00	1.71	0.30	0.90	2.61	0.50	0.2	0.0
p_delta.36	12.85	2.01	3.92	0.70	1.51	1.31	0.10	3.21	0.00	0.1	0.0
peep_cm_h2o_modified.36	11.75	2.41	4.12	1.10	0.60	0.00	0.10	3.41	0.00	0.0	0.0

The missing clinical data at 36 weeks and 44 weeks can be important information to be considered when deriving the model. For model derivation purposes, multiple imputation method is used to impute the missing values in the data set. Multiple imputation is performed in R by using the mice package. The seed is set to 500 for consistency, and the number of multiple imputations is set to 5 to account for the variability and randomness in the missing data. After obtaining the imputed data sets, the 5 data sets is then used to perform variable selection separately.

Model Derivation

The Lasso method is used for variable selection. The property of Lasso's penalty form is particularly useful for models with high multicollinearity and for feature selection purposes. Cross-validation of 10 folds is also used to determine the optimal tuning parameter (lambda) in Lasso regression. It involves dividing the data into subsets, and each iteration uses one subset for testing and the rest for training. The errors are accumulated, and the computed average error and its variation over the folds are used to assess model performance. Cross-validation in this case aims to minimize the lasso model's prediction error by eliminating unimportant effects. So Multiple imputation coupled with cross validation helps in terms of balancing the bias-variance trade-off. Another method used is the Best Subset Selection method which is also a useful feature selection tool. It essentially tests all possible combination of the predictor variables, and then selects the best model according to some statistical criteria. Cross-validation is also used to minimize the prediction error of the best subset models.

Table 3 shows the coefficient estimates for the four final models: Lasso, Lasso with interactions, Best Subset, and Best Subset with interactions. For each model, there were 5 sets of

coefficient estimates fitted for each of the 5 imputation data, and the coefficient estimates are average across the 5 sets to obtain the coefficients in the final model. In the final models, about 10 to 12 variables (record_id, Trach, and Death are not included in the variable list, and mat_race is removed due to inconsistency of data coding) were selected. The categorical variable `ventildation_support.36` (invasive positive pressure) has the highest main effects across all final models ranging from 1.57 to 1.88. This means, on average, if the patients are on invasive positive pressure ventilation support at week 36, then the occurence of outcome is about 1.57 to 1.88 higher relative to other ventilation supports (recall that the outcome variable is a combined variable of Tracheostomy and Death) while adjusting for other predictors in the model. This variable is also selected in all 5 of the imputation data which is a high coefficient frequency of 5. The high coefficient estimate and high frequency of `ventildation_support.36` implies that at 36 weeks, improper ventilation used at this time period would cause increased risk of death associated with tracheostomy. On the other hand, the main effects of `ventildation_support.44` are smaller compared to `ventildation_support.36`. On average, if the patients are on invasive positive pressure ventilation support at week 44, then the occurence of outcome is about 0.94 to 1.43 higher relative to other ventilation supports at week 44 while adjusting for other predictors in the model.

The continuous variable `inspired_oxygen.44` tends to have high main effects for all four models as well. The coefficient estimates range from 1.05 to 2.59, which means on average, an additional unit increase in fraction of inspired oxygen at 44 weeks will result in 1.05 to 2.59 unit increase in the outcome while adjusting for other predictors in the model. The frequency of the inspired oxygen variables and their coefficient estimates are again suggesting that, controlling the proper fraction of inspired oxygen for the patients is very crucial, especially at 36 weeks. Other variables that have high frequency complete prenatal steroids, peak inspiratory pressure (cmH2O) at 44 weeks, and medication for pulmonary hypertension at 44 weeks. Again, note that many of the variables related week 36 have high frequency which suggests week 36 is an important time frame for patients.

Comparing the coefficient estimates for the four final models, the selected variables are very similar with most of the variables relating to the two time points (week 36 and week 44). The interaction terms are all selected with high frequency across the 5 imputation set, and the absolute main effects are not low which implies the better performances of the model with interactions.

Table 3. Coefficient estimates and coefficient frequency for LASSO and Best Subset models.

Coefficient	Lasso Coef Estimate	Lasso Coef with Interactions	Best Subset Coef Estimate	Best Subset Coef with Interactions
(Intercept)	-3.11189	-6.30538	-5.370130	-7.73418
del_method2	0.36384	0.59416	0.556630	0.61914
prenat_ster2	0.35389	0.57018	0.344909	0.43856
com_prenat_ster2	0.23865	0.30538	0.629010	0.54440
ventilation_support_level.361	0.00000	-1.14719	-1.742050	-1.04105
ventilation_support_level.362	1.56766	1.56766	1.882930	1.75272
inspired_oxygen.36	1.21557	1.57018	0.397270	0.96102
peep_cm_h2o_modified.36	0.00000	-0.13493	-0.104580	0.00000
ventilation_support_level_modified.441	0.00000	0.00000	-0.279020	-1.64707
ventilation_support_level_modified.442	0.97943	1.27261	1.429170	0.93978
inspired_oxygen.44	1.50329	2.59416	1.045850	0.00000
peep_cm_h2o_modified.44	0.10915	0.10833	0.436190	0.30011
med_ph.441	1.42188	2.95273	1.629010	1.86540
interaction1	0.00000	-0.10388	0.000000	-0.52280
interaction2	0.00000	-0.47560	0.000000	0.86549
interaction3	0.00000	3.28261	0.000000	0.94246
interaction4	0.00000	-2.08819	0.000000	-1.09544

Model Evaluation

After obtaining the final model and train data model, evaluation metrics are calculated for the 2 models. Based on the scores on Table 4, it is obvious and expected that the final model performs better since it is fitted using the full data sets. For the full data model, the AUC score is about 0.8043 which close to 1 meaning fairly good prediction accuracy. Based on other BPD studies, that mortality rates from the time of tracheostomy to the time of initial hospital discharge have a range of 9–23% (Miller et al). Thus, the AUC scores are calculated using a threshold within this range. For Brier score, a score of 0 represents perfect accuracy and a score of 0.25 is the same as a chance. The Brier scores in this case range from 0.09 to 0.14 which are all lower than 0.25 and very close to 0. The Brier Scores are also lower in the model with interactions cases. This indicates the inclusion of interaction allowed better calibration of predicted probability of the outcome, and that the predicted probability increasingly equals to the observed probability. The F1 score is a measure of the harmonic mean of precision and recall and commonly used as an evalutaion metric for binary classification problems. It ranges

from 0 to 1, and 1 indicating perfect precision and recall. All four F1 scores are very close to 1 which means fairly good model performance.

Table 4. AUC, Brier scores, and F1 scores are calculated using the actual values and predicted values from the four models.

	AUC	Brier Score	F1 Score
LASSO	0.8610	0.1151	0.9243
LASSO with Interactions	0.8964	0.0932	0.9449
Best Subset	0.8474	0.1365	0.9168
Best Subset with Interactions	0.8908	0.0994	0.9420

Conclusion

Nevertheless, our final model gives some meaningful results in predicting the need for Tracheostomy in infants with severe BPD. The model coefficient estimates suggest that week 36 is an important time frame to refer a patient for proper tracheostomy. Since there are only 4 time points available, it is still unclear whether week 36 is too early for referal of tracheostomy or/and week 36 is a crucial time point for patients to receive proper ventilation that suit their individual condition. From the model result, Non-invasive positive pressure ventilation support at week 36 and invasive positive pressure ventildation support at week 36 are positively impacting the outcome (death associated with tracheostomy) which implying higher risk of death associated with tracheostomy.

In conclusion, the final models of this project performs fairly ideal with the given data set but may need further analyses to determine its statistical power using external data sets. In this project, only Lasso and Best Subset Selection were used, other methods such as Forward Stepwise and Ridge regression can be applied. There are also limitations cause by the missingness and imbalances of the given data set. Even though performing multiple imputations can help in filling in the missing data, the imputation process gets more complicated and computationally intensive as more methods (Lasso, Forward-Stepwise, and Best Subset Selection, etc) are performed and more data sets are imputed. As more methods are being tried for the model derivation process, it also gets more difficult to choose among the final models. It is important to think about the research goal and make sure the model selected is appropriate and matched the purpose of the analysis.

Reference

McKinney, R., & Levin, J. (2023, Oct 16). *Predicting the need for tracheostomy in infants with severe bronchopulmonary dysplasia* [PowerPoint slides]. Alpert Medical School, Brown University.

Miller AN, Shepherd EG, Manning A, Shamim H, Chiang T, El-Ferzli G, Nelin LD. Tracheostomy in Severe Bronchopulmonary Dysplasia-How to Decide in the Absence of Evidence. Biomedicines. 2023 Sep 19;11(9):2572. doi: 10.3390/biomedicines11092572. PMID: 37761012; PMCID: PMC10526913.

Silberzahn, R., Uhlmann, E. L., Martin, D. P., Anselmi, P., Aust, F., Awtrey, E., . . . Nosek, B. A.
(2018). Many analysts, one data set: Making transparent how variations in analytic choices affect results. Advances in Methods and Practices in Psychological Science, 1, 337–356. doi:10.1177/2515245917747646

Trevor Hastie, Robert Tibshirani, Ryan Tibshirani "Best Subset, Forward Stepwise or Lasso? Analysis and Recommendations Based on Extensive Comparisons," Statistical Science, Statist. Sci. 35(4), 579-592, (November 2020)

Appendix

Table 2: Overall missing percentages for each variable in the data set and missing percentages grouped by center for variable.

	Overall missingness	Center 1	Center 2	Center 3	Center 4	Center 5	Center 7	Center 12	Center 16	Center 20	Center 21
inspired oxygen.44	44.98	0.90	25.40	3.82	6.02	0.90	2.11	2.51	3.31	0.0	0.0
p delta.44	44.98	0.70	24.90	3.82	6.02	1.41	2.21	2.61	3.31	0.0	0.0
weight today.44	44.78	0.70	25.30	3.82	6.02	0.90	2.11	2.61	3.31	0.0	0.0
peep cm h2o modified.44	44.78	0.70	25.10	3.92	6.02	0.90	2.01	2.81	3.31	0.0	0.0
any surf	43.47	2.51	29.62	0.10	4.42	0.50	2.61	1.20	2.21	0.2	0.1

	Overall missingness	Center 1	Center 2	Center 3	Center 4	Center 5	Center 7	Center 12	Center 16	Center 20	Center 21
ventilation support level modified.44	42.57	0.40	24.00	3.71	6.02	0.90	2.01	2.21	3.31	0.0	0.0
med ph.44	42.57	0.40	24.00	3.71	6.02	0.90	2.01	2.21	3.31	0.0	0.0
com prenat ster p	19.38	1.61	10.54	1.00	1.71	0.30	0.90	2.61	0.50	0.2	0.0
delta.36	12.85	2.01	3.92	0.70	1.51	1.31	0.10	3.21	0.00	0.1	0.0
hosp dc ga	12.45	6.43	0.00	0.00	6.02	0.00	0.00	0.00	0.00	0.0	0.0
peep cm h2o modified.36	11.75	2.41	4.12	1.10	0.60	0.00	0.10	3.41	0.00	0.0	0.0
weight today.36	9.24	1.71	3.61	0.30	0.60	0.00	0.10	2.91	0.00	0.0	0.0
inspired oxygen.36	9.24	1.81	3.61	0.20	0.30	0.00	0.10	2.91	0.00	0.2	0.1
blength	7.83	0.90	2.41	0.00	0.10	0.10	0.60	3.71	0.00	0.0	0.0
birth hc	7.73	0.90	2.91	0.00	0.20	0.00	0.60	3.11	0.00	0.0	0.0
mat chorio	6.22	3.01	0.00	2.31	0.10	0.10	0.10	0.00	0.50	0.1	0.0
mat ethn	5.72	2.51	0.00	0.20	0.20	0.00	2.71	0.00	0.00	0.1	0.0
prenat ster	3.51	0.40	0.10	0.30	0.10	0.00	0.20	2.31	0.10	0.0	0.0

	Overall missingness	Center 1	Center 2	Center 3	Center 4	Center 5	Center 7	Center 12	Center 16	Center 20	Center 21
ventilation_support_level.36	3.01	0.10	0.90	0.10	0.00	0.00	0.00	1.91	0.00	0.0	0.0
meds_ph.36	3.01	0.10	0.90	0.10	0.00	0.00	0.00	1.91	0.00	0.0	0.0
sga	1.51	0.10	1.00	0.30	0.10	0.00	0.00	0.00	0.00	0.0	0.0
gender	0.40	0.10	0.20	0.10	0.00	0.00	0.00	0.00	0.00	0.0	0.0
del_method	0.30	0.10	0.00	0.00	0.00	0.00	0.00	0.20	0.00	0.0	0.0
Death	0.20	0.00	0.10	0.00	0.10	0.00	0.00	0.00	0.00	0.0	0.0
outcome	0.20	0.00	0.10	0.00	0.10	0.00	0.00	0.00	0.00	0.0	0.0
record_id	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0
center	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0
bw	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0
ga	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0
Trach	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.0	0.0

Code Appendix

```
library(tidyverse)
library(mice)
library(gtsummary)
library(dplyr)
library(glmnet)
library(leaps)
library(tableone)
library(DescTools)
library(Metrics)
library(pROC)
library(GGally)

data <- read.csv("project2.csv") %>%
  select(-mat_race)
head(data)
```

```r
dim(data)

complete_perc <- nrow(data[complete.cases(data) == TRUE,])/nrow(data)
complete_perc
# complete cases are only about 15%

# row with all na data
na_ind <- apply(data, 1, function(x) all(is.na(x)))
length(na_ind[na_ind == TRUE]) # no row with all na data

unique(data$center)

# fill the missing center numbers
data$center[is.na(data$center)] <- 1
data[is.na(data$center),]

head(data)

# look for duplicated observations and remove
data <- data[!duplicated(data) == TRUE,]

# transform categorical variables to factors

data <- data %>%
  mutate(mat_chorio = case_when(mat_chorio == "No" ~ 2,
                                mat_chorio == "Yes" ~ 1,
                                mat_chorio == "Unknown" ~ 3,
                                mat_chorio == NA ~ NA)) %>%
  mutate(sga = case_when(sga == "Not SGA" ~ 0,
                         sga == "SGA" ~ 1,
                         sga == NA ~ NA)) %>%
  mutate(any_surf = case_when(any_surf == "No" ~ 2,
                              any_surf == "Yes" ~ 1,
                              any_surf == "Unknown" ~ 3,
                              any_surf == NA ~ NA)) %>%
  mutate(prenat_ster = case_when(prenat_ster == "No" ~ 2,
                                 prenat_ster == "Yes" ~ 1,
                                 prenat_ster == "Unknown" ~ 3,
                                 prenat_ster == NA ~ NA)) %>%
  mutate(com_prenat_ster = case_when(com_prenat_ster == "No" ~ 2,
                                 com_prenat_ster == "Yes" ~ 1,
```

```
                                com_prenat_ster == "Unknown" ~ 3,
                                com_prenat_ster == NA ~ NA)) %>%
  mutate(gender = case_when(gender == "Male" ~ 1,
                            gender == "Female" ~ 2,
                            gender == "Ambiguous" ~ 3,
                            gender == NA ~ NA))
data$center <- as.factor(data$center)
data$mat_ethn <- as.factor(data$mat_ethn)
data$del_method <- as.factor(data$del_method)
data$prenat_ster <- as.factor(data$prenat_ster)
data$com_prenat_ster <- as.factor(data$com_prenat_ster)
data$mat_chorio <- as.factor(data$mat_chorio)
data$gender <- as.factor(data$gender)
data$sga <- as.factor(data$sga)
data$med_ph.36 <- as.factor(data$med_ph.36)
data$med_ph.44 <- as.factor(data$med_ph.44)
data$any_surf <- as.factor(data$any_surf)
data$ventilation_support_level.36 <- as.factor(data$ventilation_support_level.36)
data$ventilation_support_level_modified.44 <- as.factor(data$ventilation_support_level_mod

# create outcome variable based on death and trach status
data <- data %>%
  mutate(Death = case_when(Death == "No" ~ 0,
                           Death == "Yes" ~ 1,
                           Death == NA ~ NA)) %>%
  mutate(outcome = case_when(Death == 1 & Trach == 1 ~ 1,
                             Death == 1 & Trach == 0 ~ 0,
                             Death == 0 & Trach == 1 ~ 0,
                             Death == 0 & Trach == 0 ~ 0,
                             Death == NA & Trach == NA ~ NA,
                             Death == 1 & Trach == NA ~ NA,
                             Death == NA & Trach == 1 ~ NA))
data$Trach <- as.factor(data$Trach)
data$Death <- as.factor(data$Death)
data$outcome <- as.factor(data$outcome)

head(data, 10)

# calculate missing percentage for each variables
missing <- round(apply(data, 2, function(x) sum(is.na(x)))/nrow(data), 4)
missing1 <- round(apply(data %>% filter(center == 1), 2, function(x) sum(is.na(x)))/nrow(d
```

```r
missing2 <- round(apply(data %>% filter(center == 2), 2, function(x) sum(is.na(x)))/nrow(d
missing3 <- round(apply(data %>% filter(center == 3), 2, function(x) sum(is.na(x)))/nrow(d
missing4 <- round(apply(data %>% filter(center == 4), 2, function(x) sum(is.na(x)))/nrow(d
missing5 <- round(apply(data %>% filter(center == 5), 2, function(x) sum(is.na(x)))/nrow(d
missing7 <- round(apply(data %>% filter(center == 7), 2, function(x) sum(is.na(x)))/nrow(d
missing12 <- round(apply(data %>% filter(center == 12), 2, function(x) sum(is.na(x)))/nrow
missing16 <- round(apply(data %>% filter(center == 16), 2, function(x) sum(is.na(x)))/nrow
missing20 <- round(apply(data %>% filter(center == 20), 2, function(x) sum(is.na(x)))/nrow
missing21 <- round(apply(data %>% filter(center == 21), 2, function(x) sum(is.na(x)))/nrow
missing <- missing * 100
missing_table <- cbind(missing, missing1, missing2, missing3, missing4,
                       missing5, missing7, missing12, missing16, missing20,
                       missing21)
missing_table <- data.frame(missing_table) %>%
  arrange(desc(missing))

missing_table

# brief look at the data set variables
# data %>%
#   select(-record_id) %>%
#   tbl_summary(by = center,
#               missing = "no",
#             type = list(
#               all_continuous() ~ "continuous2"),
#             statistic = list(
#               all_continuous() ~ c("{mean} ({sd})",
#                                    "{N_miss} ({p_miss}%)"),
#                 all_categorical() ~ "{n} ({p}%)"
#               ),
#               missing_text = "(Missing)") %>%
#   modify_table_body(
#     dplyr::mutate,
#     label = ifelse(label == "N missing (% missing)",
#                    "Unknown",
#                    label))

data %>%
  select(-record_id) %>%
  tbl_summary(by = center,
              all_continuous() ~ "{mean} ({sd})",
```

```
                all_categorical() ~ "{n} ({p}%)"
              ,
              missing_text = "(Missing)")

# calculate descriptive statistics
des_summary <- data %>%
  group_by(center) %>%
  mutate(num_patients = n()) %>%
  mutate(trach_freq = mean(ifelse(Trach == 0, 0, 1), na.rm = T)) %>%
  mutate(death_freq = mean(ifelse(Death == 0, 0, 1), na.rm = T)) %>%
  mutate(outcome_freq = mean(ifelse(outcome == 0, 0, 1), na.rm = T)) %>%
  mutate(female_freq = mean(ifelse(data$gender == 1, 0, 1), na.rm=T)) %>%
  mutate(bw_mean = mean(bw)) %>%
  mutate(ga_mean = mean(ga)) %>%
  mutate(blength_mean = mean(blength, na.rm = T)) %>%
  mutate(birth_hc_mean = mean(birth_hc, na.rm = T)) %>%
  mutate(sga_freq = mean(ifelse(sga == 1, 1, 0), na.rm =T)) %>%
  select( center, num_patients, trach_freq, death_freq, outcome_freq, bw_mean, ga_mean, bl
         birth_hc_mean, sga_freq) %>%
  mutate(across(where(is.numeric), round, digits=4))


kableone(unique(des_summary))

# a function to customize fitted lines in scatter plots
lower_func <- function(data, mapping, method = "lm", ...) {
#' @description customize fitted lines in ggally plots
#' @param data input data for plotting
#' @param mapping the mapping for variables in the input data
#' @param method the method to be used for fitted lines
#' @return a ggplot with fitted lines and the corresponding mapping

  p <- ggplot(data = data, mapping = mapping) +
    geom_point(colour = "blue") +
    geom_smooth(method = method, color = "red", ...)

  return(p)
}

ggpair_df1 <- data[, c("bw","ga",
                       "weight_today.36", "weight_today.44",
```

```
                              "inspired_oxygen.36", "outcome")]
ggpair_df1$outcome <- ifelse(ggpair_df1$outcome == 0, 0, 1)

ggpairs(ggpair_df1, columns = 1:ncol(ggpair_df1),
        lower = list(continuous = wrap(lower_func, method = "lm")),
        title = "",
        axisLabels = "show", columnLabels = colnames(ggpair_df1))


# multiple imputation
data_mice <- mice(data, m = 5, seed = 500)

data_mice1 <- mice::complete(data_mice,1)[,c(2:27, 30)]
data_mice2 <- mice::complete(data_mice,2)[,c(2:27, 30)]
data_mice3 <- mice::complete(data_mice,3)[,c(2:27, 30)]
data_mice4 <- mice::complete(data_mice,4)[,c(2:27, 30)]
data_mice5 <- mice::complete(data_mice,5)[,c(2:27, 30)]

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 <- as.matrix(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_cv_mod <- cv.glmnet(x.ord, y.ord, nfolds = 10, foldid = folds,
                            alpha = 1, family = "binomial")
  lasso_mod <- glmnet(x.ord, y.ord, nfolds = 10,
                      lambda = lasso_cv_mod$lambda.min, alpha = 1)

  # Get coefficients
  coef <- coef(lasso_mod)
  return(coef)
```

```
}

# obtain coefficient estimates from lasso
coef1 <- lasso(data_mice1)
mice_coef1 <- as.vector(lasso(data_mice1))
mice_coef2 <- as.vector(lasso(data_mice2))
mice_coef3 <- as.vector(lasso(data_mice3))
mice_coef4 <- as.vector(lasso(data_mice4))
mice_coef5 <- as.vector(lasso(data_mice5))

# predict using coefs
model_predict <- function(df, coefs){
  #' @param df a dataframe
  #' @param coefs a vector of coefficient etsimates
  #' @return model predictions

  outcome_preds <- rep(NA, nrow(df))
  for (i in 1:nrow(df)){
    preds <- coefs[1] + (as.numeric(df$center[i] == 2) * coefs[2]) +
      (as.numeric(df$center[i] == 3) * coefs[3]) +
      (as.numeric(df$center[i] == 5) * coefs[5]) +
      (as.numeric(df$center[i] == 7) * coefs[6]) +
      (as.numeric(df$center[i] == 12) * coefs[7]) +
      (as.numeric(df$center[i] == 16) * coefs[8]) +
      (as.numeric(df$center[i] == 21) * coefs[10]) +
      (df$bw[i] * coefs[12]) + (df$blength[i] * coefs[14]) +
      (df$birth_hc[i] * coefs[15]) + (as.numeric(df$del_method[i] == 2) * coefs[16]) +
      (as.numeric(df$prenat_ster[i] == 2) * coefs[17]) +
      (as.numeric(df$com_prenat_ster[i] == 2) * coefs[18]) +
      (as.numeric(df$mat_chorio[i] == 2) * coefs[19]) +
      (as.numeric(df$gender[i] == 2) * coefs[20]) +
      (as.numeric(df$any_surf[i] == 2) * coefs[22]) +
      (df$weight_today.44[i] * coefs[23]) +
      (as.numeric(df$ventilation_support_level.36[i] == 1) * coefs[24]) +
      (as.numeric(df$ventilation_support_level.36[i] == 2) * coefs[25]) +
      (df$inspired_oxygen.36[i] * coefs[26]) + (df$p_delta.36[i] * coefs[27]) +
      (df$peep_cm_h2o_modified.36[i] * coefs[28]) +
      (as.numeric(df$med_ph.36[i] == 1) * coefs[29]) +
      (df$weight_today.44[i] * coefs[30]) +
      (as.numeric(df$ventilation_support_level_modified.44[i] == 2) * coefs[32]) +
      (df$inspired_oxygen.44[i] * coefs[33]) + (df$p_delta.44[i] * coefs[34]) +
```

```
      (as.numeric(df$med_ph.44[i] == 1) * coefs[36]) +
      (df$hosp_dc_ga[i] * coefs[37])

      outcome_preds[i] <- preds
  }
  return(outcome_preds)
}

# split data into test and train sets (25% and 75%)
set.seed(1)
test_indice <- sample(nrow(data), 249, replace = FALSE)
train_data <- data[-test_indice, ]
test_data <- data[test_indice, ]

# use mice to impute the missingness in cross validation sets
train_mice <- mice(train_data, m = 5, print = FALSE, seed = 1)
test_mice <- mice.mids(train_mice, newdata = test_data)

train_mice1 <- mice::complete(train_mice,1)[,c(2:27, 30)]
test_mice1 <- mice::complete(test_mice,1)[,c(2:27, 30)]
train_mice2 <- mice::complete(train_mice,2)[,c(2:27, 30)]
test_mice2 <- mice::complete(test_mice,2)[,c(2:27, 30)]
train_mice3 <- mice::complete(train_mice,3)[,c(2:27, 30)]
test_mice3 <- mice::complete(test_mice,3)[,c(2:27, 30)]
train_mice4 <- mice::complete(train_mice,4)[,c(2:27, 30)]
test_mice4 <- mice::complete(test_mice,4)[,c(2:27, 30)]
train_mice5 <- mice::complete(train_mice,5)[,c(2:27, 30)]
test_mice5 <- mice::complete(test_mice,5)[,c(2:27, 30)]

# use lasso to obtain coefficients for cross validation sets
train_mice_coef1 <- as.vector(lasso(train_mice1))
train_mice_coef2 <- as.vector(lasso(train_mice2))
train_mice_coef3 <- as.vector(lasso(train_mice3))
train_mice_coef4 <- as.vector(lasso(train_mice4))
train_mice_coef5 <- as.vector(lasso(train_mice5))

# create table for coefficients and their frequencies
mean_mice_mat <- cbind(mice_coef1, mice_coef2, mice_coef3, mice_coef4, mice_coef5)
train_mean_mice_mat <- cbind(train_mice_coef1, train_mice_coef2, train_mice_coef3,
                             train_mice_coef4, train_mice_coef5)
mean_mice_coef <- rowMeans(mean_mice_mat)
```

```r
train_mean_mice_coef <- rowMeans(train_mean_mice_mat)

coef_names <- rownames(as.matrix(coef1))
coef_freq <- rowSums(mean_mice_mat != 0)
train_coef_freq <- rowSums(train_mean_mice_mat != 0)

coef_table <- data.frame(coef_names,
                         round(mean_mice_coef, 6), coef_freq,
                         round(train_mean_mice_coef, 6), train_coef_freq)
coef_table <- coef_table %>%
  filter(!(coef_freq == 0 & train_coef_freq == 0))
colnames(coef_table) <- c("Coefficient", "Coefficient Estimate (Full Data)",
                          "Coefficient Frequency (Full Data)",
                          "Coefficient Estimate (Train Data)",
                          "Coefficient Frequency (Train Data)")
coef_table

# predict for train data model
train_model_predict <- function(df, coefs){
  #' @param df a dataframe
  #' @param coefs a vector of coefficient etsimates
  #' @return model predictions

  outcome_preds <- rep(NA, nrow(df))
  for (i in 1:nrow(df)){
    preds <- coefs[1] + (as.numeric(df$center[i] == 2) * coefs[2]) +
      (as.numeric(df$center[i] == 12) * coefs[7]) +
      (df$blength[i] * coefs[14]) +
      (as.numeric(df$del_method[i] == 2) * coefs[16]) +
      (as.numeric(df$prenat_ster[i] == 2) * coefs[17]) +
      (as.numeric(df$mat_chorio[i] == 2) * coefs[19]) +
      (as.numeric(df$gender[i] == 2) * coefs[20]) +
      (as.numeric(df$any_surf[i] == 2) * coefs[22]) +
      (df$weight_today.44[i] * coefs[23]) +
      (as.numeric(df$ventilation_support_level.36[i] == 1) * coefs[24]) +
      (as.numeric(df$ventilation_support_level.36[i] == 2) * coefs[25]) +
      (df$inspired_oxygen.36[i] * coefs[26]) +
      (df$peep_cm_h2o_modified.36[i] * coefs[28]) +
      (as.numeric(df$med_ph.36[i] == 1) * coefs[29]) +
      (as.numeric(df$ventilation_support_level_modified.44[i] == 1) * coefs[31]) +
      (as.numeric(df$ventilation_support_level_modified.44[i] == 2) * coefs[32]) +
```

```
      (df$inspired_oxygen.44[i] * coefs[33]) + (df$p_delta.44[i] * coefs[34]) +
      (df$hosp_dc_ga[i] * coefs[37])

      outcome_preds[i] <- preds
  }
  return(outcome_preds)
}


# obtain predictions from 2 models
data_mice_m5 <- rbind(data_mice1, data_mice2, data_mice3, data_mice4, data_mice5)
test_data_mice_m5 <- rbind(test_mice1, test_mice2, test_mice3, test_mice4,
                           test_mice5)
full_imputed_preds <- model_predict(data_mice_m5, mean_mice_coef)
full_imputed_preds <- ifelse(full_imputed_preds > 0.17, 1, 0)
test_imputed_preds <- train_model_predict(test_data_mice_m5, train_mean_mice_coef)
test_imputed_preds <- ifelse(test_imputed_preds > 0.17, 1, 0)

# calculate evalution metrics
full_auc <- auc(data_mice_m5$outcome, full_imputed_preds)
test_auc <- auc(test_data_mice_m5$outcome, test_imputed_preds)

full_rmse <- rmse(ifelse(data_mice_m5$outcome == 1, 1, 0), full_imputed_preds)
test_rmse <- rmse(ifelse(test_data_mice_m5$outcome == 1, 1, 0), test_imputed_preds)

full_bs <- BrierScore(model_predict(data_mice_m5, mean_mice_coef),
          ifelse(data_mice_m5$outcome == 1, 1, 0))
test_bs <- BrierScore(train_model_predict(test_data_mice_m5, train_mean_mice_coef),
          ifelse(test_data_mice_m5$outcome == 1, 1, 0))

# create tables for the values
aucs <- c(full_auc, test_auc)
rmses <- c(full_rmse, test_rmse)
bss <- c(full_bs, test_bs)
model_names <- c("Full Data Model", "Train Data Model")
eval_table <- data.frame(model_names, aucs, rmses, bss)
colnames(eval_table) <- c("", "AUC", "RMSE", "Brier")
eval_table
```