Developing Prediction Models for Tracheostomy in Infants with Severe Bronchopulmonary Dysplasia

Alitzel Serrano Laguna

2023-12-16

Abstract

This paper conducts a thorough examination of five tracheostomy outcome prediction models, employing four different fitting methods to select the most effective model. Models undergo evaluation using positive and negative predictive value comparison, accuracy, AUC measure, and coefficient estimates. The results showed that the lasso model with two-way interactions consistently outperforms its counterparts, demonstrating better predictive accuracy compared to all other fitting models.

Introduction

According to Higgins et al., bronchopulmonary dysplasia (BPD) remains a critical concern in neonatal health. BPD is characterized by abnormal development of lung tissue which primarily affects preterm infants, posing significant challenges in respiratory function. Infants born prematurely frequently require extensive medical intervention, including mechanical ventilation and oxygen therapy. Hence, it is imperative for physicians to monitor the health trajectories of preterm infants to ensure they are receiving adequate care. In cases where an infant develops BPD, it is important know whether their condition evolves to the point of needing a tracheostomy. Receiving a tracheostomy in a timely manner can prevent worsening conditions and potentially avoid infant death. It is essential to understand the gravity of BPD on infants born at or before 32 weeks postmenstrual age (PMA). In this project, our analysis pertains to respiratory support information at critical time points, particularly at 36 weeks PMA, a crucial milestones in the neonatal development.

Data Overview

Pre-processing

Data for this project comes from the (BPD) Collaborative Registry which contains multicenter interdisciplinary BPD programs located in the United States and Sweden. The dataset includes information only from U.S. centers. The data includes baseline and demographic characteristics in addition to respiratory support information at 36 and 44 weeks PMA for infants born at or earlier than 32 weeks postmenstrual age (PMA) who are at higher risk for BPD. Severity of BPD categories (I, II, III) at week 36 PMA were coded using the NHLBI(2018) definition. This is a revised definition of the 2001 NHLBI guidelines. In this definition, cutoff points for invasive intermittent positive pressure (IPPV), noninvasive positive pressure ventilation (NIPPV), and other oxygen levels are established for each category [1]. Additionally, duplicate IDs were removed from the dataset.

Missing Data

Table 1 summarizes missing data for covariates with greater than 10% missingness. From this table, it is observed that respiratory covariate measures at week 44 have more than 40% missingness. This can be due to the infants being discharged early, observing an outcome, or being transferred to another center. Additionally, Table 2 summarizes the mean percentage of missing data for participants with greater than 30% missing data, grouped by center and trach outcome. Only centers 1,2,4,7,and 12 had at least 7 participants with more than 30% missing data. We also observe that one infant is missing data from which center they received care. Infants who received a tracheostomy also tended to have a larger percentage of missing data. In going forward with the analysis, we only consider data for week 36, based on the patterns described previously, this data appears to be missing at random, and thus imputation can be considered.

Population Characteristics

Table 3 summarizes the overall population characteristics and also stratifies by tracheostomy outcome. The dataset includes a total of 996 infants, 58.9% being male, 14.7% receiving a tracheostomy, and 5.4% having died. This data was collected across ten different centers. In stratifying by tracheostomy outcome, we observe n = 146 infants had a tracheostomy. There are also statistically significant differences between infants who had and did not have a tracheostomy. Infants who received a tracheostomy, had a significantly lower mean weight at birth and 36 weeks but not at 44 weeks compared to infants who did not receive a tracheostomy. Infants who received a tracheostomy also had higher values for fraction of inspired oxygen, peak inspiratory pressure(cmH2O), and positive and exploratory pressure (cm H2O) at weeks 36 and 44. Their mean gestational age of hospital discharge is 79.94 compared to 48.92 for infants who did not receive a tracheostomy. The outcome of interest is limited to a tracheostomy according to 2001 NHLBI guidelines which accounts for death as an outcome of interest under certain cisrcumstances. In not having available information for cause of death, this outcome was excluded.

Moreover, in Table 4, we observe differences across centers, with centers 1, 2, and 12 having the highest number of infants who received a tracheostomy. The majority of infants (n = 630) are from center 2. Additionally from Table 3, we observed differences among infants who did and did not receive a tracheostomy for ventilation support and pulmonary hypertension at both weeks 36 and 44. In Table 4, stratified by centers, we observe that center 21 only has data for one infant (removed from the table due to NA values), center 20 only has data for four infants (removed due to small number of observations for center), and center 4 has completely missing data for week 44 measurements.

In addition, correlations between variables of interest are explored Particularly, many baseline covariates such as birth length, birth weight, head circumference, and gestational age tend to be positively and strongly correlated.This is expected as preterm infants do not have the time to fully develop in utero.

Table 1: Summary of Missing Data by Variable

Variable	Count	Percentage
inspired_oxygen.44	448	44.98
p_delta.44	448	44.98
weight_today.44	446	44.78
peep_cm_h2o_modified.44	446	44.78
any_surf	433	43.47
ventilation_support_level_modified.44	424	42.57
med_ph.44	424	42.57
com_prenat_ster	193	19.38
p_delta.36	128	12.85
hosp_dc_ga	124	12.45
peep_cm_h2o_modified.36	117	11.75

Table 2: Infants with greater than 30 percent missing data

Center	Trach	Mean Percentage	Std. Dev.	n
1	0	38.71	4.56	4
1	1	38.71	5.59	3
2	0	34.67	1.61	4
2	1	39.43	5.54	9
3	0	32.26	NA	1
4	0	33.87	1.76	6
4	1	36.02	3.77	6
7	0	33.18	1.57	7
12	0	33.55	2.88	5
12	1	36.56	2.86	12
NA	1	38.71	NA	1

Table 3: Population Characteristics Stratified by Tracheostomy Outcome

	Overall	No Trach	Trach	p	test
n	981	840	141		
center (%)				<0.001	
1	55 (5.6)	32 (3.8)	23 (16.3)		
2	630 (64.2)	566 (67.4)	64 (45.4)		
3	57 (5.8)	56 (6.7)	1 (0.7)		
4	60 (6.1)	49 (5.8)	11 (7.8)		
5	40 (4.1)	35 (4.2)	5 (3.5)		
7	32 (3.3)	31 (3.7)	1 (0.7)		
12	69 (7.0)	34 (4.0)	35 (24.8)		
16	38 (3.9)	37 (4.4)	1 (0.7)		
Maternal Race (%)				0.024	
0	536 (57.6)	472 (59.2)	64 (48.1)		
1	286 (30.8)	240 (30.1)	46 (34.6)		
2	108 (11.6)	85 (10.7)	23 (17.3)		
Maternal Ethnicity = 2 (%)	859 (92.5)	733 (92.1)	126 (94.7)	0.371	
Birth weight (g) (mean (SD))	806.03 (295.74)	812.67 (293.69)	766.50 (305.79)	0.086	
Gestational Age (mean (SD))	25.78 (2.14)	25.76 (2.14)	25.85 (2.16)	0.651	
Birth Length (cm) (mean (SD))	32.51 (3.80)	32.57 (3.78)	32.04 (4.00)	0.165	
birth_hc (mean (SD))	23.19 (2.76)	23.21 (2.71)	23.02 (3.11)	0.487	
del_method = 2 (%)	696 (71.1)	587 (70.0)	109 (77.9)	0.071	
prenat_ster = Yes (%)	821 (86.8)	703 (85.7)	118 (93.7)	0.021	
com_prenat_ster = Yes (%)	604 (76.4)	520 (76.1)	84 (77.8)	0.801	
mat_chorio = Yes (%)	156 (16.9)	135 (17.0)	21 (16.2)	0.910	
Gender = Male (%)	577 (59.0)	493 (58.9)	84 (59.6)	0.954	
sga = SGA (%)	196 (20.3)	156 (18.8)	40 (29.2)	0.007	
any_surf = Yes (%)	454 (81.8)	390 (80.9)	64 (87.7)	0.218	
weight_today.36 (mean (SD))	2120.62 (412.00)	2130.18 (410.94)	2034.13 (413.82)	0.037	
ventilation_support_level.36 (%)				<0.001	
0	115 (12.1)	109 (13.1)	6 (4.9)		
1	583 (61.2)	555 (66.9)	28 (22.8)		
2	254 (26.7)	165 (19.9)	89 (72.4)		
inspired_oxygen.36 (mean (SD))	0.34 (0.15)	0.32 (0.13)	0.49 (0.20)	<0.001	
p_delta.36 (mean (SD))	5.24 (9.76)	4.23 (8.88)	15.14 (12.29)	<0.001	
peep_cm_h2o_modified.36 (mean (SD))	6.33 (2.90)	6.21 (2.89)	7.41 (2.74)	<0.001	
med_ph.36 = 1 (%)	65 (6.8)	41 (4.9)	24 (19.5)	<0.001	
weight_today.44 (mean (SD))	3646.32 (681.50)	3665.75 (666.75)	3554.91 (743.99)	0.152	
ventilation_support_level_modified.44 (%)				<0.001	
0	265 (47.6)	258 (57.2)	7 (6.6)		
1	140 (25.1)	124 (27.5)	16 (15.1)		
2	152 (27.3)	69 (15.3)	83 (78.3)		
inspired_oxygen.44 (mean (SD))	0.34 (0.15)	0.32 (0.13)	0.44 (0.19)	<0.001	
p_delta.44 (mean (SD))	7.64 (14.30)	4.80 (12.10)	21.62 (16.08)	<0.001	
peep_cm_h2o_modified.44 (mean (SD))	4.24 (4.44)	3.35 (4.07)	8.60 (3.56)	<0.001	
med_ph.44 = 1 (%)	95 (17.1)	47 (10.4)	48 (45.3)	<0.001	
hosp_dc_ga (mean (SD))	52.74 (26.58)	48.89 (23.66)	80.07 (30.05)	<0.001	
Trach = 1 (%)	141 (14.4)	0 (0.0)	141 (100.0)	<0.001	
Death = Yes (%)	54 (5.5)	37 (4.4)	17 (12.1)	0.001	
severity.36 (%)				<0.001	
mild	260 (28.9)	253 (31.2)	7 (7.8)		
moderate	232 (25.8)	226 (27.9)	6 (6.7)		
severe	408 (45.3)	331 (40.9)	77 (85.6)		

Table 4: Population Characterstics Stratified by Center

	1	2	3	4	5	7	12	16	p	test
n	55	630	57	60	40	32	69	38		
Maternal Race (%)									<0.001	
0	20 (52.6)	391 (62.1)	31 (59.6)	40 (69.0)	12 (30.0)	1 (20.0)	16 (23.2)	25 (65.8)		
1	18 (47.4)	192 (30.5)	10 (19.2)	16 (27.6)	26 (65.0)	3 (60.0)	16 (23.2)	5 (13.2)		
2	0 (0.0)	47 (7.5)	11 (21.2)	2 (3.4)	2 (5.0)	1 (20.0)	37 (53.6)	8 (21.1)		
Birth weight (g) (mean (SD))	684.62 (196.95)	832.35 (312.62)	764.81 (254.94)	833.25 (258.78)	605.35 (107.03)	724.88 (225.73)	781.00 (253.55)	889.29 (354.17)	<0.001	
Gestational Age (mean (SD))	25.65 (1.78)	25.87 (2.19)	25.70 (2.08)	25.75 (2.03)	24.08 (1.54)	25.09 (1.86)	26.07 (1.91)	26.29 (2.36)	<0.001	
Birth Length (cm) (mean (SD))	30.77 (4.03)	32.75 (3.85)	32.18 (3.65)	33.20 (3.47)	29.46 (2.01)	32.08 (3.33)	32.41 (3.01)	33.71 (3.96)	<0.001	
birth_hc (mean (SD))	22.43 (2.07)	23.31 (2.73)	23.46 (3.41)	23.73 (2.88)	21.05 (1.51)	22.27 (2.22)	23.23 (2.79)	23.76 (2.92)	<0.001	
del_method = 2 (%)	40 (72.7)	453 (71.9)	40 (70.2)	42 (70.0)	26 (65.0)	22 (68.8)	49 (73.1)	24 (63.2)	0.932	
prenat_ster = Yes (%)	46 (90.2)	544 (86.5)	47 (87.0)	47 (79.7)	37 (92.5)	26 (86.7)	41 (89.1)	33 (89.2)	0.693	
com_prenat_ster = Yes (%)	29 (72.5)	415 (79.0)	41 (87.2)	25 (58.1)	27 (73.0)	15 (65.2)	26 (60.5)	26 (78.8)	0.003	
mat_chorio = Yes (%)	14 (48.3)	105 (16.7)	4 (11.8)	8 (13.6)	14 (35.9)	3 (9.7)	6 (8.7)	2 (6.1)	<0.001	
Gender = Male (%)	34 (61.8)	379 (60.4)	34 (60.7)	32 (53.3)	23 (57.5)	16 (50.0)	41 (59.4)	18 (47.4)	0.690	
sga = SGA (%)	21 (38.9)	117 (18.9)	13 (24.1)	5 (8.5)	8 (20.0)	8 (25.0)	17 (24.6)	7 (18.4)	0.008	
any_surf = Yes (%)	31 (91.2)	265 (79.1)	54 (96.4)	11 (68.8)	33 (94.3)	3 (50.0)	48 (84.2)	9 (56.2)	<0.001	
weight_today.36 (mean (SD))	2072.56 (440.04)	2134.88 (408.32)	2105.93 (424.61)	2126.17 (339.43)	1921.83 (401.40)	2169.32 (408.61)	2039.92 (478.50)	2219.50 (408.51)	0.037	
ventilation_support_level.36 (%)									<0.001	
0	7 (12.7)	50 (8.1)	5 (8.9)	8 (13.3)	0 (0.0)	22 (68.8)	1 (2.0)	22 (57.9)		
1	19 (34.5)	425 (68.4)	35 (62.5)	34 (56.7)	31 (77.5)	8 (25.0)	17 (34.0)	14 (36.8)		
2	29 (52.7)	146 (23.5)	16 (28.6)	18 (30.0)	9 (22.5)	2 (6.2)	32 (64.0)	2 (5.3)		
inspired_oxygen.36 (mean (SD))	0.43 (0.21)	0.32 (0.14)	0.32 (0.10)	0.40 (0.12)	0.36 (0.13)	0.36 (0.10)	0.40 (0.19)	0.35 (0.11)	<0.001	
p_delta.36 (mean (SD))	7.46 (8.40)	5.30 (10.77)	6.74 (7.72)	5.21 (5.73)	4.06 (6.74)	0.14 (0.79)	9.11 (7.17)	1.29 (4.67)	0.001	
peep_cm_h2o_modified.36 (mean (SD))	7.41 (4.39)	6.48 (2.37)	7.70 (3.15)	5.61 (2.49)	8.82 (1.57)	1.68 (2.75)	6.51 (2.01)	3.34 (4.12)	<0.001	
med_ph.36 = 1 (%)	13 (23.6)	25 (4.0)	3 (5.4)	11 (18.3)	3 (7.5)	2 (6.2)	4 (8.0)	4 (10.5)	<0.001	
weight_today.44 (mean (SD))	3612.10 (880.43)	3703.56 (625.28)	3643.95 (745.27)	NaN (NA)	3489.00 (622.19)	3805.00 (732.85)	3303.72 (767.89)	3235.60 (986.58)	0.007	
ventilation_support_level_modified.44 (%)									NaN	
0	9 (17.6)	198 (50.6)	12 (60.0)	0 (NaN)	19 (61.3)	10 (83.3)	12 (25.5)	5 (100.0)		
1	14 (27.5)	97 (24.8)	7 (35.0)	0 (NaN)	9 (29.0)	0 (0.0)	13 (27.7)	0 (0.0)		
2	28 (54.9)	96 (24.6)	1 (5.0)	0 (NaN)	3 (9.7)	2 (16.7)	22 (46.8)	0 (0.0)		
inspired_oxygen.44 (mean (SD))	0.39 (0.20)	0.33 (0.13)	0.30 (0.11)	NaN (NA)	0.30 (0.08)	0.37 (0.16)	0.41 (0.20)	0.27 (0.04)	0.002	
p_delta.44 (mean (SD))	10.35 (10.95)	8.36 (15.78)	0.47 (2.06)	NaN (NA)	0.69 (3.53)	0.00 (0.00)	8.31 (10.13)	0.00 (0.00)	0.006	
peep_cm_h2o_modified.44 (mean (SD))	8.48 (5.29)	3.83 (4.06)	2.89 (4.51)	NaN (NA)	3.29 (4.28)	1.58 (3.75)	5.68 (4.17)	0.00 (0.00)	<0.001	
med_ph.44 = 1 (%)	26 (51.0)	41 (10.5)	1 (5.0)	0 (NaN)	5 (16.1)	4 (33.3)	17 (36.2)	1 (20.0)	NaN	
hosp_dc_ga (mean (SD))	59.70 (NA)	52.95 (18.40)	55.35 (72.95)	NaN (NA)	54.18 (17.75)	44.60 (6.75)	57.69 (33.76)	41.41 (3.25)	NA	
Trach = 1 (%)	23 (41.8)	64 (10.2)	1 (1.8)	11 (18.3)	5 (12.5)	1 (3.1)	35 (50.7)	1 (2.6)	<0.001	
Death = Yes (%)	7 (12.7)	29 (4.6)	1 (1.8)	1 (1.7)	2 (5.0)	0 (0.0)	14 (20.3)	0 (0.0)	<0.001	
severity.36 (%)									<0.001	
mild	8 (18.6)	177 (29.7)	10 (18.2)	10 (17.5)	4 (10.0)	23 (74.2)	5 (12.2)	23 (60.5)		
moderate	6 (14.0)	177 (29.7)	15 (27.3)	9 (15.8)	8 (20.0)	1 (3.2)	6 (14.6)	10 (26.3)		
severe	29 (67.4)	241 (40.5)	30 (54.5)	38 (66.7)	28 (70.0)	7 (22.6)	30 (73.2)	5 (13.2)		

Correlations Among Covariates

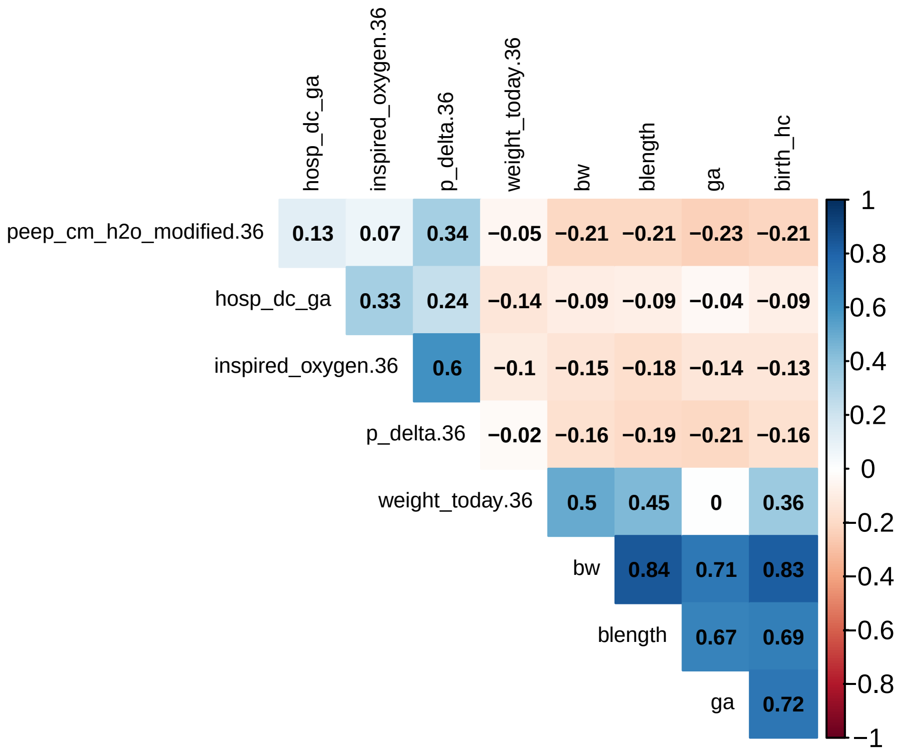

Table 5: Zero Coefficients for Models

Lasso	Lasso2	Backward
7	30	1

Table 6: Close Coefficients

Column Pair	Close Count
lass01_vs_lass02	15
lass01_vs_Backward	6
lass02_vs_Backward	7

Model Validation, Derivation, and Selection

All statistical analyses were performed using R Version 4.3.1. Using the mice package in R, 5 imputed train and test datasets were generated with a 70-30 train-test split. The models were fit on the training data and validated on the test data. A total of 7 models were developed from the 30 available predictor variables to predict whether an infant will undergo a tracheostomy. Model 1 included only main interactions and was derived and cross-validated using a ridge model selection procedure. Model 2 was allowed to include potential two-way interactions and Model 3 included only main interactions. Both Model 2 and 3 were derived and cross-validated through a lasso variable selection procedure. The glmnet package in R was used to fit both ridge and lasso models. Lastly, models 4 and 5, were fit and allowed to include only one-way terms, with model 4 following a forward stepwise and model 5 following a backward stepwise model selection procedure.

Discussion

Table 7 summarizes the positive predictive values, negative predictive values, and accuracy measure for each fitted model. As expected, the ridge model performed the worse out of all models, with the lowest PPV value of 0.159. Despite the addition of two-way interaction terms, the lasso with only one-way terms performed similar to the lasso two-way interaction in terms of negative predictive values. However the lasso with two-way interactions performed better in terms of overall accuracy, having an accuracy of 0.35. The ridge, forward and backward performed similarly in overall accuracy.

Table 8 summarizes the ROC curves plotted above. The lasso two-way had the highest sensitivity and specificity out of all the models. Ridge, forward, and backward performed similarly, with slight tradeoffs between ridge and forward/backward for specificity and sensitivity. The optimal threshold selected for the two-way lasso is 0.115, a lower threshold compared to all other models. The final model selected from examining the overall performance is the two-way lasso model.

In Tables 5, we observe that the lasso two-way model shrunk 17 coefficients towards zero. In contrast, the ridge, forward, and backward only had four zero coefficients. Having more nonzero coefficients within the model most likely affected the accuracy of the ridge, forward, and backwards models when being applied to

Table 7: Metrics by Model

Models	Threshold	Specificity	Sensitivity
Lasso one-way interactions only	0.1376275	0.5472441	0.8285714
Lasso two-way interactions	0.1365600	0.8377953	0.8530612
Backward	0.1573878	0.3448819	0.7551020

Table 8: Coeffient Estimates for Models

	Lasso1	Lasso2	Backward
(Intercept)	-5.560	-4.379	-3.521
x.ord(Intercept)	0.000	0.000	0.000
x.ordcenter2	-0.202	0.000	-1.725
x.ordcenter3	-0.962	0.000	-8.841
x.ordcenter4	-0.008	0.000	-1.476
x.ordcenter5	-0.244	0.000	-2.054
x.ordcenter7	-0.020	0.000	-2.104
x.ordcenter12	1.339	0.000	0.289
x.ordcenter16	0.000	0.000	-1.645
x.ordmat_race1	0.263	0.000	0.434
x.ordmat_race2	0.000	0.000	0.484
x.ordmat_ethn2	0.006	0.000	0.341
x.ordbw	0.000	0.000	0.002
x.ordga	0.026	0.000	0.004
x.ordblength	0.000	0.000	-0.110
x.ordbirth_hc	0.009	0.000	0.052
x.orddel_method2	0.363	0.000	0.616
x.ordprenat_sterYes	0.428	0.000	1.103
x.ordcom_prenat_sterYes	0.021	0.000	0.124
x.ordmat_chorioYes	-0.010	0.000	-0.453
x.ordgenderMale	-0.123	0.000	-0.534
x.ordsgaSGA	0.025	0.000	0.002
x.ordweight_today.36	0.000	0.000	0.000
x.ordventilation_support_level.361	-0.360	-0.050	-9.960
x.ordventilation_support_level.362	0.208	0.000	-9.436
x.ordinspired_oxygen.36	2.963	0.000	2.575
x.ordp_delta.36	0.000	0.000	0.004
x.ordpeep_cm_h2o_modified.36	0.000	0.000	-0.019
x.ordmed_ph.361	0.017	0.000	0.072
x.ordhosp_dc_ga	0.014	0.000	0.029
x.ordseverity.36moderate	0.000	0.000	9.284
x.ordseverity.36severe	0.823	0.000	10.428

Table 9: Coeffient Estimates for Lasso Two-Way

	Coefficients
(Intercept)	-4.3793806
weight_today.36	-0.0000463
ventilation_support_level.361	-0.0501652
center12:mat_race1	0.0594861
center12:mat_race2	0.3564630
center2:mat_ethn2	-0.1620506
center3:mat_ethn2	-0.0966907
center5:mat_ethn2	-0.2356167
center12:bw	0.0006263
center5:ga	-0.0015971
center2:del_method2	-0.4914436
center12:del_method2	0.1444263
center12:com_prenat_sterYes	0.0221428
center2:genderMale	-0.0842595
center4:genderMale	-0.6971119
center3:sgaSGA	-0.0000503
center4:sgaSGA	0.0295775
center5:sgaSGA	-0.2374234
center2:any_surfYes	-0.0237075
center2:ventilation_support_level.361	-0.1695294
center7:ventilation_support_level.362	-0.3259379
center12:ventilation_support_level.362	0.1443081
center12:inspired_oxygen.36	0.0753084
center4:p_delta.36	0.0031917
center7:p_delta.36	-0.0006170
center12:p_delta.36	0.0075498
center2:peep_cm_h2o_modified.36	-0.0366278
center7:peep_cm_h2o_modified.36	-0.0187800
center12:peep_cm_h2o_modified.36	0.0312224
center2:hosp_dc_ga	0.0074630
center3:hosp_dc_ga	-0.0107475
center12:severity.36moderate	0.1553621
center12:severity.36severe	0.1839371
center16:severity.36severe	0.1377586
mat_race1:bw	0.0001255
mat_race1:mat_chorioYes	0.0289028
mat_race2:genderMale	0.0858669
mat_race1:any_surfYes	0.0123003
mat_race1:ventilation_support_level.362	0.0861233
mat_race1:med_ph.361	-0.0344618
mat_race1:hosp_dc_ga	0.0022638
mat_race2:severity.36severe	0.0163931
mat_ethn2:genderMale	-0.0203698
mat_ethn2:hosp_dc_ga	0.0053809
bw:hosp_dc_ga	0.0000007
ga:inspired_oxygen.36	0.0070230
blength:hosp_dc_ga	0.0000539
birth_hc:inspired_oxygen.36	0.0268816
birth_hc:hosp_dc_ga	0.0000752
del_method2:genderMale	-0.0178302
del_method2:any_surfYes	-0.0723423
del_method2:inspired_oxygen.36	0.0031899
del_method2:hosp_dc_ga	0.0142080
prenat_sterYes:genderMale	-0.0562531
prenat_sterYes:any_surfYes	0.0425244

the test data. In Table 6, we compare the closeness of the predicted coefficients for between each model. As expected, forward and backward had the closets number of coefficient estimates due to the zero estimated coefficients which were seen in the two-way models but not in the one-way models. Ridge had 31 similar coefficient estimates to forward and backward. Lasso two-way had the least similar estimated coefficients to all other models. This difference in the lasso two-way's coefficient estimates led to the differences in model performance observed.

Limitations

A major limitation of this study is due to the amount of missing data. Using multiple imputation, with a predictive mean method, a large portion of the data for 44 weeks measures was imputed, thus creating assumptions for what the data looks like at these time points. Additionally, the data provided included indicator for whether an infant died. This was not taken into consideration in this model derivation since the cause of death was not specified. The outcome of death may be due to non-tracheostomy related comorbidity. In having such information, a composite score of death and tracheostomy would be better suited. Lastly, the models were fit to predict tracheostomy outcome, but the time at which it occurs is not specified. These models assume that the infant is alive at 36 and 44 week time points as the models incorporate these data measures.

References

Higgins, R. D., Jobe, A. H., Koso-Thomas, M., Bancalari, E., Viscardi, R. M., Hartert, T. V., Ryan, R. M., Kallapur, S. G., Steinhorn, R. H., Konduri, G. G., Davis, S. D., Thebaud, B., Clyman, R. I., Collaco, J. M., Martin, C. R., Woods, J. C., Finer, N. N., & Raju, T. N. K. (2018). Bronchopulmonary Dysplasia: Executive Summary of a Workshop. The Journal of pediatrics, 197, 300–308. https://doi.org/10.1016/j.jpeds.2018.01.043

Code Appendix:

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

#Load Needed Libraries
library(mice)
library(naniar)
library(tidyverse)
library(gtsummary)
library(gridExtra)
library(tableone)
library(lme4)
library(ggplot2)
library(glmnet)
require(lattice)
require(pan)
library(tableone)
library(leaps)
library(knitr)
library(kableExtra)
library(pROC)

#########################################################
# Data Pre-processing
#########################################################

trach_data <- read.csv("~/Project2/project2.csv")

#delete duplicate ids
#which(trach_data$record_id == 2000824)
trach_data <- trach_data[-c(790,791,792),]

#change numeric to factors
trach_data[,c(2:4,9:15,17,21,23,27, 29:30)] <- lapply(trach_data[,c(2:4,9:15,17,21,23,27,29:30)], facto

#Create severity of BPD variable indicator at week 36
trach_data <- trach_data %>% mutate(severity.36 = case_when(
  (ventilation_support_level.36 == 0 |
     (ventilation_support_level.36 == 1  & inspired_oxygen.36 < 0.22)) ~ 'mild',
  ((ventilation_support_level.36 == 2 & inspired_oxygen.36 <= 0.21) |
     (ventilation_support_level.36 == 1 & inspired_oxygen.36 < 0.30 & inspired_oxygen.36 >= 0.22)) ~ 'mo
   ((ventilation_support_level.36 == 2 & inspired_oxygen.36 > 0.21) |
      (ventilation_support_level.36 == 1 & inspired_oxygen.36 >= 0.30)) ~ 'severe'))



# which(is.na(trach_data$severity))
# sum(is.na(trach_data$severity))
# trach_data %>% group_by(severity, Trach) %>% summarize(n())
#########################################################
# Explore Missingness
```

```
#########################################################

#...by variable
miss_var <- as.data.frame(miss_var_summary(trach_data))
miss_var$pct_miss <- round(miss_var$pct_miss, 2)
Tab1 <- miss_var[miss_var$pct_miss > 10,] %>%
  knitr::kable(caption = "Summary of Missing Data by Variable",
                                                    col.names = c("Variable",  "Count",
                                                                  "Percentage")) %>%
  kable_styling(full_width = F)
Tab1
#summarize missingness by row
pct_na_r <- round(rowSums(is.na(trach_data)) / ncol(trach_data) * 100,2)
row_na <- data.frame(patient_id = trach_data$record_id, pct_na = pct_na_r, Center = trach_data$center, T

row_na <- row_na[row_na$pct_na > 30,]  # participants w/ greater than % missing
df <- row_na %>% group_by(Center, Trach) %>% summarize(Median = round(mean(pct_na),2), SD = round(sd(pct

Tab2<- knitr::kable(df, caption = "Infants with greater than 30 percent missing data",
      col.names = c("Center", "Trach", "Mean Percentage", "Std. Dev.","n")) %>%
  kable_styling(full_width = F)
Tab2
#########################################################
# Summary Statistics
#########################################################

#create summary table overall by trach
subset <- trach_data %>% filter(center != "21" & center != "20") %>% dplyr::select(!c("record_id"))
names(subset) <- c("center", "Maternal Race", "Maternal Ethnicity", "Birth weight (g)","Gestational Age
subset$center <- factor(subset$center )

vars <- c("center", "Maternal Race", "Maternal Ethnicity", "Birth weight (g)","Gestational Age", "Birth

#subset$center <- as.numeric(subset$center)
#subset <- subset %>% filter(center != 10)
tab3 <- CreateTableOne(data = subset, vars = vars, strata = "Trach", addOverall = T)
names(tab3$ContTable) <- c("Overall", "No Trach", "Trach")
names(tab3$CatTable) <- c("Overall", "No Trach", "Trach")
tab3%>% kableone( booktabs=TRUE, caption = "Population Characteristics Stratified by Tracheostomy Outcom

#create summary table stratified by center
subset <- subset %>% filter(center != "21" & center != "20")
subset$center <- factor(subset$center )

vars = c("Maternal Race", "Meternal Ethnicity" , "Birth weight (g)", "Gestational Age", "Birth Length (
, "mat_chorio", "Gender","sga",  "any_surf", "weight_today.36",  "ventilation_support_level.36" ,"inspi
, "inspired_oxygen.44",  "p_delta.44","peep_cm_h2o_modified.44", "med_ph.44","hosp_dc_ga","Trach","Death

Tab4 <- CreateTableOne(data = subset, vars = vars, strata = c("center"))
#kableone(tab3)

outfinish4 <- kableone(Tab4 , align = 'c' , booktabs=TRUE, caption = "Population Characterstics Stratif
```

```r
outfinish4
library(corrplot)
example <- trach_data %>% select_if(is.numeric) %>%
                            dplyr::select(!c(weight_today.44, inspired_oxygen.44, p_delta.44,
                                      peep_cm_h2o_modified.44))

example <- na.omit(example)

correlation_matrix <- cor(example)

par(mar = c(1, 1, 3, 1))
corrplot(
  correlation_matrix,
  method = "color",
  addrect = 2,
  order = "hclust",
  type = "upper",
  number.cex = 0.7,
  tl.cex = 0.7, tl.col = "black",
  addCoef.col = "black",
  diag = FALSE,
  title = "Correlations Among Covariates",  mar=c(1,1,2,1)
)
#################################
# Impute data: train and test
#################################

set.seed(22112) # for reproducibility

#exclude centers with potential influence
trach_data <- trach_data %>% filter(center != "21" & center != "20")
trach_data$center <- factor(trach_data$center )

ignore <- sample(c(TRUE, FALSE), size = nrow(trach_data), replace = TRUE, prob = c(0.3, 0.7))

# Train and test in separate datasets
trach_dat <- trach_data %>%
 dplyr::select(!c(severity.36, weight_today.44, ventilation_support_level_modified.44,
                                     inspired_oxygen.44, p_delta.44,peep_cm_h2o_modified.44,
                                     med_ph.44)) #remove from imputation

traindata <- trach_dat[!ignore, ]
testdata <- trach_dat[ignore, ]
#excluded from imp
traindata_excl <- trach_data[!ignore, ]
testdata_excl <- trach_data[ignore, ]
#conduct imputation
imp.train <- mice(traindata, m = 5, maxit = 5, print = FALSE, seed = 22112)
imp.test2 <- mice.mids(imp.train, newdata = testdata, print = F)

trainingdata <- list()
validationdata <- list()
```

```
#Add excluded variables
for(i in 1:5)
{

   trainingdata[[i]] <- complete(imp.train, i)

  #code severity variable
  trainingdata[[i]] <- trainingdata[[i]] %>% mutate(severity.36 = case_when(
   (ventilation_support_level.36 == 0 |
      (ventilation_support_level.36 == 1  & inspired_oxygen.36 < 0.22)) ~ 'mild',
   ((ventilation_support_level.36 == 2 & inspired_oxygen.36 <= 0.21) |
      (ventilation_support_level.36 == 1 & inspired_oxygen.36 < 0.30 & inspired_oxygen.36 >= 0.22)) ~ 'm
    ((ventilation_support_level.36 == 2 & inspired_oxygen.36 > 0.21) |
       (ventilation_support_level.36 == 1 & inspired_oxygen.36 >= 0.30)) ~ 'severe'))
  trainingdata[[i]]$severity.36 <- as.factor(trainingdata[[i]]$severity.36)

  testdata[[i]] <- complete(imp.test2, i)

  #code severity variable
  testdata[[i]] <- testdata[[i]] %>% mutate(severity.36 = case_when(
   (ventilation_support_level.36 == 0 |
      (ventilation_support_level.36 == 1  & inspired_oxygen.36 < 0.22)) ~ 'mild',
   ((ventilation_support_level.36 == 2 & inspired_oxygen.36 <= 0.21) |
      (ventilation_support_level.36 == 1 & inspired_oxygen.36 < 0.30 & inspired_oxygen.36 >= 0.22)) ~ 'm
    ((ventilation_support_level.36 == 2 & inspired_oxygen.36 > 0.21) |
       (ventilation_support_level.36 == 1 & inspired_oxygen.36 >= 0.30)) ~ 'severe'))
 testdata[[i]]$severity.36 <- as.factor(testdata[[i]]$severity.36)

}

#which(is.na(trainingdata[[1]]$severity.36))
#sum(is.na(testdata[[1]]$severity.36))

###########################
# LASSO
###########################

fit_lasso <- function(train) {
  #' Runs 10-fold CV for one-way and two-way lasso and returns corresponding coefficients
  #' @param train, data set
  #' @return lasso_models, returns a list of models and coefficients for minimum cv error

  #grid range for lambda
  grid <- 10^ seq (10 , -2, length = 100)
  train <- train %>% dplyr::select(!c(Death, record_id)) #remove death

  # Matrix form for ordered variables
  #including two-way interactions
  x.ord2 <- model.matrix(Trach~.^2, data = train)[,-c(22)] #remove trach
  #only single terms
  x.ord <- model.matrix(Trach~., data = train)[,-c(22)]
  y.ord <- as.numeric(train$Trach)
  y.ord <- ifelse(y.ord == 2, 1, 0)
  # Generate folds
  k <- 10
  set.seed(1) # consistent seeds between imputed data sets
  folds <- sample(1:k, nrow(train), replace=TRUE)

  # Lasso model without interactions
  lasso_mod_cv <- cv.glmnet(x.ord, y.ord, nfolds = 10, foldid = folds, alpha = 1, lambda = grid, family
  bestlam <- lasso_mod_cv$lambda.min
  lasso_mod1 <- glmnet(x.ord, y.ord, nfolds = 10, foldid = folds,alpha = 1, lambda = bestlam, family =

  # Lasso model with interactions
  lasso_mod_cv2 <- cv.glmnet(x.ord2, y.ord, nfolds = 10, foldid = folds, alpha = 1, lambda = grid, famil

  bestlam2 <- lasso_mod_cv2$lambda.min

  lasso_mod2 <- glmnet(x.ord2, y.ord, nfolds = 10, foldid = folds,  alpha = 1, lambda = bestlam2, famil

  # Get coefficients
  coef1 <- coef(lasso_mod1)
  coef2 <- coef(lasso_mod2)

  #return best model and its coefficients
  lasso_models <- list()
  lasso_models <- list(coef1, coef2, lasso_mod1, lasso_mod2)

  return(lasso_models)
}

# trainingdata[[1]]%>% dplyr::select(!c(Death, record_id))

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

lasso_coef_all_1 <- cbind(lasso_coef1[[1]], lasso_coef2[[1]], lasso_coef3[[1]],lasso_coef4[[1]], lasso_

lasso_coef_all_2 <- cbind(lasso_coef1[[2]], lasso_coef2[[2]], lasso_coef3[[2]],lasso_coef4[[2]], lasso_

#for the model with  main interactions
avg_coefs_lasso1 <- apply(lasso_coef_all_1, 1, mean)
var_coefs_lasso1 <- apply(lasso_coef_all_1, 1, var)
#for the model with two-way interactions
avg_coefs_lasso2 <- apply(lasso_coef_all_2, 1, mean)
var_coefs_lasso2 <- apply(lasso_coef_all_2, 1, var)

# Find predicted probabilities on long imputed test data (no rounding applied in this case!)
trach_df_long <- complete(imp.test2,action="long")
subset_long <- trach_df_long %>% dplyr::select(!c(.imp, .id,record_id,Death))
subset_long <- subset_long %>% mutate(severity.36 = case_when(
   (ventilation_support_level.36 == 0 |
      (ventilation_support_level.36 == 1  & inspired_oxygen.36 < 0.22)) ~ 'mild',
   ((ventilation_support_level.36 == 2 & inspired_oxygen.36 <= 0.21) |
      (ventilation_support_level.36 == 1 & inspired_oxygen.36 < 0.30 & inspired_oxygen.36 >= 0.22)) ~ '
    ((ventilation_support_level.36 == 2 & inspired_oxygen.36 > 0.21) |
       (ventilation_support_level.36 == 1 & inspired_oxygen.36 >= 0.30)) ~ 'severe'))

subset_long$severity.36 <- as.factor(subset_long$severity)

#for one way interactions
x_vars <- model.matrix(Trach~. , subset_long)
subset_long$score_lasso1 <- x_vars %*% (avg_coefs_lasso1)
mod_lasso1 <- glm(Trach~score_lasso1, data = subset_long, family = "binomial")
predict_probs_lasso1 <- predict(mod_lasso1, type="response")

#for two way interactions
x_vars2 <- model.matrix(Trach~(.)^2 , subset_long %>%dplyr::select(!c(score_lasso1)))
subset_long$score_lasso2 <- x_vars2 %*% (avg_coefs_lasso2)
mod_lasso2 <- glm(Trach~score_lasso2, data = subset_long, family = "binomial")
predict_probs_lasso2 <- predict(mod_lasso2, type="response")


#Discrimination - ROC and AUC
roc_mod_lasso1 <- pROC::roc(predictor=predict_probs_lasso1,
                response=as.factor(mod_lasso1$y), levels = c(0,1), direction = "<")
#plot(main = "Lasso Model One-Way", roc_mod_lasso1, print.auc = T, print.thres = T)
roc_mod_lasso2  <- pROC::roc(predictor=predict_probs_lasso2,
                response=as.factor(mod_lasso2$y))
#plot(main = "Lasso Model Two-Way", roc_mod_lasso2, print.auc = T, print.thres = T)

roc_vals_lasso1 <- pROC::coords(roc=roc_mod_lasso1, x = "best")
roc_vals_lasso2 <- pROC::coords(roc=roc_mod_lasso2, x = "best")


#comparison of values
get_vals <- function(pred_probs, thresh, y)
{
pred_ys <- ifelse(pred_probs > thresh, 1, 0)
pred_ys <- factor(pred_ys, levels = c("0", "1"))
tab_outcome <- table(mod_lasso1$y, pred_ys)
tab_outcome
#sens <- tab_outcome[2,2]/(tab_outcome[2,1]+tab_outcome[2,2])
#spec <- tab_outcome[1,1]/(tab_outcome[1,1]+tab_outcome[1,2])
ppv <- tab_outcome[2,2]/(tab_outcome[1,2]+tab_outcome[2,2])
npv <- tab_outcome[1,1]/(tab_outcome[1,1]+tab_outcome[2,1])
acc <- (tab_outcome[1,1]+tab_outcome[2,2])/sum(tab_outcome)
vals <- data.frame(Measures = c("PPV", "NPV", "Acc"),
           Values = round(c(ppv, npv, acc),3))

return(vals)
}


vals_lasso1 <- get_vals(predict_probs_lasso1,roc_vals_lasso1$threshold,mod_lasso1$y)
```

```
vals_lasso2 <- get_vals(predict_probs_lasso2,roc_vals_lasso2$threshold,mod_lasso2$y)


logistic_backward <- function(train)
 {
   #' Fits logistic model and returns corresponding coefficients
   #' @param train, data set
   #' @return dat, a list containing the fitted model and its coefficients

 train <- train %>% dplyr::select(!c(Death, record_id)) #remove death and id

   # Matrix form for ordered variables
  #including two-way interactions
  x.ord2 <- model.matrix(Trach~.^2, data = train)[,-c(22)] #remove trach
  #only single terms
  x.ord <- model.matrix(Trach~., data = train)[,-c(22)]
  y.ord <- train$Trach

  log_modfull <- glm(y.ord ~ x.ord, family = "binomial")
   log_modnull <- glm(y.ord ~ 1, family = "binomial")

backwardstep <- step(log_modfull, scope = formula(log_modnull), direction='backward', trace = 0)
coef <- coef(backwardstep)

dat <- list(coef, backwardstep)
return(dat)

}

logisticb1 <- logistic_backward(trainingdata[[1]])
logisticb2 <- logistic_backward(trainingdata[[2]])
logisticb3 <- logistic_backward(trainingdata[[3]])
logisticb4 <- logistic_backward(trainingdata[[4]])
logisticb5 <- logistic_backward(trainingdata[[5]])

logistic_b_coef <- cbind(logisticb1[[1]], logisticb2[[1]], logisticb3[[1]],logisticb4[[1]],logisticb5[[
#' logistic_coef[is.na(logistic_coef)] <- 0
#for the model with  main interactions
avg_coefs_backward <- apply(logistic_b_coef, 1, mean)
var_coefs_backward <- apply(logistic_b_coef, 1, var)

#for one way interactions
x_vars <- model.matrix(Trach~. , subset_long %>% dplyr::select(!c(score_lasso1, score_lasso2)))
x_vars[2] <- 0
avg_coefs_backward[2] <- 0 #x.ord intercept
subset_long$score_bwd <- x_vars %*% (avg_coefs_backward)
mod_logistic_bwd <- glm(Trach~score_bwd, data = subset_long, family = "binomial")
predict_probs_bwd <- predict(mod_logistic_bwd, type="response")

#Discrimination - ROC and AUC
roc_mod_backward <- pROC::roc(predictor=predict_probs_bwd,
                response=as.factor(mod_logistic_bwd$y), levels = c(0,1), direction = "<")
#plot(main = "Backward Model",roc_mod_backward, print.auc = T, print.thres = T)
```

```
roc_vals_backward <- pROC::coords(roc=roc_mod_backward, x = "best")

#comparison of values
vals_backward<- get_vals(predict_probs_bwd,roc_vals_backward$threshold, mod_logistic_bwd$y)

#
# df <- data.frame(
#   Metric = c("Sensitivity", "Specificity", "AUC", "Best Threshold"),
#   Lasso1 = c(
#     roc_vals_lasso1$sensitivity,
#     roc_vals_lasso1$specificity,
#     auc(roc_mod_lasso1),
#     roc_vals_lasso1$threshold
#   ),
#    Lasso2 = c(
#     roc_vals_lasso2$sensitivity,
#     roc_vals_lasso2$specificity,
#     auc(roc_mod_lasso2),
#     roc_vals_lasso2$threshold
#   ),
#    Backward = c(
#     roc_vals_backward$sensitivity,
#     roc_vals_backward$specificity,
#     auc(roc_mod_backward),
#     roc_vals_backward$threshold
#   )
# )
#
# df %>% knitr::kable(caption = "Performance Metrics for Models")

#Table of coefficients
lass1 <- avg_coefs_lasso1  #[avg_coefs_lasso1 != 0.00]
lass2 <- avg_coefs_lasso2[1:length(lass1)] #[avg_coefs_lasso2 != 0.00]
bwd <- avg_coefs_backward

coef_df <- cbind(lass01 = round(lass1,4),lass02 = round(lass2,4), Backward = round(bwd,4))

df <- as.data.frame(coef_df)

df2 <- df %>%
  summarise_all(~sum(. == 0))

Tab10 <- df2 %>%
  knitr::kable(col.names = c("Lasso", "Lasso2", "Backward"), booktabs = T, caption = "Zero Coefficients
   kable_styling(latex_options = c("striped"),full_width = F)
Tab10


#closeness of values
threshold = 0.02

close_counts <- combn(names(df), 2, function(cols) {
  sum(abs(df[[cols[1]]] - df[[cols[2]]]) < threshold)
```

```
}, simplify = TRUE)

# Summarize the total count of close values for each pair of columns
summary_result <- data.frame(
  Column_Pair = combn(names(df), 2, paste, collapse = "_vs_"),
  Close_Count = close_counts
)
Tab8 <- summary_result %>% knitr::kable(col.names = c("Column Pair", "Close Count"), booktabs = T, capti
Tab8
#Tables for model comparison
values <- cbind(Measure = c("PPV", "NPV", "Acc"), Lasso = vals_lasso1[,2], Lasso_int= vals_lasso2[,2],
                Backward = vals_backward[,2])

tab4 <- as.data.frame(values)


Tab4 <- knitr::kable(tab4, caption = "Metrics by Model", col.names = c("Measure",  "Lasso one-way intera
  kable_styling(latex_options = c("striped"),full_width = F)
#Tab4


tab <- rbind(roc_vals_lasso1, roc_vals_lasso2, roc_vals_backward)
Models <- c("Lasso one-way interactions only", "Lasso two-way interactions","Backward")



tab5 <- cbind(Models, tab)

Tab5 <- knitr::kable(tab5, caption = "Metrics by Model", col.names = c("Models", "Threshold","Specifici
   kable_styling(latex_options = c("striped"),full_width = F)
 Tab5




#TABLES OF COEFFICIENTS

lass1 <- avg_coefs_lasso1
lass2 <- avg_coefs_lasso2[1:32]
bwd <- avg_coefs_backward

# Assuming lass1, lass2, and bwd are your coefficient vectors
# with potentially different lengths

# Combine coefficients into a data frame
coefficients_df <- data.frame(
  Lasso1 = c(lass1, rep(NA, max(length(lass1), length(lass2), length(bwd)) - length(lass1))),
  Lasso2 = c(lass2, rep(NA, max(length(lass1), length(lass2), length(bwd)) - length(lass2))),
  Backward = c(bwd, rep(NA, max(length(lass1), length(lass2), length(bwd)) - length(bwd)))
)

coefficients_df <- round(coefficients_df[1:length(bwd),],3)

coefficients_df %>% knitr::kable(caption = "Coeffient Estimates for Models")
```

```r
lasso_all <- avg_coefs_lasso2[avg_coefs_lasso2 != 0.00]
new <- data.frame(Coefficients = c(lasso_all))
new %>% knitr::kable(caption = "Coeffient Estimates for Lasso Two-Way")
```