Predicting the Need for Tracheostomy in Infants with Severe Bronchopulmonary Dysplasia

Author: Wanyi Chen Date: Nov 12, 2023

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 Collaboratice 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 inspiratiory 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.

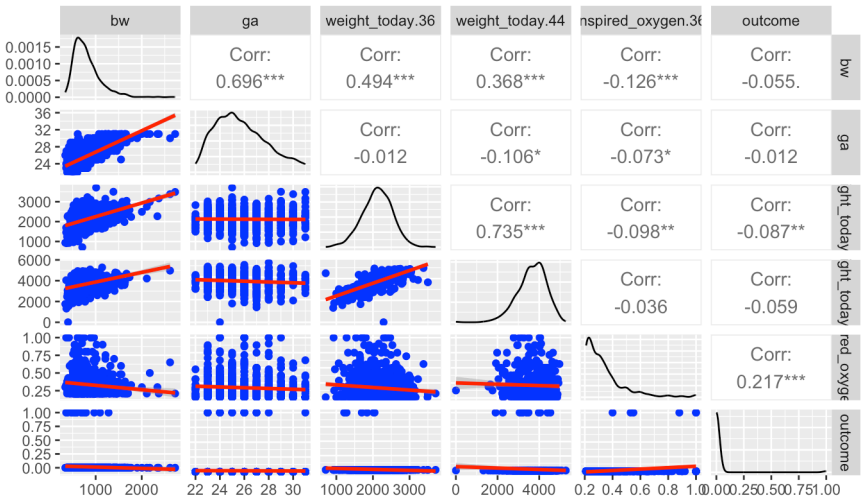

Figure 1: Correlations of selected variables

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.

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.

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

	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 ventilation.44	43.47	2.51	29.62	0.10	4.42	0.50	2.61	1.20	2.21	0.2	0.1
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
compre natster p delta.36	19.38	1.61	10.54	1.00	1.71	0.30	0.90	2.61	0.50	0.2	0.0
hosp dc ga	12.85	2.01	3.92	0.70	1.51	1.31	0.10	3.21	0.00	0.1	0.0
	12.45	6.43	0.00	0.00	6.02	0.00	0.00	0.00	0.00	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
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

Model Derivation

The Lasso method is used for variable selection. Lasso has an equivalent penalty form, which can be expressed as minimizing

$$\|Y - X\beta\|^2 + \lambda\|\beta\|_1$$

, where λ is a regularization parameter that controls the strength of the penalty applied to the l_1 norm of the coefficients. In this penalty form, larger values of λ may shrink some coefficients to zero, effectively performing variable selection. This property makes the lasso method 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.

The second column of Table 3 shows the coefficient estimates of the final model obtained by fitting the 5 lasso models using the 5 imputation data. There are 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 model, 22 out 26 variables (record_id, Trach, and Death are not included in the variable list, and mat_race is removed due to inconsistency of data coding) are selected. Out of coefficient estimates in the final model, the contunous variable inspired_oxygen.36 has the highest effect and a coefficient value of 0.246451. This means, on average, an additional unit increase in fraction of inspired oxygen at 36 weeks will result in a 0.246451 unit increase in the outcome (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 inspired_oxygen.36 implies that at 36 weeks, improper ventilation used at this time period would cause increased risk of death associated with tracheostomy. The second highest coefficient estimate is Center 12 which

also has the high frequency score of 5. This means, on average, if a patient is from Center 12 then the outcome is 0.077707 higher relative to other centers and adjusting for other variables in the model. However, this does not necessary mean Center 12 has worse outcomes, because the data is unbalanced and the characteristics of patients in each center can be very different from between other centers.

The third and fifth highest coefficient estimates are from Center 2 and Center 3 which again does not necessary mean they have worse outcomes due to the differences between centers. The fourth highest coefficient estimates is inspired_oxygen.44 which means, on average, an additional unit increase in fraction of inspired oxygen at 44 weeks will result in a 0.012669 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 are birth head circumference, complete prenatal steroids, maternal chorioamnionitis, gender, weight at 36 weeks, peak inspiratory pressure (cmH2O) at 36 weeks, medication for pulmonary hypertension at 36 weeks, peak inspiratory pressure (cmH2O) 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.

To investigate the generalizability and to evaluate the final model, cross validation is performed. 5 train data sets are imputed using the train data (75% or 747 observations randomly sampled from the original data set) and 5 test data sets are imputed using the test data (25% or 249 observations randomly sampled from the original data set). Then, the same model derivation process is followed using the lasso method. The coefficient estimates for the train data model is shown in column 4 of Table 3. Since less information is considered in the train data models, the final train data model has much fewer variables selected, only 16 out of 26 variables are being selected. In this case, Center 12 has the highest coefficient estimate of 0.113631 as well as a high frequency of 5. The second highest coefficient estimate is inspired_oxygen.44 (estimate = 0.081330), which means on average, an additional unit increase in fraction of inspired oxygen at 44 weeks will result in a 0.081330 unit increase in the outcome while adjusting for other predictors in the model. The third highest coefficient estimate is inspired_oxygen.36 (estimate = 0.026164), which means on average, an additional unit increase in fraction of inspired oxygen at 36 weeks will result in a 0.026164 unit increase in the outcome while adjusting for other predictors in the model.This finding again implies the importance proper ventilation for patients at week 35 and week 44.

Table 3: Model coefficient estimates and their frequencies in the 5 imputed data sets using full imputation data and train imputation data.

Coefficient	Coefficient Estimate (Full Data)	Coefficient Frequency (Full Data)	Coefficient Estimate (Train Data)	Coefficient Frequency (Train Data)
(Intercept)	-0.030183	5	-0.010482	5

Coefficient	Coefficient Estimate (Full Data)	Coefficient Frequency (Full Data)	Coefficient Estimate (Train Data)	Coefficient Frequency (Train Data)
center2	0.017161	5	0.001083	2
center3	0.011886	4	0.000000	0
center5	-0.000003	1	0.000000	0
center7	-0.002393	1	0.000000	0
center12	0.077707	5	0.113631	5
center16	-0.001097	2	0.000000	0
center21	0.002325	1	0.000000	0
bw	0.000006	1	0.000000	0
blength	-0.000691	4	-0.000091	2
birth_hc	-0.001323	5	0.000000	0
del_method2	0.002047	3	0.000106	2
prenat_ster2	-0.005775	4	-0.000790	2
com_prenat_ster2	-0.006262	5	0.000000	0
mat_chorio2	0.006310	5	0.000000	0
gender2	-0.008317	5	-0.004890	5
any_surf2	0.010880	2	0.000645	1
weight_today.36	-0.000009	5	-0.000007	2
ventilation support level.361	0.000197	1	-0.004363	3
ventilation support level.362	0.027540	3	0.000337	1
inspired_oxygen.36	0.246451	5	0.026164	3
p_delta.36	-0.002094	5	0.000000	0
peep cm h2o modified.36	-0.000056	1	0.000021	2
med_ph.361	-0.013834	5	0.002422	2
weight_today.44	0.000001	2	0.000000	0
ventilation support level modified.441	0.000000	0	-0.000449	1
ventilation support level modified.442	-0.003695	1	0.011064	3
inspired_oxygen.44	0.012669	4	0.081330	5
p_delta.44	0.001459	5	0.000461	3
med_ph.441	0.000446	1	0.000000	0
hosp_dc_ga	-0.000037	4	-0.000019	2

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 of 17% which is the middle percentage in that range. The RMSE value is also low and very close to 0 which suggests ideal prediction accuracy. For Brier score, a score of 0 represents perfect accuracy and a score of 0.25 is the same as a chance. The Brier score in this case is 0.0129 which is lower than 0.25 and very close to 0. This indicates better calibration of predicted probability of the outcome, and that the predicted probability increasingly equals to the observed probability.

Even though the scores for the full data model is fairly ideal, the train data model has much lower AUC score. The AUC score is only 0.4980 which means the prediction accuracy is the same as randomly guessing. Besides AUC, the RMSE value is also slightly higher and the Brier score also increased by more than 0.01. The low score of AUC can also be related to the threshold used in calculation since mortality rate in severe BPD studies has a wide range due differences in settings. The relatively worse evaluation results in the train data model suggests that there are still much room for improvement of the model's generalizability and the model derivation process.

Table 4: AUC, RMSE, and Brier scores are calculated using the actual values and predicted values in the full data set, and train and test sets.

	AUC	RMSE	Brier
Full Data Model	0.8042572	0.1194028	0.0128810
Train Data Model	0.4979508	0.1552301	0.0242956

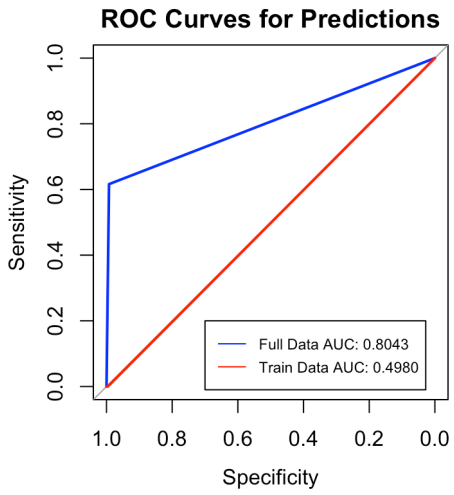

Figure 2: ROC curve for the full data set and train data set predictions

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 model of this project performs fairly ideal with the original data sets but still needs improvement on generalizability. In this project, Lasso is the only method used in the variable selection process, in future studies, other methods such as forward-stepwise regression and best subset selection can also be performed to get more accurate and meaningful predictions. There are also limitations cause by the missingness and imbalances of the original 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 the more methods (Lasso, Forward-Stepwise, and Best Subset Selection, etc) are performed and more data sets are imputed.

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.

Appendix

Table 5: 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
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

	Overall missingness	Center 1	Center 2	Center 3	Center 4	Center 5	Center 7	Center 12	Center 16	Center 20	Center 21
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 pre-nat 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
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 modi-fied.36	11.75	2.41	4.12	1.10	0.60	0.00	0.10	3.41	0.00	0.0	0.0
weight to-day.36	9.24	1.71	3.61	0.30	0.60	0.00	0.10	2.91	0.00	0.0	0.0
inspired oxy-gen.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
ventilation sup-port level.36	3.01	0.10	0.90	0.10	0.00	0.00	0.00	1.91	0.00	0.0	0.0
med 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

	Overall missingness	Center 1	Center 2	Center 3	Center 4	Center 5	Center 7	Center 12	Center 16	Center 20	Center 21
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)
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)))
```

```r
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)
```

```r
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
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
```

```r
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))
```

```r
# 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))
```

```r
# 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)
```

```r
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)
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 %>%
```

```r
  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
```

```r
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
```