PHP2550

PROJECT 2

2023-11-12

Model to Predict Tracheostomy Placement in Neonates

ABSTRACT

Background: In recent past extended ventilation due to severe bronchopulmonary dysplasia (BPD) has risen as more infants survive severe BPD , making it the primary reason for tracheostomy in infants under the age of one year (Akangire & Manimtim, 2023). The timing for tracheostomy placement has been an ongoing discussion with previous studies showing its benefits on growth (Akangire & Manimtim, 2023). The criteria and timing for tracheostomy in neonates with sBPD differs significantly among centers and their timings are currently unclear in pediatric care. The existing methods for predicting tracheostomy placement have shown an accurate prediction of the likelihood of tracheostomy placement based on baseline demographics and clinical diagnoses but lacks details on respiratory parameters and fail to provide predictions at various postmenstrual ages (PMA).This project aims to develop regression models that predicts the outcome of tracheostomy to inform the criteria for its indication and optimal timing of tracheostomy placement.

Method: This project utilizes data collected from the BPD Collaborative Registry, where interdisciplinary BPD programs from the United States and Sweden collaborate to bridge knowledge gaps and improve care for children with severe BPD. Data were collected on respiratory parameters and demographic information at 36 and 44 weeks. The project utilizes a trained generalized linear mixed-effects regression approach for predicting tracheostomy placement. To enhance variable selection, Lasso and Ridge regression methods were compared. The variables selected by Lasso were then used to develop the predictive model.The model was trained and evaluated using the same dataset. The model's performance were evaluated using metrics such as sensitivity, specificity, and the area under the receiver operating characteristic curve (AUC).

Results: Data with 995 observations was used in developing the predictive model using data collected at two different time points. On the test data, the model achieved an AUC of about 0.9 demonstrating a sensitivity of 0.55 and specificity of 0.98. The overall accuracy of the model on the test set was 0.91.

Conclusion: This predictive model is the first predictive model for timing of tracheostomy placement in infants with sBDP. These findings underscore the model's capacity to distinguish between positive and negative cases and shows its accuracy in identifying infants who truly need a tracheostomy and those whom immediate intervention might not be required. The model's overall accuracy further shows its reliability for informing clinical decision-making. Future work may involve evaluating the model using a different data source.

Introduction

Tracheostomy placement in infants with severe bronchopulmonary dysplasia (sBPD) is a critical intervention with great implications for both medical management and long term outcomes (Akangire & Manimtim, 2023). As advancements in neonatal care have led to increased survival rates among infants with severe BPD, the need for prolonged ventilation has become a great indication for tracheostomy in this vulnerable population (Akangire & Manimtim, 2023). In pediatric healthcare, deciding when to do a tracheostomy in

infants is crucial and the exact indication criteria and timing of tracheostomy placement in neonates with severe bronchopulmonary dysplasia (sBPD) remains a challenge. The decision making process surrounding the timing of tracheostomy placement remains complex, with ongoing debates regarding optimal criteria and potential benefits, particularly for infants with severe forms of BPD. Previous studies have shown the importance of early tracheostomy placement in influencing positive growth outcomes (Cheung & Napolitano, 2014). However, despite these considerations, precise predictive models to guide the optimal timing of tracheostomy in infants with severe BPD remain a critical area of research. Dr. Chris Schmid's project aims at solving the question surrounding when to perform tracheostomy in neonates with severe bronchopulmonary dysplasia (sBPD). Analyses done in the past have successful predicted the likelihood of a tracheostomy placement based on the baseline patient characteristics and clinical diagnoses but none have utilized the respiratory parameters which may provide accurate predictions for the need for tracheostomy during early postmenstrual ages (PMA) which could greatly help in advising on the timing of the procedure.

Methods

The data utilized in the analysis is sourced from a national database and comprises demographic details, diagnostic information, and respiratory parameters of infants diagnosed with sBPD. These infants were admitted to collaborative NICUs, and the dataset specifically includes data on their respiratory support status at 36 weeks and 44 weeks post menstrual age (PMA). The analysis used 995 records of data collected both at 36 and 44 weeks. Minimal data preprocessing was performed before the analysis. Duplicated records were removed, outlier points as observed in `hosp_dc_ga` were normalized on z scale and variables were renamed to more meaningful naming. Multiple Imputation by Chained Equations (MICE) package in R was used to handle missing data. Binary variables were adjusted to the appropriate format to enhance consistency and interpretation of the data. Data for 36 weeks and 44 weeks were handled independently.

Missing Data

Missingness were inspected for in each variable in the two datasets. There were notable differences in missingness between 36 and 44 weeks data.

The observed missingness pattern at 36 weeks is likely missing not ar random (MNAR),and that missingness is not completely random and may be influenced by other factors that were not measured in the study, `any_surf` had an abnormally different value for missingness . Contrary, the missingness at 44 weeks is likely to be missing at random (MAR). There is a notable pattern in missingness in some of the variables (`inspired_oxygen_44wks`, `peak_delta_44wks`, `weight_44`, `peep_cmH20_44`, `any_surf`, `med_ph_44` and `vent_support_44`) which indicates that the absence of data in these variable is related to the values of other measured covariates. We however noted that these variables were less correlated but there seem to be a fair correlation in the respiratory parameters as noted in the correlation figure. The variables in the summary table are normalized to a common scale.

Table 1: Missing Data at 36weeks

Variable	n	% Proportion
any_surf	433	43.52
comp_prenat_steroids	193	19.40
peak_delta_36wks	128	12.86
hosp_dc_ga	124	12.46
peep_cmH2o_mod_36wks	117	11.76
weight_36wks	92	9.25
inspired_oxygen_36wks	92	9.25
blength	78	7.84
birth_hc	77	7.74
mat_chorio	62	6.23
mat_ethn	57	5.73
mat_race	56	5.63
prenat_steroids	35	3.52
ventilation_support_36wks	30	3.02
med_ph_36wks	30	3.02
sga	15	1.51
center	10	1.01
deliv_method	3	0.30
Death	2	0.20

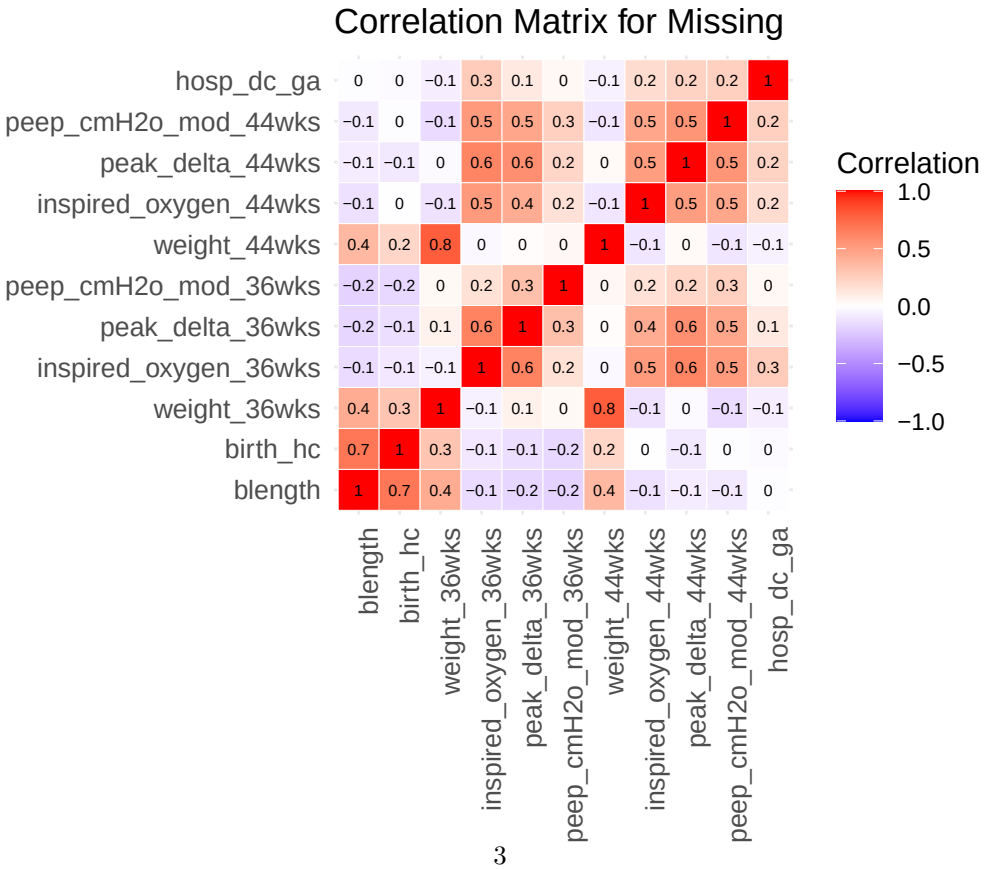


Table 2: Missing Data at 44weeks

Variable	n	% Proportion
inspired_oxygen_44wks	447	44.92
peak_delta_44wks	447	44.92
weight_44wks	445	44.72
peep_cmH2o_mod_44wks	445	44.72
any_surf	433	43.52
vent_support_level_mod_44wks	423	42.51
med_ph_44wks	423	42.51
comp_prenat_steroids	193	19.40
hosp_dc_ga	124	12.46
blength	78	7.84
birth_hc	77	7.74
mat_chorio	62	6.23
mat_ethn	57	5.73
mat_race	56	5.63
prenat_steroids	35	3.52
sga	15	1.51
center	10	1.01
deliv_method	3	0.30
Death	2	0.20

Table 3: Summary Statistics

Variable	Percentage Missing	Mean	SD	Min	Max
bw	0	806.10	296.77	340.00	2725.00
ga	0	25.77	2.14	22.00	31.00
blength	8	32.49	3.82	18.00	48.00
birth_hc	8	23.19	2.76	13.50	38.30
weight_36wks	9	2120.90	413.58	710.00	3710.00
inspired_oxygen_36wks	9	0.34	0.15	0.21	1.00
peak_delta_36wks	13	5.27	9.74	0.00	46.00
peep_cmH2o_mod_36wks	12	6.33	2.91	0.00	18.00
weight_44wks	45	3646.12	682.09	3.00	5275.00
inspired_oxygen_44wks	45	0.34	0.15	0.21	1.00
peak_delta_44wks	45	7.62	14.19	0.00	52.00
peep_cmH2o_mod_44wks	45	4.30	4.46	0.00	20.00
hosp_dc_ga	12	0.46	0.89	0.00	19.64

Imputation Method For Missing Data

The Mice package was used to do multiple imputations. Five complete datasets with imputed values were created for both (36 and 44 weeks) variables that were used in the analysis.

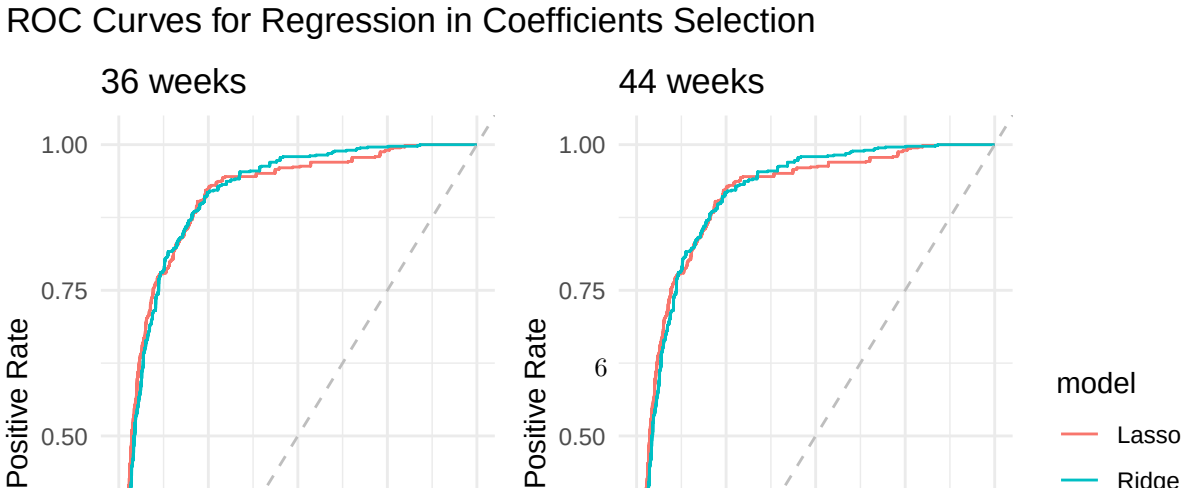
Variable Selection

Variables used in the analysis were selected using the Lasso and Ridge regression approaches to identify the most influential predictors in the predictive model. A 10 -fold cross-validation procedure was employed for evaluating the performance of lasso and ridge regression models in each imputed dataset. Data were partitioned into ten folds, with nine folds being used for training and one for testing the model in each iteration over 10 iterations with each fold being used as the test set exactly once. Using the `cv.glmnet` function in R, both lasso and ridge regressions were applied to each imputed dataset, with lasso model being regularized using the L1 norm penalty and alpha set to 1, establishing a pure lasso model while the ridge regression model was regularized using the L0 norm penalty and alpha set to 0 . Cross-validation served as a penalty to identify the optimal lambda value that minimized the cross-validation error, guiding the selection of coefficients for the predictive model. An appropriate lambda in lasso shrinks variable coefficients to zero when there is no association with the outcome while ridge shrinks coefficients towards zero but does not force them to be exactly zero. The optimal lambda was determined for each imputed set, the models were refitted to each full imputed dataset, and the coefficients were extracted for subsequent analysis. An aggregate for the coefficients from each imputed dataset were averaged to find the coefficients that were used in predictive modelling.

Lasso selects 16 predictors with non-zero coefficients from 36 weeks data and 10 predictors from weeks 44 data. Inclusion of `center`, `maternal ethnicity`, `delivery method`, `prenatal steroids`, `maternal chorioamnionitis`, `any surfactant use`, `ventilation support`, ‘`peep_cmH2o modification`,`hosp_dc_ga` and `medication administration for pulmonary hypertension` predictors are observed in both 36 weeks and 44 weeks.These coeffiecients were included in developing the predictive models.

Table 4: Estimated Coefficients

	36wks Lasso	36wks Ridge	44wks Lasso	44wks Ridge
(Intercept)	-4.1321	-5.3640	-0.8075	-3.9487
center2	-0.3344	-0.6374	-0.1512	-0.7310
center3	-1.3652	-1.8810	-0.3917	-1.5325
center4	-0.0061	-0.2043	0.0000	-0.9318
center5	-0.0837	-0.2344	0.0000	-0.0104
center7	-0.1949	-0.8486	-0.3196	-1.4043
center12	1.1472	1.2519	0.2745	1.4694
center16	-0.2826	-0.9532	-0.0872	-1.0004
center20	-0.3868	-1.7783	0.0000	-1.3334
center21	1.9712	3.6211	0.6039	4.7811
mat_ethn2	0.2335	0.5911	0.0229	0.4970
bw	0.0000	0.0001	0.0000	-0.0001
ga	0.0053	0.0044	0.0000	-0.0412
blength	0.0000	0.0049	0.0000	0.0013
birth_hc	0.0084	0.0339	0.0000	0.0411
deliv_method1	-0.1228	-0.2199	-0.0263	-0.2303
prenat_steroids1	0.8256	1.0373	0.2187	1.0107
comp_prenat_steroidsYes	0.0352	0.1724	0.0412	0.4243
mat_chorioYes	0.0000	-0.1411	0.0000	0.0438
sexFemale	0.0210	0.0893	0.0000	0.0482
sexMale	0.0000	-0.0539	0.0000	-0.0103
sga1	0.0000	-0.0582	0.0000	0.1344
any_surf1	0.0841	0.2027	0.0222	0.0514
weight_36wks	-0.0001	-0.0003	0.0000	-0.0001
ventilation_support_36wks1	-0.0894	-0.6324	0.0000	-0.1708
ventilation_support_36wks2	1.0164	0.8073	0.2488	1.1144
inspired_oxygen_36wks	1.8162	2.0808	0.0000	-0.1981
peak_delta_36wks	0.0005	0.0032	0.0000	0.0144
peep_cmH2o_mod_36wks	0.0213	0.0734	0.0234	0.1056
med_ph_36wks1	0.0647	0.1862	0.1196	0.4792
hosp_dc_ga	0.5725	0.5417	0.1294	0.4709



Model Derivation

Multilevel Logistic Regression Model

In the development of the predictive model for tracheostomy placement, the analysis centered on the imputed dataset obtained through the Multiple Imputation by Chained Equations (MICE) approach. Transforming the imputed dataset into a long format facilitated subsequent steps in model development. To evaluate the model's performance, we implemented a train-test split, allocating 70% of the data for training and the remaining 30% for testing. We employed a mixed effects logistic regression model utilizing the generalized logistic mixed-effects regression approach, introducing a random intercept for the center variable and a random slope `for hosp_dc_ga` variable on the assumption that gestational age at discharge varied across centers. Variables drawn from Lasso regression models were included as fixed effects in both 36 and 44 weeks models. The 36weeks model included twice more predictors than the 44weeks model (`birth_hc`, `sex`, `any_surf`, `weight`, `inspired_ocygen`). The model was trained on the training set and subsequent evaluation on the test set, offering predictions based on a predefined threshold of 0.5 for the probability of tracheostomy placement.

The mixed effects model at 36 weeks is represented with the equation below. The model includes the fixed effects and random effects. The Fixed effects contains labeled, the values of β_i and u_j as the random slope for hospital discharge age which is presumed to be varying across the hospitals in the different centers.

$$\begin{aligned}\text{logit}(\Pr(\text{Trach} = 1)) =& \beta_0 + \beta_1 \times \text{mat\_ethn} + \beta_2 \times \text{birth\_hc} + \beta_3 \times \text{deliv\_method} \\ &+ \beta_4 \times \text{prenat\_steroids} + \beta_5 \times \text{comp\_prenat\_steroids} + \beta_6 \times \text{sex} + \beta_7 \times \text{any\_surf} \\ &+ \beta_8 \times \text{weight\_36wks} + \beta_9 \times \text{ventilation\_support\_36wks} + \beta_{10} \times \text{inspired\_oxygen\_36wks} \\ &+ \beta_{11} \times \text{hosp\_dc\_ga} + \beta_{12} \times \text{med\_ph\_36wks} + \beta_{13} \times \text{peep\_cmH2o\_mod\_36wks} \\ &+ u_{0\text{center}} + u_{1\text{center}} \times \text{hosp\_dc\_ga}\end{aligned}$$

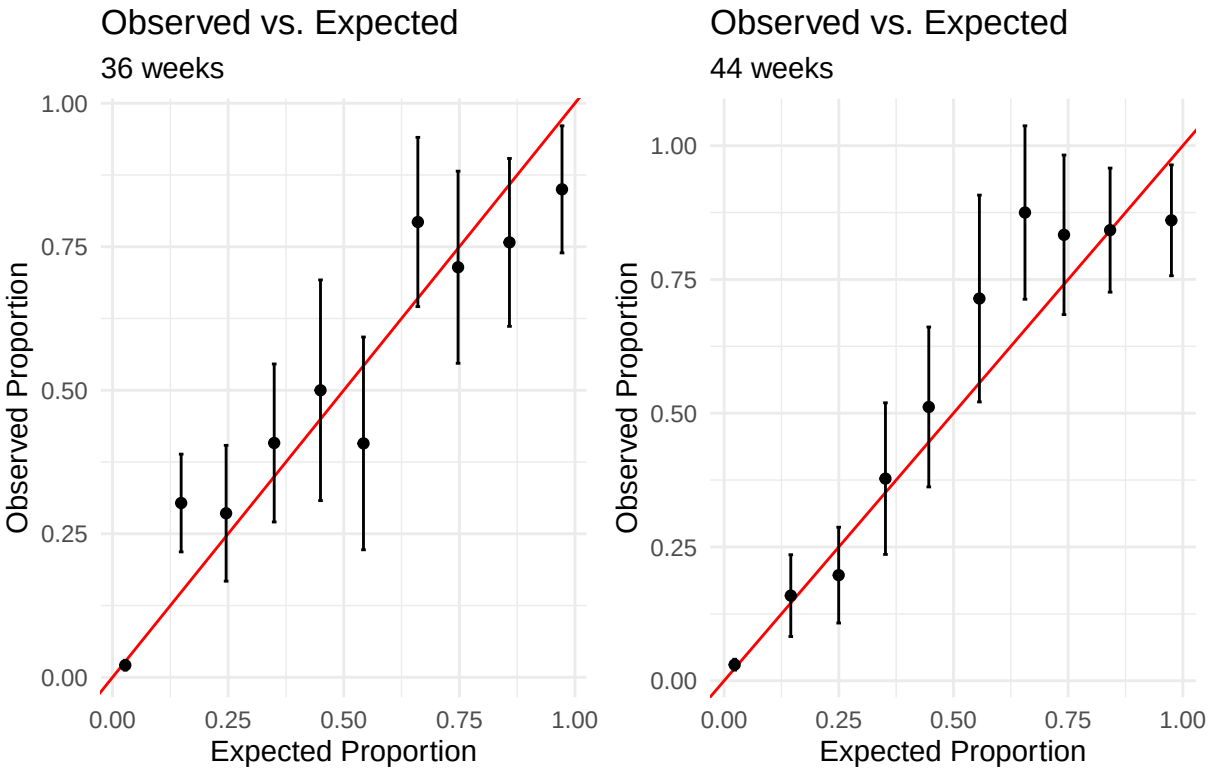
Model Validation

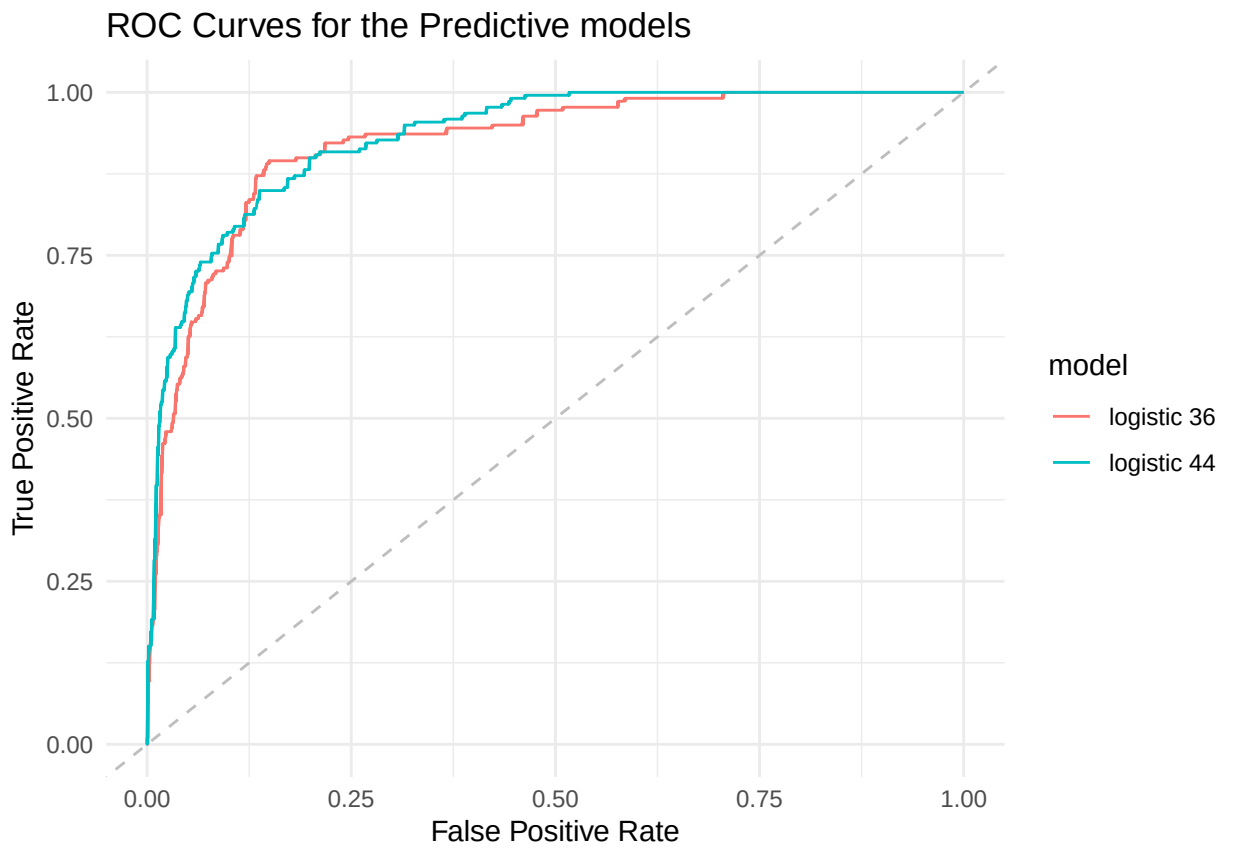
Table 5: Measures of discrimination and calibration

Metric	Model for 44wks	Model for 36wks
Accuracy	0.9162198	0.8994638
AUC	0.9312450	0.9206993
Brier Score	0.0657296	0.0720105
F1 Score	0.6537396	0.6010638
Precision	0.8309859	0.7197452
Sensitivity	0.5388128	0.5159817
Specificity	0.9811469	0.9654360

On the test data, there were no major differences between the predictive model's for predicting tracheostomy placement using 36 weeks and 44 data. The 36 weeks model achieved slightly highere metrics compared to the 44 weeks model as indicated in Table 4.

Calibration Plot −Multilevel Logistic Regression Model





Discussion

Interpretation of results

The predictive multilevel logistics model showed a balanced performance, with an AUC of (0.9380 and 0.9293), demonstrating a good discrimination ability. Sensitivity was measured at 0.5251 and 0.5707, suggesting that the models are effective in identifying infants who really need tracheostomy placement. High specificity, with a value of 0.97, showed that the models are capable to correctly identify infants who do not require tracheostomy placement. The overall accuracy of the model on the test data was 0.55, emphasizing the models reliability in making correct predictions, With a precision measure at 0.8156. Also had lower Brier score (0.0646 and 0.0645) indicating the best calibration. Based on the above, the multilevel logistics regression models show better performance.Plots of the observed versus predicted values for each model showed that the line of best fit is quite accurate in predicting the timing and indication in the proportion of expected infants tracheostomy placement against the observed proportion of those requiring tracheostomy on average as seen in calibration figure. However there is still room for improvement as they do not perform optimally. Overall, these findings showed the potential of the predictive model to assist clinicians in making informed decisions about the timing of tracheostomy placement in infants, contributing to improved patient care and resource management.

One strength in this project is that the included predictors for each model had limited overlap which shows that both models (36weeks and 44 weeks) could be used to predict the timing for tracheostomy placement among. Also the missingness was Overall, the models performed quite well but not optimal.

Limitations

This project has quite a number of limitations. First the data were collected at different centers, with some of the centers having missing data. While there were no huge numbers of missingness, this limited our ability to develop predictive models for each center due to limited data. Also, there was limited observations for the two potential outcomes which may have limited our analysis. To help increase the sample size for the outcome, a composite outcome variable would have been constructed by combining `death` and `tracheostomy` but these outcomes seem to be very different and would limit the model's applicability/generalization. Additionally, missingness varied in the two time points when the data collected and this would have been as a result of the multicenter different data collection periods. Certain variables such as `any_surf` which may be strong predictors for the outcome had the largest percentage of missingness. Lastly, the model was built and trained on the same dataset and this may limit its generalizability to other populations. Additional future work will need to evaluate the model on an external dataset.

Conclusion

This project puts forth predictive models from infants with severe bronchopulmonary dysplasia from centers in the United States and Sweden. Based on lasso regression models, the measurements of several respiratory measures at two different time points allows for the prediction of the timing criteria for tracheostomy placements. These are among the first models that can help decide when a tracheostomy is required in infants with sBDP. The clinical application of these models is to aid in determining the optimal timing for tracheostomy placement in infants. We hope that the model's performance, in identifying infants who truly need a tracheostomy and those whom an immediate intervention may not be required, will be valuable tool for clinical decision making and management of infants which could potentially improve the care of infants needing tracheostomy placement and contribute to better healthcare outcomes and resource management.

References

1. Akangire G, Manimtim W. Tracheostomy in infants with severe bronchopulmonary dysplasia: A review. Front Pediatr. 2023 Jan 12;10:1066367. doi: 10.3389/fped.2022.1066367. PMID: 36714650; PMCID: PMC9878282

2. Cheung, N. H., & Napolitano, L. M. (2014). Respiratory Care. Respiratory Care, 59(6), 895-919. https://doi.org/10.4187/respcare.02971

Supplemental Information

Github page: The code that was used in the analysis can be found on github https://github.com/kasigi234/PHP2550-Project2

Code Appendix

```
# load library
library(mice)
library(gtsummary)
library(tidyverse)
library(glmnet)
library(leaps)
library(kableExtra)
library(knitr)
library(pROC)
library(bestglm)
library(latex2exp)
library(ggplot2)
library(tableone)
library(Matrix)
library(MASS)
library(tidyr)
library(naniar)
library(dplyr)
library(lattice)
library(reshape2)
library(formatR)
library(lme4)
library(caret)
library(patchwork)
library(cowplot)

# load the data
trach_df = read.csv("project2.csv")

#check for duplicated IDs
dup_ids <- duplicated(trach_df$record_id) | duplicated(trach_df$record_id, fromLast = TRUE)

# rows with duplicated IDs
dup_entries <- trach_df[dup_ids, ] #789, 790, 791, 792

# keep only unique records
trach <- trach_df[!dup_ids, ] #995 30

# Select variables and rename
trach <- trach %>% #dplyr::select(-record_id, -center) %>%
  rename(prenat_steroids = prenat_ster,
         comp_prenat_steroids = com_prenat_ster,
         sex = gender, weight_36wks = weight_today.36,
         peep_cmH2o_mod_36wks= peep_cm_h2o_modified.36,
         weight_44wks = weight_today.44,med_PH_44wks = med_ph.44,
         peep_cmH2o_mod_36wks = peep_cm_h2o_modified.36,
         deliv_method = del_method,
         ventilation_support_36wks = ventilation_support_level.36,
         inspired_oxygen_36wks = inspired_oxygen.36,
         peak_delta_36wks = p_delta.36, med_ph_36wks = med_ph.36,
         vent_support_level_mod_44wks=ventilation_support_level_modified.44,
```

```
         peep_cmH2o_mod_44wks = peep_cm_h2o_modified.44,
         peak_delta_44wks = p_delta.44, med_ph_44wks = med_ph.44,
         Tracheostomy = Trach, inspired_oxygen_44wks = inspired_oxygen.44)

trach$deliv_method <- case_when(trach$deliv_method == 1 ~ 1,
                                trach$deliv_method == 2 ~ 0)
trach$prenat_steroids <- ifelse(trach$prenat_steroids == "No", 0,
                          ifelse(trach$prenat_steroids == "Yes", 1,
                                 trach$prenat_steroids))

trach$sex <- case_when(trach$sex == 1 ~ "1",
                         trach$sex == 2 ~ "0",
                         is.na(trach$sex) ~ "Ambiguous",
                         TRUE ~ as.character(trach$sex))

trach$sga <- ifelse(trach$sga == "Not SGA", 0,
                    ifelse(trach$sga == "SGA", 1, trach$sga))

trach$any_surf <- ifelse(trach$any_surf == "No", 0,
                           ifelse(trach$any_surf == "Yes", 1,
                                  trach$any_surf))

# create a composite outcome variable from Death and Tracheostomy
trach$Death <- ifelse(trach$Death == "No", 0,
                          ifelse(trach$Death == "Yes", 1,
                                 trach$Death))

trach$Tracheostomy <- ifelse(trach$Tracheostomy == "No", 0,
                         ifelse(trach$Tracheostomy == "Yes", 1,
                                 trach$Tracheostomy))

#trach$comp_outcome <- with(trach,
                     #   ifelse(Death == 1 | Tracheostomy ==
                       #            1,1, 0))

# Reorder levels of the outcome variable with "1" as the reference level
#trach$comp_outcome <- factor(trach$comp_outcome, levels = c("1", "0"))

#trach$comp_outcome <- as.integer(trach$comp_outcome)
trach$deliv_method <- as.integer(trach$deliv_method)

# All numeric
trach$bw <- as.numeric(trach$bw)
trach$blength <- as.numeric(trach$blength)
trach$weight_36wks <- as.numeric(trach$weight_36wks)
trach$weight_44wks <- as.numeric(trach$weight_44wks)
trach$peep_cmH2o_mod_36wks <- as.numeric(trach$peep_cmH2o_mod_36wks)
trach$peep_cmH2o_mod_44wks <- as.numeric(trach$peep_cmH2o_mod_44wks)
trach$ga <- as.numeric(trach$ga)


# All factors
fact <- function(data) {
```

```
    categorical_vars <- sapply(data, function(x) is.character(x) || is.integer(x))
    data[categorical_vars] <- lapply(data[categorical_vars], as.factor)

  return(data)
}
trach <- fact(trach)

# Standardize hosp discharge gestational age
trach$hosp_dc_ga <- as.numeric(abs(scale(trach$hosp_dc_ga)))
#trach$weight_36wks <- as.numeric(abs(scale(trach$weight_36wks)))
#trach$weight_44wks <- as.numeric(abs(scale(trach$weight_44wks)))

#View(trach)

# separate data collected at 44 weeks and 36 weeks
cols_36wks <- grep("36wks", colnames(trach), value = TRUE)
cols_44wks <- grep("44wks", colnames(trach), value = TRUE)

# data for "36wks"
trach_36wks <- trach[, -which(colnames(trach) %in% cols_44wks)]

# data for "44wks"
trach_44wks <- trach[, -which(colnames(trach) %in% cols_36wks)]


# Distribution of Missing Data at 36 weeks
missing_df <- data.frame(
  Variable = names(trach_36wks),
  missing_count = sapply(trach_36wks, function(x) sum(is.na(x))))

missing_df$percent_missing <- round(missing_df$missing_count / nrow(trach_36wks) * 100, 2)

missing_df <- missing_df %>%
  arrange(desc(percent_missing))

# Remove row names
rownames(missing_df) <- NULL

# Select only those with missing records
missing_df <- missing_df %>%
  filter(missing_count > 0)

missing_df$missing_count <- round(missing_df$missing_count, 2)
missing_df$percent_missing <- round(missing_df$percent_missing, 2)

  missing_df %>%
  mutate() %>%
  mutate_all(linebreak) %>%
  kbl(caption = "Missing Data at 36weeks",
  col.names = linebreak(c("Variable", "n", "% Proportion")),
  booktabs = T, escape = T, align = "c") %>%
  kable_styling(full_width = FALSE,
                latex_options = c('hold_position'),
```

```
                font_size = 12) #%>%
  #row_spec(0, background = "#333333", color = "white")

# Distribution of Missing Data at 44 weeks
missing_df2 <- data.frame(
  Variable = names(trach_44wks),
  missing_count = sapply(trach_44wks, function(x) sum(is.na(x))))

missing_df2$percent_missing <- round(missing_df2$missing_count / nrow(trach_44wks) * 100, 2)

missing_df2 <- missing_df2 %>%
  arrange(desc(percent_missing))

# Remove row names
rownames(missing_df2) <- NULL

# Select only those with missing records
missing_df2 <- missing_df2 %>%
  filter(missing_count > 0)

missing_df2$missing_count <- round(missing_df2$missing_count, 2)
missing_df2$percent_missing <- round(missing_df2$percent_missing, 2)

  missing_df2 %>%
  mutate() %>%
  mutate_all(linebreak) %>%
  kbl(caption = "Missing Data at 44weeks",
  col.names = linebreak(c("Variable", "n", "% Proportion")),
  booktabs = T, escape = T, align = "c") %>%
  kable_styling(full_width = FALSE,
                latex_options = c('hold_position'),
                font_size = 12)

# Check to see how correlated the missing data
var_missn <- trach[, colSums(is.na(trach)) > 0]
num_data <- var_missn[sapply(var_missn, is.numeric)]
corr_mat <- cor(num_data)

# Compute correlation matrix
cor_matrix <- cor(num_data, use = "complete.obs")

# Melt the correlation matrix for ggplot
cor_df <- melt(cor_matrix)

# Corr Matrix for missing data
ggplot(data = cor_df, aes(x = Var1, y = Var2)) +
  geom_tile(aes(fill = value), color = "white") +
  geom_text(aes(label = round(value, 1)), size = 2) +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1,1), space =
                         "Lab", name="Correlation") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, vjust = 1,
```

```
                                    size = 10, hjust = 1),
        axis.text.y = element_text(size = 10)) +
  coord_fixed() +
  labs(x = NULL, y = NULL, title=
         "Correlation Matrix for Missing")
# Calculate summary statistics for numeric columns
summary_table <- trach %>%
  dplyr::select(variable = colnames(.)[sapply(., is.numeric)]) %>%
  summarise(
    variable = colnames(trach)[sapply(trach, is.numeric)],
    Missing = round(colSums(is.na(.)) / nrow(.), 2) * 100,
    Mean = round(colMeans(., na.rm = TRUE), 2),
    SD = round(apply(., 2, sd, na.rm = TRUE), 2),
    Min = round(apply(., 2, min, na.rm = TRUE), 2),
    Max = round(apply(., 2, max, na.rm = TRUE), 2))

summary_table %>%
  mutate() %>%
  mutate_all(linebreak) %>%
  kbl(caption = "Summary Statistics",
  col.names = linebreak(c("Variable",'Percentage Missing', "Mean", "SD",
                          "Min","Max")),
  booktabs = T, escape = T, align = "c") %>%
  kable_styling(full_width = FALSE,
                latex_options = c('hold_position'),
                font_size = 12)# %>%
 # row_spec(0, background = "#333333", color = "white")

# Remove variables not in imputation
trach_sub <- trach_36wks[, !colnames(trach_36wks) %in% c("mat_race", "Death", "record_id")]
trach_sub2 <- trach_44wks[, !colnames(trach_44wks) %in% c("mat_race", "Death", "record_id")]

# impute using mice package forboth 36 weeks and 44weeks
trach_df_mice_out <- mice::mice(trach_sub, 5, pri=F)
trach_df_mice_out2 <- mice::mice(trach_sub2, 5, pri=F)

# Store each imputed data set
trach_df_imp <- vector("list",5)
for (i in 1:5){
   trach_df_imp[[i]] <- mice::complete(trach_df_mice_out,i)
}
#trach_df_imp[[1]] # Example of accessing first imputed dataset

# Store each imputed data set
trach_df_imp2 <- vector("list",5)
for (i in 1:5){
   trach_df_imp2[[i]] <- mice::complete(trach_df_mice_out2,i)
}

##########################################################
#### Lasso for 36 weeks ####
##########################################################
```

```r
library(lmerTest)
set.seed(1)
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(Tracheostomy ~., data = df)[,-1]
  y.ord <- df$Tracheostomy

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

   lasso_mod <- glmnet(x.ord, y.ord, nfolds = 10, alpha = 1,
                       family = "binomial", lambda =
                        lasso_mod_cv$lambda.min)
  # Get coefficients
  coef <- coef(lasso_mod) #, lambda = lasso_mod$lambda.min)
  return(coef)
}

# Find average lasso coefficients over imputed datasets
lasso_coef1 <- lasso(trach_df_imp[[1]])
lasso_coef2 <- lasso(trach_df_imp[[2]])
lasso_coef3 <- lasso(trach_df_imp[[3]])
lasso_coef4 <- lasso(trach_df_imp[[4]])
lasso_coef5 <- lasso(trach_df_imp[[5]])
lasso_coef <- cbind(lasso_coef1, lasso_coef2, lasso_coef3,
                    lasso_coef4, lasso_coef5)
avg_coefs_lasso <- apply(lasso_coef, 1, mean)

# Find predicted probabilities on long imputed data (no rounding applied in this case!)
trach_df_long <- mice::complete(trach_df_mice_out,action="long")
x_vars <- model.matrix(Tracheostomy~. , trach_df_long)[,-c(2,3)]
trach_df_long$score <- x_vars %*% avg_coefs_lasso
mod <- glmer(Tracheostomy~score + (1|center), data = trach_df_long, family = "binomial" , control = glm
predict_probs1 <- predict(mod, type="response")
#summary(predict_probs)

#########################################################
#### Ridge for 36weeks data ####
#########################################################
set.seed(2)
ridge <- function(df) {
  #' Runs 10-fold CV for ridge and returns corresponding coefficients
  #' @param df, data set
```

```r
  #' @return coef, coefficients for minimum cv error

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

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

  # Ridge model
  ridge_mod_cv <- cv.glmnet(x.ord, y.ord, nfolds = 10, foldid = folds, alpha = 0, family = "binomial")
  ridge_mod <- glmnet(x.ord, y.ord, nfolds = 10, alpha = 0, family = "binomial", lambda = ridge_mod_cv$

  # Get coefficients
  coef <- coef(ridge_mod)
  return(coef)
}
# Find average ridge coefficients over imputed datasets
ridge_coef1 <- ridge(trach_df_imp[[1]])
ridge_coef2 <- ridge(trach_df_imp[[2]])
ridge_coef3 <- ridge(trach_df_imp[[3]])
ridge_coef4 <- ridge(trach_df_imp[[4]])
ridge_coef5 <- ridge(trach_df_imp[[5]])
ridge_coef <- cbind(ridge_coef1, ridge_coef2, ridge_coef3, ridge_coef4, ridge_coef5)
avg_coefs_ridge <- apply(ridge_coef, 1, mean)


# Find predicted probabilities on long imputed data (no rounding applied in this case!)
trach_df_long$score_ridge <- x_vars %*% avg_coefs_ridge
mod2 <- glmer(Tracheostomy~score_ridge + (1|center), data = trach_df_long, family = "binomial")
predict_probs9 <- predict(mod2, type="response")


##########################################################
#### Lasso for 44 weeks data ####
##########################################################

set.seed(3)
lasso2 <- 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(Tracheostomy ~., data = df)[,-1]
  y.ord <- df$Tracheostomy

  # Generate folds
  k <- 10
  set.seed(123) # consistent seeds between imputed data sets
  folds <- sample(1:k, nrow(df), replace=TRUE)
```

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

   lasso_mod <- glmnet(x.ord, y.ord, nfolds = 10, alpha = 1,
                       family = "binomial", lambda =
                        lasso_mod_cv$lambda.min)
  # Get coefficients
  coef <- coef(lasso_mod) #, lambda = lasso_mod$lambda.min)
  return(coef)
}

# Find average lasso coefficients over imputed datasets
lasso_coef1.2 <- lasso2(trach_df_imp2[[1]])
lasso_coef2.2 <- lasso2(trach_df_imp2[[2]])
lasso_coef3.2 <- lasso2(trach_df_imp2[[3]])
lasso_coef4.2 <- lasso2(trach_df_imp2[[4]])
lasso_coef5.2 <- lasso2(trach_df_imp2[[5]])
lasso_coef2 <- cbind(lasso_coef1.2, lasso_coef2.2, lasso_coef3.2,
                     lasso_coef4.2, lasso_coef5.2)
avg_coefs_lasso2 <- apply(lasso_coef2, 1, mean)

# Find predicted probabilities on long imputed data (no rounding applied in this case!)
trach_df_long2 <- mice::complete(trach_df_mice_out2,action="long")
x_vars2 <- model.matrix(Tracheostomy~. , trach_df_long2)[,-c(2,3)]
trach_df_long2$score_lasso2 <- x_vars2 %*% avg_coefs_lasso2
mod.2 <- glmer(Tracheostomy~score_lasso2 + (1|center), data = trach_df_long2, family = "binomial")# , c
predict_probs1.2 <- predict(mod.2, type="response")
#summary(predict_probs)

##########################################################
#### Ridge for 44 weeks data ####
##########################################################
set.seed(4)
ridge2 <- function(df) {
  #' Runs 10-fold CV for ridge and returns corresponding coefficients
  #' @param df, data set
  #' @return coef, coefficients for minimum cv error

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

  # Generate folds
  k <- 10
  set.seed(123) # consistent seeds between imputed data sets
  folds <- sample(1:k, nrow(df), replace=TRUE)

  # Ridge model
  ridge_mod_cv <- cv.glmnet(x.ord, y.ord, nfolds = 10, foldid = folds, alpha = 0, family = "binomial")
  ridge_mod <- glmnet(x.ord, y.ord, nfolds = 10, alpha = 0, family = "binomial", lambda = ridge_mod_cv$

  # Get coefficients
```

```
  coef <- coef(ridge_mod)
  return(coef)
}
# Find average ridge coefficients over imputed datasets
ridge_coef1.2 <- ridge2(trach_df_imp2[[1]])
ridge_coef2.2 <- ridge2(trach_df_imp2[[2]])
ridge_coef3.2 <- ridge2(trach_df_imp2[[3]])
ridge_coef4.2 <- ridge2(trach_df_imp2[[4]])
ridge_coef5.2 <- ridge2(trach_df_imp2[[5]])
ridge_coef2 <- cbind(ridge_coef1.2, ridge_coef2.2, ridge_coef3.2, ridge_coef4.2, ridge_coef5.2)
avg_coefs_ridge2 <- apply(ridge_coef2, 1, mean)


# Find predicted probabilities on long imputed data (no rounding applied in this case!)
trach_df_long2$score_ridge2 <- x_vars2 %*% avg_coefs_ridge2
mod2.2 <- glmer(Tracheostomy~score_ridge2 + (1|center), data = trach_df_long2, family = "binomial")
predict_probs9.2 <- predict(mod2.2, type="response")

# table to show the coefficients from two models at 36 weeks
table <- cbind(
  "36wks Lasso" = avg_coefs_lasso,
  "36wks Ridge" = avg_coefs_ridge,
  "44wks Lasso" = avg_coefs_lasso2,
  "44wks Ridge" = avg_coefs_ridge2
)
table <- round(table, 4)

table <- as.data.frame(table)

table %>%
  mutate() %>%
  mutate_all(linebreak) %>%
  kbl(caption = "Estimated Coefficients",
  #col.names = linebreak(c("3",'Percentage Missing', "Mean", "SD",
                        #  "Min","Max")),
  booktabs = T, escape = T, align = "c") %>%
  kable_styling(full_width = FALSE,
                latex_options = c('hold_position'),
                font_size = 12)
#########################################################
#### Model evaluation for lasso Ridge at 36 weeks ####
#########################################################
# roc and auc
roc_lasso <- roc(trach_df_long$Tracheostomy,predict_probs1)
roc_ridge <- roc(trach_df_long$Tracheostomy,predict_probs9)

roc_curve_lasso <- data.frame(FPR = 1-roc_lasso$specificities,
                              TPR = roc_lasso$sensitivities)
roc_curve_ridge <- data.frame(FPR = 1-roc_ridge$specificities,
                              TPR = roc_ridge$sensitivities)

roc_data <- rbind(roc_curve_lasso,roc_curve_ridge)
roc_data$model <- c(rep("Lasso",nrow(roc_curve_lasso)),
```

```
                    rep("Ridge",nrow(roc_curve_ridge)))

# plot ROC curves for models that were used in variable selection
p36 <- ggplot(roc_data, aes(x = FPR, y = TPR, color = model)) +
  geom_line() +
  geom_abline(intercept = 0, slope = 1, color = "grey", linetype = "dashed") +
  labs(x = "False Positive Rate", y = "True Positive Rate",
       title = "36 weeks") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5,size = 8)) +
  theme_minimal()

#########################################################
#### Model evaluation for lasso Ridge at 44 weeks ####
#########################################################
# roc and auc
roc_lasso2 <- roc(trach_df_long2$Tracheostomy,predict_probs1.2)
roc_ridge2 <- roc(trach_df_long2$Tracheostomy,predict_probs9.2)

roc_curve_lasso2 <- data.frame(FPR = 1-roc_lasso$specificities,
                               TPR = roc_lasso$sensitivities)
roc_curve_ridge2 <- data.frame(FPR = 1-roc_ridge$specificities,
                               TPR = roc_ridge$sensitivities)

roc_data2 <- rbind(roc_curve_lasso2,roc_curve_ridge2)
roc_data2$model <- c(rep("Lasso",nrow(roc_curve_lasso2)),
                     rep("Ridge",nrow(roc_curve_ridge2)))

# plot ROC curves for models that were used in variable selection
p44 <- ggplot(roc_data2, aes(x = FPR, y = TPR, color = model)) +
  geom_line() +
  geom_abline(intercept = 0, slope = 1, color = "grey", linetype = "dashed") +
  labs(x = "False Positive Rate", y = "True Positive Rate",
       title = "44 weeks") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5,size = 8)) +
  theme_minimal()

# Arrange plots side by side
#calib_plot <- plot_grid(p36, p44,ncol = 2, align = "h",
   #                     rel_widths = c(1, 1)) +
 # plot_annotation(title = "ROC Curves for Variable Selection Regression Model")

#calib_plot

# Arrange plots side by side
calib_plot <- p36 + p44 +
  plot_layout(guides = "collect") +
  plot_annotation(title = "ROC Curves for Regression in Coefficients Selection")
calib_plot
#####################################################################
#### DEVELOPING A PREDICTIVE MODEL using 36 weeks data ####
#####################################################################
```

```r
trach_df_long <- mice::complete(trach_df_mice_out,action="long")
y <- trach_df_long$Tracheostomy

# partion data into train-test split
set.seed(2550)
train_indices <- createDataPartition(y, p = 0.7, list = FALSE)
train_data <- trach_df_long[train_indices, ]
test_data <- trach_df_long[-train_indices, ]

y_test <- as.factor(test_data$Tracheostomy)

# Fit a logistic model with random intercept and slope
model2 <- glmer(Tracheostomy ~ mat_ethn + birth_hc  + deliv_method + prenat_steroids + comp_prenat_ster

# Predict on test set
preds <- predict(model2, newdata = test_data, type = "response")

#####################################################################
#### DEVELOPING A PREDICTIVE MODEL using 44 weeks data ####
#####################################################################
trach_df_long2 <- mice::complete(trach_df_mice_out2,action="long")
yy <- trach_df_long2$Tracheostomy

# train and test split
set.seed(2550)
train_idx <- createDataPartition(y, p = 0.7, list = FALSE)
train_df <- trach_df_long2[train_idx , ]
test_df <- trach_df_long2[-train_idx , ]

yy_test <- as.factor(test_df$Tracheostomy)

# Fit a mixed-effects logistic regression model
model2.2 <- glmer(Tracheostomy ~ mat_ethn  + deliv_method + prenat_steroids + comp_prenat_steroids + ver

preds_44 <- predict(model2.2, newdata = test_df, type="response")

# model evaluation indicators for 36 weeks
# AUC, sensitivity, specificity, accuracy, precision and Brier score values of the fitted model
evaluation <- function(pred,y_test,threshold=0.5){
  #' get AUC, sensitivity, specificity, accuracy, precision and Brier scorevalues of the fitted model
  #' @param pred, the prediction values
  #' @param y.test, the labels of test dataset
  #' @param threshold, a numeric number of threshold to classify the probability to classes
  #' @return AUC, sensitivity, specificity, accuracy, and precision values
  #'
   pred_numeric <- as.numeric(as.character(pred))
  y_test_numeric <- as.numeric(as.character(y_test))

  # Check if the outcome is binary
  if (length(unique(y_test_numeric)) > 2) {
    stop("Response variable must be binary for ROC analysis.")
  }
```

```
  # ROC curve
  roc_object <- roc(y_test_numeric, pred_numeric, levels = levels(y_test), direction = "<")

  # AUC
  auc <- roc_object$auc

  df <- data.frame(pred = as.numeric(pred_numeric > threshold), label = as.numeric(y_test_numeric))

  TP <- dim(df[(df$pred==1&df$label==1),])[1]
  TN <- dim(df[(df$pred==0&df$label==0),])[1]
  FP <- dim(df[(df$pred==1&df$label==0),])[1]
  FN <- dim(df[(df$pred==0&df$label==1),])[1]

  Recall = TP / (TP + FN)
  Precision = TP / (TP + FP)
  Brier_score = mean((pred_numeric - y_test_numeric)^2)
  F1_score = 2 * (Precision * Recall) / (Precision + Recall)

  return(c(AUC = auc, sensitivity = Recall ,
           specificity = TN / (TN + FP),
            accuracy = (TP + TN) / (TP + TN + FP + FN),
           precision = Precision,
           "F1 score" = F1_score,
            "Brier Score" = Brier_score))

  }
evaluation_metrics <- evaluation(preds, y_test, threshold = 0.5)

# model evaluation indicators for 44weeks
# AUC, sensitivity, specificity, accuracy, precision and Brier score values of the fitted model
evaln <- function(pred,yy_test,threshold=0.5){
  #' get AUC, sensitivity, specificity, accuracy, precision and Brier scorevalues of the fitted model
  #' @param pred, the prediction values
  #' @param y.test, the labels of test dataset
  #' @param threshold, a numeric number of threshold to classify the probability to classes
  #' @return AUC, sensitivity, specificity, accuracy, and precision values
  #'
   pred_numeric_44 <- as.numeric(as.character(pred))
  yy_test_numeric <- as.numeric(as.character(yy_test))

  # Check if response is binary
  if (length(unique(yy_test_numeric)) > 2) {
    stop("Response variable must be binary for ROC analysis.")
  }

  # ROC curve
  roc_obj <- roc(yy_test_numeric, pred_numeric_44, levels = levels(yy_test), direction = "<")

  # AUC
  auc <- roc_obj$auc

  df <- data.frame(pred = as.numeric(pred_numeric_44 > threshold), label = as.numeric(yy_test_numeric))
```

```
  TP <- dim(df[(df$pred==1&df$label==1),])[1]
  TN <- dim(df[(df$pred==0&df$label==0),])[1]
  FP <- dim(df[(df$pred==1&df$label==0),])[1]
  FN <- dim(df[(df$pred==0&df$label==1),])[1]

  recall = TP / (TP + FN)
  precision = TP / (TP + FP)
  brier_score = mean((pred_numeric_44 - yy_test_numeric)^2)
  f1_score = 2 * (precision * recall) / (precision + recall)

  return(c(AUC = auc, sensitivity = recall ,
           specificity = TN / (TN + FP),
            accuracy = (TP + TN) / (TP + TN + FP + FN),
           precision = precision,
           "F1 score" = f1_score,
            "Brier Score" = brier_score))

  }
evaluation_metric <- evaln(preds_44, yy_test, threshold = 0.5)

# Evaluation Metrics at 36 and 44 weeks
evaluation_36 <- evaluation(preds, y_test, threshold = 0.5)
evaluation_44 <- evaln(preds_44, yy_test, threshold = 0.5)

evaln_df <- data.frame(
  Metric = c("AUC", "Sensitivity", "Specificity", "Accuracy", "Precision",
             "F1 Score", "Brier Score"),Value = evaluation_metric)

eval_df <- data.frame(
  Metric = c("AUC", "Sensitivity", "Specificity", "Accuracy", "Precision","F1 Score",
             "Brier Score"),Value = evaluation_metrics)

# Combine the two data frames
merged_df <- merge(evaln_df, eval_df, by = "Metric", suffixes = c("_44wks", "_36wks"))

colnames(merged_df) <- c("Metric", "Model for 44wks", "Model for 36wks")

   merged_df%>%
  kable(caption = "Measures of discrimination and calibration", booktabs = TRUE, align = "c") %>%
  kable_styling(full_width = TRUE, latex_options = c('HOLD_position'))


# calibration plot for logistic regression at 36weeks
num_cuts <- 10
calib_dat <-  data.frame(probs = preds,
                         bin = cut(preds, breaks = num_cuts),
                         class=
                         as.numeric(test_df$Tracheostomy)-1)
calib_dat <- calib_dat %>%
             group_by(bin) %>%
             dplyr::summarize(observed = sum(class)/n(),
                       expected = sum(probs)/n(),
                       se = sqrt(observed * (1- observed)/n()))
```

```r
plot_36 <- ggplot(calib_dat) +
  geom_abline(intercept = 0, slope = 1, color="red") +
  geom_errorbar(aes(x = expected, ymin = observed - 1.96*se,
                    ymax = observed + 1.96*se),
                position = position_identity(),
                colour="black", width = .01)+
  geom_point(aes(x = expected, y = observed)) +
  labs(x="Expected Proportion", y="Observed Proportion",title = 'Observed vs. Expected',
    subtitle = "36 weeks")+
  theme_minimal()


# calibration plot for logistic regression at 44 weeks
num_cuts <- 10
calib_df <-  data.frame(probs = preds_44,
                        bin = cut(preds_44, breaks = num_cuts),
                        class = as.numeric(test_df$Tracheostomy)-1)
calib_df <- calib_df %>%
             group_by(bin) %>%
             dplyr::summarize(observed = sum(class)/n(),
                      expected = sum(probs)/n(),
                      se = sqrt(observed * (1- observed)/n()))
plot_44 <-ggplot(calib_df) +
  geom_abline(intercept = 0, slope = 1, color="red") +
  geom_errorbar(aes(x = expected, ymin = observed - 1.96*se,
                    ymax = observed + 1.96*se),
                position = position_identity(),
                colour="black", width = .01)+
  geom_point(aes(x = expected, y = observed)) +
  labs(x ="Expected Proportion", y="Observed Proportion", title = 'Observed vs. Expected ',
       subtitle = "44 weeks") +
  theme_minimal()

# Arrange plots side by side
calib_plots <- plot_36 + plot_44 +
  plot_layout(guides = "collect") +
  plot_annotation(title = "Calibration Plot -Multilevel Logistic Regression Model")
calib_plots
#########################################################
#### Model evaluation for 36 weeks and 44weeks ####
#########################################################
# roc and auc
preds <- as.numeric(preds)
preds_44 <- as.numeric(preds_44)

logistic_36wks <- roc(test_data$Tracheostomy,preds)
logistic_44wks <- roc(test_df$Tracheostomy, preds_44) #preds_44)

roc_curve_36 <- data.frame(FPR = 1-logistic_36wks$specificities,
                           TPR = logistic_36wks$sensitivities)
roc_curve_44 <- data.frame(FPR = 1-logistic_44wks$specificities,
                           TPR = logistic_44wks$sensitivities)
```

```r
roc_dat <- rbind(roc_curve_36,roc_curve_44)
roc_dat$model <- c(rep("logistic 36",nrow(roc_curve_36)),
                   rep("logistic 44",nrow(roc_curve_44)))

# plot ROC curves for models that were used in variable selection
ggplot(roc_dat, aes(x = FPR, y = TPR, color = model)) +
  geom_line() +
  geom_abline(intercept = 0, slope = 1, color = "grey", linetype = "dashed") +
  labs(x = "False Positive Rate", y = "True Positive Rate",
       title = "ROC Curves for the Predictive models") +
  theme(plot.title = element_text(hjust = 0.5),
        plot.subtitle = element_text(hjust = 0.5,size = 8)) +
  theme_minimal()
```

.