Exercise & Assignment: Dealing with High Dimensional Data

Megan Marziali

Mar 14, 2021

## Exercise 1: Feature selection using regularization methods

This exercise is *loosely* based on the following paper: Integration of an interpretable machine learning algorithm to identify early life risk factors of childhood obesity among preterm infants: a prospective birth cohort <https://doi.org/10.1186/s12916-020-01642-6>. The data used in this exercise are an altered version of data available from the HHEAR Data Center, with dois <https://doi.org/10.36043/2017-1740_EPI_58> and <https://doi.org/10.36043/2017-1740_MAIN_84>

In this exercise, you will utilize the caret package to optimize a regularization algorithm for feature selection. You will compare results when you include variables that could induce confounding as features entered into the algorithm. You will also consider how study design and source of data can impact the conclusions drawn by a machine learning analysis.

### Description of the Theoretical Study and Data

The goal of this study is to identify prenatal social and environmental risk factors for childhood overweight/obesity among preterm infants.This study is a prospective birth cohort involving mother-child pairs. Women were enrolled during the first or second trimester of pregnancy and were followed up via visiting clinics until the birth of their children. Women and children were then followed up periodically during infancy and childhood. A total of 1447 singleton children were born preterm, prior to 37 weeks gestation and had complete data on maternal demographics and pregnancy, birth characteristics, lifestyle factors, biospecimen analyzed for exposure to metals and to define childhood obesity at age 5. You have recently been hired as a research data analyst, and tasked with performing the analysis for this study.You are provided with a dataset containing a number of features, in addition to a binary outcome indicating childhood overweight or obesity vs normal weight.

Features in the dataset have informative names. The following categorical features use codes to indicate the different labels:

Child.Human.Biological.Sex: sex assigned at birth of child; 110:Female 111:Male

HHIncome: Household income during pregnancy; 159: <$5,000 204:$5,000-$10,000, 205: $10,000-$20,000, 206: $20,000-$40,000, 207: $40,000-$70,000, 208: >$70,000

Race\_ethnicity: Race or ethnicity of child, as reported by parent; 47:“Hispanic or Latino Ethnicity” 54:“Multiracial” 210:“Black Non-Hispanic” 212:“American Indian Non-Hispanic” 214:“White Non-Hispanic” 217:“Asian Non-Hispanic” 855824:“Other race/ethnicity than white/black/hispanic/asian/american indian/multiracial”

Mother\_Education: Highest Educational Attainment at time of Pregnancy 4:“Advanced Graduate Degree” 12:“College Graduate” 32:“Graduated From High School” 203:“Some College or Technical School” 215:“Less than High School”

Smoking\_Preg: maternal smoking during pregnancy 1: No smoking during pregnancy 2: Active smoker during pregnancy 81: Quit smoking before pregnancy

ow\_obesity: overweight or obesity during childhood 1: Overweight or Obese >= 85th percentile 0: Typical weight <85th percentile (no underweight children in sample)

###Before Data Analysis Question 1: What additional information, if any, would you want from the principal study investigators in regards to the above features?

We would want to obtain domain knowledge from the principal study investigators. This includes when and where the data was collected, to get a better understanding of the study context. They may have come across abnormalities within the data, or ways in which the study was carried out, that they would flag to other investigators. For example, they may be aware of ways in which the questions were posed which could bias the way the respondents answered and would want to flag those questions to respondents. As smoking during pregnancy is highly stigmatized, we would want to obtain information regarding how the question was asked in order to be able to gauge whether misclassification is of concern. They additionally may be able to advise regarding appropriate categorization or recoding of variables, as they have done in prior studies or have seen done in the literature. They may have also considered possible confounders and modifiers that they would want to flag for other investigators.

In relation to other features in the dataset, we would need to know clinically relevant values, which the investigators could provide.

Question 2: Look at the features in the dataset before you start your analysis. Are there any you want to exclude from your analysis completely? Why or why not? Are there any you want to recode or transform? Why or why not?

To be able to answer whether we would want to recode some variables, we might want to begin data exploration and determine whether small cell sizes are of concern. Once we have a better understanding of this, we could then recode variables appropriately and potentially collapse categories. We would also want to exclude variables that have a lot of missing data, which would need to be assessed. We could also consider removing variables that we hypothesize to be highly correlated. We would likely want to recode the chemical concentration variables as greater than or equal to a concentration of clinical significance; if I had access to appropriate resources, I would want to recode all the concentration variables to be greater than or below clinically significant levels, as this would be more interpretable.

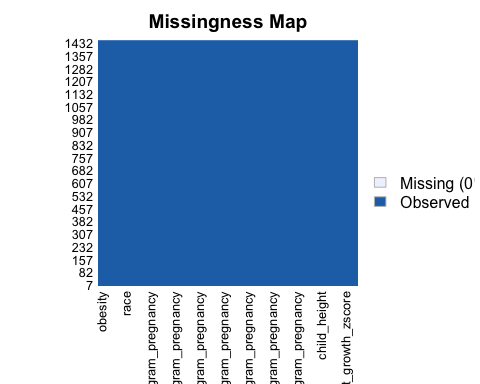
Question 3: Are any of the features not of interest as modifiable contributors to childhood overweight/obesity themselves, but in an explanatory model, you would typically include them? Will you include them in your analysis?

Of the features included, there are non-modifiable contributors listed. **Gestational length** could be non-modifiable, if we assume that the mother had given birth preterm resultant of a medical condition. **Child height** could be considered non-modifiable, as well as **mothers age at birth** and **sex** of the child. I would also argue that **race/ethnicity** of the child is non-modifiable.

### Step 1: Load Packages and Prepare Data

You will start by loading the needed packages. Some are already listed, but you can choose to use different ones. You will need to clean the data, check that values are plausible, ensure that all variables are the correct type for the algorithm and packages you want to use, etc.

library(tidyverse)  
library(caret)  
library(glmnet)  
library(Amelia)  
library(arsenal)  
  
#Importing and recoding the data to be appropriate for the variable type  
bw.data = read\_csv("./data/birthcohort\_data.csv") %>%   
 janitor::clean\_names() %>%   
 mutate(  
 sex = as.factor(child\_human\_biological\_sex),  
 income = as.factor(hh\_income),  
 race = as.factor(race\_ethnicity),  
 education = as.factor(mother\_education),  
 smoking = as.factor(smoking\_preg),  
 obesity = as.factor(ow\_obesity)  
 ) %>%   
 select(  
 -child\_human\_biological\_sex,  
 -hh\_income,  
 -race\_ethnicity,  
 -mother\_education,  
 -smoking\_preg,  
 -ow\_obesity  
 )  
  
#Checking for missing values  
missmap(bw.data)



#Checking reasonable values  
table.1 = tableby(obesity ~ ., data = bw.data,  
 numeric.stats = c("mean","median", "range"))  
summary(table.1, text = TRUE)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 0 (N=1105) | 1 (N=342) | Total (N=1447) | p value |
| birth\_weight |  |  |  | < 0.001 |
| - Mean | 3453.409 | 3589.670 | 3485.614 |  |
| - Median | 3487.000 | 3595.000 | 3515.000 |  |
| - Range | 481.000 - 5273.000 | 1389.000 - 5528.000 | 481.000 - 5528.000 |  |
| infant\_growth\_zscore |  |  |  | < 0.001 |
| - Mean | 0.069 | 1.206 | 0.337 |  |
| - Median | 0.070 | 1.160 | 0.340 |  |
| - Range | -3.430 - 4.170 | -3.260 - 3.820 | -3.430 - 4.170 |  |
| gestational\_length |  |  |  | 0.215 |
| - Mean | 36.469 | 36.610 | 36.502 |  |
| - Median | 36.710 | 36.710 | 36.710 |  |
| - Range | 23.290 - 39.710 | 26.290 - 39.860 | 23.290 - 39.860 |  |
| birth\_length |  |  |  | 0.113 |
| - Mean | 49.922 | 49.717 | 49.873 |  |
| - Median | 50.100 | 50.000 | 50.100 |  |
| - Range | 43.100 - 56.100 | 43.200 - 54.900 | 43.100 - 56.100 |  |
| child\_height |  |  |  | 0.061 |
| - Mean | 66.856 | 66.528 | 66.778 |  |
| - Median | 67.000 | 66.700 | 66.900 |  |
| - Range | 55.600 - 76.000 | 55.600 - 73.600 | 55.600 - 76.000 |  |
| mother\_age |  |  |  | 0.475 |
| - Mean | 32.252 | 32.457 | 32.300 |  |
| - Median | 32.360 | 32.700 | 32.430 |  |
| - Range | 16.280 - 44.900 | 16.870 - 44.490 | 16.280 - 44.900 |  |
| mother\_bmi\_pre\_preg |  |  |  | 0.051 |
| - Mean | 24.805 | 25.471 | 24.962 |  |
| - Median | 23.480 | 23.940 | 23.540 |  |
| - Range | 16.190 - 48.510 | 16.960 - 50.140 | 16.190 - 50.140 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_aluminum\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.087 |
| - Mean | 20.927 | 17.627 | 20.147 |  |
| - Median | 17.500 | 17.850 | 17.700 |  |
| - Range | -51.100 - 648.000 | -34.900 - 164.000 | -51.100 - 648.000 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_antimony\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.470 |
| - Mean | 0.162 | 0.096 | 0.147 |  |
| - Median | 0.075 | 0.073 | 0.075 |  |
| - Range | -1.140 - 56.500 | -0.025 - 1.010 | -1.140 - 56.500 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_arsenic\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.999 |
| - Mean | 1.296 | 1.296 | 1.296 |  |
| - Median | 0.855 | 0.828 | 0.849 |  |
| - Range | -0.250 - 34.100 | -0.144 - 12.400 | -0.250 - 34.100 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_barium\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.366 |
| - Mean | 5.308 | 5.652 | 5.389 |  |
| - Median | 3.160 | 3.370 | 3.190 |  |
| - Range | -0.140 - 59.200 | -0.249 - 39.700 | -0.249 - 59.200 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_cadmium\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.014 |
| - Mean | 0.483 | 0.549 | 0.499 |  |
| - Median | 0.400 | 0.389 | 0.395 |  |
| - Range | 0.083 - 4.380 | 0.087 - 6.940 | 0.083 - 6.940 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_cesium\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.598 |
| - Mean | 2.679 | 2.647 | 2.672 |  |
| - Median | 2.530 | 2.485 | 2.530 |  |
| - Range | 0.839 - 10.000 | 0.455 - 7.450 | 0.455 - 10.000 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_chromium\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.016 |
| - Mean | 1.920 | 1.632 | 1.852 |  |
| - Median | 1.400 | 1.370 | 1.390 |  |
| - Range | 0.081 - 25.100 | 0.053 - 11.500 | 0.053 - 25.100 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_cobalt\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.060 |
| - Mean | 0.039 | 0.031 | 0.037 |  |
| - Median | 0.032 | 0.027 | 0.031 |  |
| - Range | -0.115 - 1.130 | -0.069 - 0.374 | -0.115 - 1.130 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_copper\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.568 |
| - Mean | 572.414 | 577.112 | 573.524 |  |
| - Median | 566.000 | 559.000 | 564.000 |  |
| - Range | 256.000 - 1050.000 | 86.300 - 3970.000 | 86.300 - 3970.000 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_lead\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.078 |
| - Mean | 19.501 | 20.518 | 19.741 |  |
| - Median | 17.700 | 17.600 | 17.700 |  |
| - Range | 4.110 - 78.400 | 4.850 - 90.800 | 4.110 - 90.800 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_magnesium\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.832 |
| - Mean | 41883.891 | 41782.368 | 41859.896 |  |
| - Median | 41100.000 | 41100.000 | 41100.000 |  |
| - Range | 16800.000 - 77500.000 | 7070.000 - 68800.000 | 7070.000 - 77500.000 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_manganese\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.407 |
| - Mean | 17.369 | 17.058 | 17.295 |  |
| - Median | 16.300 | 15.650 | 16.200 |  |
| - Range | 5.360 - 45.300 | 6.040 - 44.300 | 5.360 - 45.300 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_molybdenum\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.969 |
| - Mean | 0.284 | 0.283 | 0.284 |  |
| - Median | 0.260 | 0.262 | 0.260 |  |
| - Range | -0.444 - 4.740 | -0.412 - 2.510 | -0.444 - 4.740 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_nickel\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.358 |
| - Mean | 0.852 | 0.669 | 0.808 |  |
| - Median | 0.673 | 0.633 | 0.663 |  |
| - Range | -0.318 - 121.000 | -0.044 - 5.120 | -0.318 - 121.000 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_selenium\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.218 |
| - Mean | 256.599 | 252.073 | 255.529 |  |
| - Median | 251.000 | 241.500 | 248.000 |  |
| - Range | 98.400 - 1010.000 | 39.000 - 1090.000 | 39.000 - 1090.000 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_thallium\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.419 |
| - Mean | -0.027 | -0.030 | -0.028 |  |
| - Median | -0.014 | -0.016 | -0.014 |  |
| - Range | -0.213 - 0.128 | -0.206 - 0.095 | -0.213 - 0.128 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_tin\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.071 |
| - Mean | 1.549 | 1.468 | 1.529 |  |
| - Median | 1.500 | 1.390 | 1.470 |  |
| - Range | -0.176 - 7.490 | -0.152 - 5.850 | -0.176 - 7.490 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_vanadium\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.370 |
| - Mean | 0.105 | 0.095 | 0.103 |  |
| - Median | 0.086 | 0.083 | 0.085 |  |
| - Range | -0.008 - 6.270 | 0.004 - 1.470 | -0.008 - 6.270 |  |
| pregnancy\_sample\_red\_blood\_cell\_concentration\_zinc\_nanogram\_per\_gram\_pregnancy |  |  |  | 0.002 |
| - Mean | 10556.724 | 10202.544 | 10473.013 |  |
| - Median | 10500.000 | 10150.000 | 10400.000 |  |
| - Range | 4630.000 - 26000.000 | 2310.000 - 14900.000 | 2310.000 - 26000.000 |  |
| sex |  |  |  | 0.021 |
| - 110 | 554 (50.1%) | 147 (43.0%) | 701 (48.4%) |  |
| - 111 | 551 (49.9%) | 195 (57.0%) | 746 (51.6%) |  |
| income |  |  |  | 0.841 |
| - 159 | 4 (0.4%) | 3 (0.9%) | 7 (0.5%) |  |
| - 204 | 17 (1.5%) | 4 (1.2%) | 21 (1.5%) |  |
| - 205 | 27 (2.4%) | 10 (2.9%) | 37 (2.6%) |  |
| - 206 | 120 (10.9%) | 39 (11.4%) | 159 (11.0%) |  |
| - 207 | 267 (24.2%) | 80 (23.4%) | 347 (24.0%) |  |
| - 208 | 670 (60.6%) | 206 (60.2%) | 876 (60.5%) |  |
| race |  |  |  | 0.162 |
| - 47 | 63 (5.7%) | 13 (3.8%) | 76 (5.3%) |  |
| - 54 | 98 (8.9%) | 28 (8.2%) | 126 (8.7%) |  |
| - 210 | 141 (12.8%) | 49 (14.3%) | 190 (13.1%) |  |
| - 212 | 1 (0.1%) | 0 (0.0%) | 1 (0.1%) |  |
| - 214 | 763 (69.0%) | 238 (69.6%) | 1001 (69.2%) |  |
| - 217 | 39 (3.5%) | 12 (3.5%) | 51 (3.5%) |  |
| - 855824 | 0 (0.0%) | 2 (0.6%) | 2 (0.1%) |  |
| education |  |  |  | 0.926 |
| - 4 | 328 (29.7%) | 106 (31.0%) | 434 (30.0%) |  |
| - 12 | 436 (39.5%) | 126 (36.8%) | 562 (38.8%) |  |
| - 32 | 82 (7.4%) | 27 (7.9%) | 109 (7.5%) |  |
| - 203 | 246 (22.3%) | 78 (22.8%) | 324 (22.4%) |  |
| - 215 | 13 (1.2%) | 5 (1.5%) | 18 (1.2%) |  |
| smoking |  |  |  | 0.754 |
| - 1 | 753 (68.1%) | 235 (68.7%) | 988 (68.3%) |  |
| - 2 | 131 (11.9%) | 44 (12.9%) | 175 (12.1%) |  |
| - 81 | 221 (20.0%) | 63 (18.4%) | 284 (19.6%) |  |

**NOTE**: the table is readable in R, but it won’t convert properly to word.

From table 1, some of the values that the cellular concentrations take on are not plausible. It is not reasonable to have a negative concentration of a metal in the blood, the plausible values range from 0 upwards. We would want to consult with the study PIs to understand how these values were collected, and whether there was a measurement error or data input error. As preterm births are generally <37 weeks, it is not plausible that the gestational length exceeds 37 weeks; however, the maximum values observed from Table 1 exceed 37 weeks. To run this analysis properly, we should likely exclude births that are not preterm.

This table also allows us to understand that there are small cell sizes. Income levels 159, 204 and 205 should be collapsed and will be recoded as such. Race level 212 and 855824 should be combined or excluded from analysis. Education level 215 should be collapsed with 203.

bw.data = bw.data %>%   
 mutate(  
 income =   
 recode(income,  
 "159" = "<20000",  
 "204" = "<20000",  
 "205" = "<20000",  
 "206" = "20000-40000",  
 "207" = "40000-70000",  
 "208" = ">70000"),  
 race =   
 recode(race,  
 "47" = "Hispanic",  
 "54" = "Other",  
 "855824" = "Other",  
 "210" = "Black",  
 "212" = "Other",  
 "214" = "White",  
 "217" = "Asian"),  
 education =  
 recode(education,  
 "4" = "Advanced grad",  
 "12" = "College",  
 "32" = "Some high school or less",  
 "203" = "Some college",  
 "215" = "Some high school or less")  
 )  
  
table.1 = tableby(obesity ~ ., data = bw.data,  
 numeric.stats = c("mean","median", "range"))  
summary(table.1, test = TRUE)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 0 (N=1105) | 1 (N=342) | Total (N=1447) | p value |
| **birth\_weight** |  |  |  | < 0.001 |
| Mean | 3453.409 | 3589.670 | 3485.614 |  |
| Median | 3487.000 | 3595.000 | 3515.000 |  |
| Range | 481.000 - 5273.000 | 1389.000 - 5528.000 | 481.000 - 5528.000 |  |
| **infant\_growth\_zscore** |  |  |  | < 0.001 |
| Mean | 0.069 | 1.206 | 0.337 |  |
| Median | 0.070 | 1.160 | 0.340 |  |
| Range | -3.430 - 4.170 | -3.260 - 3.820 | -3.430 - 4.170 |  |
| **gestational\_length** |  |  |  | 0.215 |
| Mean | 36.469 | 36.610 | 36.502 |  |
| Median | 36.710 | 36.710 | 36.710 |  |
| Range | 23.290 - 39.710 | 26.290 - 39.860 | 23.290 - 39.860 |  |
| **birth\_length** |  |  |  | 0.113 |
| Mean | 49.922 | 49.717 | 49.873 |  |
| Median | 50.100 | 50.000 | 50.100 |  |
| Range | 43.100 - 56.100 | 43.200 - 54.900 | 43.100 - 56.100 |  |
| **child\_height** |  |  |  | 0.061 |
| Mean | 66.856 | 66.528 | 66.778 |  |
| Median | 67.000 | 66.700 | 66.900 |  |
| Range | 55.600 - 76.000 | 55.600 - 73.600 | 55.600 - 76.000 |  |
| **mother\_age** |  |  |  | 0.475 |
| Mean | 32.252 | 32.457 | 32.300 |  |
| Median | 32.360 | 32.700 | 32.430 |  |
| Range | 16.280 - 44.900 | 16.870 - 44.490 | 16.280 - 44.900 |  |
| **mother\_bmi\_pre\_preg** |  |  |  | 0.051 |
| Mean | 24.805 | 25.471 | 24.962 |  |
| Median | 23.480 | 23.940 | 23.540 |  |
| Range | 16.190 - 48.510 | 16.960 - 50.140 | 16.190 - 50.140 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_aluminum\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.087 |
| Mean | 20.927 | 17.627 | 20.147 |  |
| Median | 17.500 | 17.850 | 17.700 |  |
| Range | -51.100 - 648.000 | -34.900 - 164.000 | -51.100 - 648.000 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_antimony\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.470 |
| Mean | 0.162 | 0.096 | 0.147 |  |
| Median | 0.075 | 0.073 | 0.075 |  |
| Range | -1.140 - 56.500 | -0.025 - 1.010 | -1.140 - 56.500 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_arsenic\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.999 |
| Mean | 1.296 | 1.296 | 1.296 |  |
| Median | 0.855 | 0.828 | 0.849 |  |
| Range | -0.250 - 34.100 | -0.144 - 12.400 | -0.250 - 34.100 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_barium\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.366 |
| Mean | 5.308 | 5.652 | 5.389 |  |
| Median | 3.160 | 3.370 | 3.190 |  |
| Range | -0.140 - 59.200 | -0.249 - 39.700 | -0.249 - 59.200 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_cadmium\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.014 |
| Mean | 0.483 | 0.549 | 0.499 |  |
| Median | 0.400 | 0.389 | 0.395 |  |
| Range | 0.083 - 4.380 | 0.087 - 6.940 | 0.083 - 6.940 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_cesium\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.598 |
| Mean | 2.679 | 2.647 | 2.672 |  |
| Median | 2.530 | 2.485 | 2.530 |  |
| Range | 0.839 - 10.000 | 0.455 - 7.450 | 0.455 - 10.000 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_chromium\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.016 |
| Mean | 1.920 | 1.632 | 1.852 |  |
| Median | 1.400 | 1.370 | 1.390 |  |
| Range | 0.081 - 25.100 | 0.053 - 11.500 | 0.053 - 25.100 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_cobalt\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.060 |
| Mean | 0.039 | 0.031 | 0.037 |  |
| Median | 0.032 | 0.027 | 0.031 |  |
| Range | -0.115 - 1.130 | -0.069 - 0.374 | -0.115 - 1.130 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_copper\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.568 |
| Mean | 572.414 | 577.112 | 573.524 |  |
| Median | 566.000 | 559.000 | 564.000 |  |
| Range | 256.000 - 1050.000 | 86.300 - 3970.000 | 86.300 - 3970.000 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_lead\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.078 |
| Mean | 19.501 | 20.518 | 19.741 |  |
| Median | 17.700 | 17.600 | 17.700 |  |
| Range | 4.110 - 78.400 | 4.850 - 90.800 | 4.110 - 90.800 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_magnesium\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.832 |
| Mean | 41883.891 | 41782.368 | 41859.896 |  |
| Median | 41100.000 | 41100.000 | 41100.000 |  |
| Range | 16800.000 - 77500.000 | 7070.000 - 68800.000 | 7070.000 - 77500.000 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_manganese\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.407 |
| Mean | 17.369 | 17.058 | 17.295 |  |
| Median | 16.300 | 15.650 | 16.200 |  |
| Range | 5.360 - 45.300 | 6.040 - 44.300 | 5.360 - 45.300 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_molybdenum\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.969 |
| Mean | 0.284 | 0.283 | 0.284 |  |
| Median | 0.260 | 0.262 | 0.260 |  |
| Range | -0.444 - 4.740 | -0.412 - 2.510 | -0.444 - 4.740 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_nickel\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.358 |
| Mean | 0.852 | 0.669 | 0.808 |  |
| Median | 0.673 | 0.633 | 0.663 |  |
| Range | -0.318 - 121.000 | -0.044 - 5.120 | -0.318 - 121.000 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_selenium\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.218 |
| Mean | 256.599 | 252.073 | 255.529 |  |
| Median | 251.000 | 241.500 | 248.000 |  |
| Range | 98.400 - 1010.000 | 39.000 - 1090.000 | 39.000 - 1090.000 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_thallium\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.419 |
| Mean | -0.027 | -0.030 | -0.028 |  |
| Median | -0.014 | -0.016 | -0.014 |  |
| Range | -0.213 - 0.128 | -0.206 - 0.095 | -0.213 - 0.128 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_tin\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.071 |
| Mean | 1.549 | 1.468 | 1.529 |  |
| Median | 1.500 | 1.390 | 1.470 |  |
| Range | -0.176 - 7.490 | -0.152 - 5.850 | -0.176 - 7.490 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_vanadium\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.370 |
| Mean | 0.105 | 0.095 | 0.103 |  |
| Median | 0.086 | 0.083 | 0.085 |  |
| Range | -0.008 - 6.270 | 0.004 - 1.470 | -0.008 - 6.270 |  |
| **pregnancy\_sample\_red\_blood\_cell\_concentration\_zinc\_nanogram\_per\_gram\_pregnancy** |  |  |  | 0.002 |
| Mean | 10556.724 | 10202.544 | 10473.013 |  |
| Median | 10500.000 | 10150.000 | 10400.000 |  |
| Range | 4630.000 - 26000.000 | 2310.000 - 14900.000 | 2310.000 - 26000.000 |  |
| **sex** |  |  |  | 0.021 |
| 110 | 554 (50.1%) | 147 (43.0%) | 701 (48.4%) |  |
| 111 | 551 (49.9%) | 195 (57.0%) | 746 (51.6%) |  |
| **income** |  |  |  | 0.946 |
| <20000 | 48 (4.3%) | 17 (5.0%) | 65 (4.5%) |  |
| 20000-40000 | 120 (10.9%) | 39 (11.4%) | 159 (11.0%) |  |
| 40000-70000 | 267 (24.2%) | 80 (23.4%) | 347 (24.0%) |  |
| >70000 | 670 (60.6%) | 206 (60.2%) | 876 (60.5%) |  |
| **race** |  |  |  | 0.680 |
| Hispanic | 63 (5.7%) | 13 (3.8%) | 76 (5.3%) |  |
| Other | 99 (9.0%) | 30 (8.8%) | 129 (8.9%) |  |
| Black | 141 (12.8%) | 49 (14.3%) | 190 (13.1%) |  |
| White | 763 (69.0%) | 238 (69.6%) | 1001 (69.2%) |  |
| Asian | 39 (3.5%) | 12 (3.5%) | 51 (3.5%) |  |
| **education** |  |  |  | 0.846 |
| Advanced grad | 328 (29.7%) | 106 (31.0%) | 434 (30.0%) |  |
| College | 436 (39.5%) | 126 (36.8%) | 562 (38.8%) |  |
| Some high school or less | 95 (8.6%) | 32 (9.4%) | 127 (8.8%) |  |
| Some college | 246 (22.3%) | 78 (22.8%) | 324 (22.4%) |  |
| **smoking** |  |  |  | 0.754 |
| 1 | 753 (68.1%) | 235 (68.7%) | 988 (68.3%) |  |
| 2 | 131 (11.9%) | 44 (12.9%) | 175 (12.1%) |  |
| 81 | 221 (20.0%) | 63 (18.4%) | 284 (19.6%) |  |

The recoded data looks better in terms of small cell sizes.

#Excluding variables previously hypothesized to not contribute to obesity  
bw.new = bw.data %>%   
 select(  
 -child\_height,  
 -gestational\_length,  
 -sex,  
 -race,  
 -mother\_age  
 )  
  
#Checking whether data is unbalanced  
summary(bw.new$obesity) %>%   
 knitr::kable()

|  |  |
| --- | --- |
|  | x |
| 0 | 1105 |
| 1 | 342 |

### Step 2: Decide on a pipeline

Question 4: In previous exercises, we often partition our sample into training and testing. We optimize hyperparameters using cross-validation. Is this pipeline still necessary if our goal is feature selection and not building a prediction model to apply to new data? What do you think?

Yes, we want to partition the data. We are using a data-driven pipeline, and by doing so, we need to be taking the right approaches to check how our algorithms are functioning. Optimizing hyperparameters for feature selection, like varying the number of trees for example, can create notable changes in terms of variable importance. Partitioning the data in order to optimize these hyperparameters is needed, even if the research question is not related to prediction.

Regardless of your answer above, partition the data into a 70/30 split just to get the practice with the programming code to partition.

set.seed(100)  
  
train.indices = createDataPartition(y = bw.new$obesity,p = 0.7,list = FALSE)  
  
training = bw.new[train.indices,]  
testing = bw.new[-train.indices,]

### Step 3: Construct a model using a regularization algorithm and the features of interest in the training data

Question 5: Which regularization algorithm seems most appropriate for this research question? Justify your choice.

I would use a LASSO algorithm. LASSO is highly interpretable, which is necessary when conducting feature selection, as we want to be able to distinguish and understand the features that have been selected; we want to be able to communicate that the selected features are probable risk factors. LASSO is also the algorithm most suitable for feature selection, as the impact of the penalty is to shrink to zero.

Question 6: Which metrics will you use to evaluate your model? Consider your research question and the outcome of interest.

I will use accuracy as an evaluation metric to evaluate the model, using a confusion matrix as the outcome of interest is binary.

Assess how the metric(s) change(s) based on values of the hyperparameters. Construct a grid to explore various values (do not just use the default parameters). Once you have a final model, determine the features that are considered “important” based on the model output.

set.seed(100)  
  
#Create grid to search lambda  
lambda = 10^seq(-3,3, length = 100)  
  
lasso.bw = train(  
 obesity ~.,   
 data = training,   
 method = "glmnet",   
 trControl = trainControl("cv", number = 10, sampling = "down"),   
 tuneGrid = expand.grid(alpha = 0, lambda = lambda)  
)  
  
#Print the values of alpha and lambda that gave best prediction  
lasso.bw$bestTune

## alpha lambda  
## 21 0 0.01629751

#Print all of the options examined  
lasso.bw$results

## alpha lambda Accuracy Kappa AccuracySD KappaSD  
## 1 0 1.000000e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 2 0 1.149757e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 3 0 1.321941e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 4 0 1.519911e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 5 0 1.747528e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 6 0 2.009233e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 7 0 2.310130e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 8 0 2.656088e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 9 0 3.053856e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 10 0 3.511192e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 11 0 4.037017e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 12 0 4.641589e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 13 0 5.336699e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 14 0 6.135907e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 15 0 7.054802e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 16 0 8.111308e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 17 0 9.326033e-03 0.8845855 0.72163034 0.036434406 0.07695893  
## 18 0 1.072267e-02 0.8845855 0.72163034 0.036434406 0.07695893  
## 19 0 1.232847e-02 0.8845855 0.72163034 0.036434406 0.07695893  
## 20 0 1.417474e-02 0.8845855 0.72163034 0.036434406 0.07695893  
## 21 0 1.629751e-02 0.8845855 0.72163034 0.036434406 0.07695893  
## 22 0 1.873817e-02 0.8826247 0.71780521 0.038224049 0.08084678  
## 23 0 2.154435e-02 0.8796738 0.71071533 0.037994479 0.08141461  
## 24 0 2.477076e-02 0.8727723 0.69664441 0.039244082 0.08467211  
## 25 0 2.848036e-02 0.8658901 0.68241769 0.040978870 0.08582527  
## 26 0 3.274549e-02 0.8589788 0.66698391 0.045208279 0.09344668  
## 27 0 3.764936e-02 0.8589885 0.66494392 0.042448666 0.08746952  
## 28 0 4.328761e-02 0.8570375 0.65882701 0.042656924 0.09004021  
## 29 0 4.977024e-02 0.8481654 0.63651074 0.040821566 0.08711871  
## 30 0 5.722368e-02 0.8392836 0.61969915 0.042337759 0.08909088  
## 31 0 6.579332e-02 0.8303922 0.59870285 0.044051768 0.09544078  
## 32 0 7.564633e-02 0.8175888 0.57033186 0.046263784 0.10362419  
## 33 0 8.697490e-02 0.8096874 0.55361498 0.047378928 0.10678940  
## 34 0 1.000000e-01 0.8027567 0.53693118 0.045382534 0.10097168  
## 35 0 1.149757e-01 0.7998156 0.53179718 0.046473925 0.10214834  
## 36 0 1.321941e-01 0.7850417 0.49867936 0.045635339 0.10008462  
## 37 0 1.519911e-01 0.7712483 0.47109855 0.044424906 0.09774200  
## 38 0 1.747528e-01 0.7544652 0.43984116 0.045152491 0.09443467  
## 39 0 2.009233e-01 0.7446321 0.42169526 0.045661428 0.09247436  
## 40 0 2.310130e-01 0.7416715 0.41386586 0.043318917 0.08792430  
## 41 0 2.656088e-01 0.7357698 0.40072944 0.044912832 0.08692060  
## 42 0 3.053856e-01 0.7347699 0.39647746 0.045196476 0.08826927  
## 43 0 3.511192e-01 0.7318288 0.38987424 0.043217923 0.08665406  
## 44 0 4.037017e-01 0.7278781 0.38070217 0.045538585 0.09228515  
## 45 0 4.641589e-01 0.7278781 0.38070217 0.045538585 0.09228515  
## 46 0 5.336699e-01 0.7229373 0.37046788 0.048640451 0.09909555  
## 47 0 6.135907e-01 0.7199670 0.36525356 0.044513808 0.09514767  
## 48 0 7.054802e-01 0.7150359 0.35760154 0.043038557 0.09192905  
## 49 0 8.111308e-01 0.7081149 0.34437844 0.040806976 0.08732212  
## 50 0 9.326033e-01 0.7002233 0.33113744 0.037819763 0.08041762  
## 51 0 1.072267e+00 0.6972627 0.32692412 0.040328669 0.08567682  
## 52 0 1.232847e+00 0.6943021 0.32025632 0.044167925 0.08942102  
## 53 0 1.417474e+00 0.6923219 0.31757820 0.045155408 0.08964478  
## 54 0 1.629751e+00 0.6903417 0.31488126 0.046027192 0.09048406  
## 55 0 1.873817e+00 0.6893613 0.31352252 0.045763046 0.08927701  
## 56 0 2.154435e+00 0.6883809 0.31219228 0.045708137 0.08828208  
## 57 0 2.477076e+00 0.6854203 0.30467681 0.046516881 0.09206277  
## 58 0 2.848036e+00 0.6834595 0.30185084 0.045501529 0.08987211  
## 59 0 3.274549e+00 0.6824694 0.30064713 0.047027002 0.09129918  
## 60 0 3.764936e+00 0.6824694 0.30064713 0.047027002 0.09129918  
## 61 0 4.328761e+00 0.6824694 0.30064713 0.047027002 0.09129918  
## 62 0 4.977024e+00 0.6824597 0.30077172 0.048692854 0.09374635  
## 63 0 5.722368e+00 0.6824597 0.30077172 0.048692854 0.09374635  
## 64 0 6.579332e+00 0.6814793 0.29928450 0.047821098 0.09272068  
## 65 0 7.564633e+00 0.6824694 0.30061029 0.047276035 0.09172557  
## 66 0 8.697490e+00 0.6824694 0.30061029 0.047276035 0.09172557  
## 67 0 1.000000e+01 0.6824694 0.30061029 0.047276035 0.09172557  
## 68 0 1.149757e+01 0.6804989 0.29623011 0.048697632 0.09506010  
## 69 0 1.321941e+01 0.6804989 0.29623011 0.048697632 0.09506010  
## 70 0 1.519911e+01 0.6804989 0.29623011 0.048697632 0.09506010  
## 71 0 1.747528e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 72 0 2.009233e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 73 0 2.310130e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 74 0 2.656088e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 75 0 3.053856e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 76 0 3.511192e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 77 0 4.037017e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 78 0 4.641589e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 79 0 5.336699e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 80 0 6.135907e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 81 0 7.054802e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 82 0 8.111308e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 83 0 9.326033e+01 0.6795088 0.29476900 0.048513964 0.09629301  
## 84 0 1.072267e+02 0.6785187 0.29344718 0.048755926 0.09619829  
## 85 0 1.232847e+02 0.6785187 0.29344718 0.048755926 0.09619829  
## 86 0 1.417474e+02 0.6785187 0.29344718 0.048755926 0.09619829  
## 87 0 1.629751e+02 0.6785187 0.29344718 0.048755926 0.09619829  
## 88 0 1.873817e+02 0.5997282 0.25316025 0.195540079 0.15702721  
## 89 0 2.154435e+02 0.3313434 0.07301536 0.202327327 0.16432212  
## 90 0 2.477076e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 91 0 2.848036e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 92 0 3.274549e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 93 0 3.764936e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 94 0 4.328761e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 95 0 4.977024e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 96 0 5.722368e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 97 0 6.579332e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 98 0 7.564633e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 99 0 8.697490e+02 0.2366919 0.00000000 0.001203023 0.00000000  
## 100 0 1.000000e+03 0.2366919 0.00000000 0.001203023 0.00000000

# Model coefficients  
coef(lasso.bw$finalModel, lasso.bw$bestTune$lambda)

## 32 x 1 sparse Matrix of class "dgCMatrix"  
## 1  
## (Intercept) 2.188266e+01  
## birth\_weight 3.175379e-03  
## infant\_growth\_zscore 1.351351e+00  
## birth\_length -6.673418e-01  
## mother\_bmi\_pre\_preg 1.146378e-02  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_aluminum\_nanogram\_per\_gram\_pregnancy -2.112611e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_antimony\_nanogram\_per\_gram\_pregnancy -4.370786e-02  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_arsenic\_nanogram\_per\_gram\_pregnancy 5.026497e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_barium\_nanogram\_per\_gram\_pregnancy -5.900888e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cadmium\_nanogram\_per\_gram\_pregnancy 1.622783e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cesium\_nanogram\_per\_gram\_pregnancy -1.380336e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_chromium\_nanogram\_per\_gram\_pregnancy -1.665999e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cobalt\_nanogram\_per\_gram\_pregnancy -1.468437e+00  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_copper\_nanogram\_per\_gram\_pregnancy 1.267281e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_lead\_nanogram\_per\_gram\_pregnancy 3.407725e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_magnesium\_nanogram\_per\_gram\_pregnancy 5.235732e-06  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_manganese\_nanogram\_per\_gram\_pregnancy 8.423119e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_molybdenum\_nanogram\_per\_gram\_pregnancy 4.153103e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_nickel\_nanogram\_per\_gram\_pregnancy -1.080762e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_selenium\_nanogram\_per\_gram\_pregnancy -1.122372e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_thallium\_nanogram\_per\_gram\_pregnancy -6.352244e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_tin\_nanogram\_per\_gram\_pregnancy -1.309551e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_vanadium\_nanogram\_per\_gram\_pregnancy 1.148802e+00  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_zinc\_nanogram\_per\_gram\_pregnancy -1.569881e-04  
## income20000-40000 4.601553e-02  
## income40000-70000 2.808543e-01  
## income>70000 1.253970e-01  
## educationCollege -1.905419e-01  
## educationSome high school or less 2.240316e-01  
## educationSome college -9.722870e-02  
## smoking2 -9.133932e-02  
## smoking81 9.980464e-02

varImp(lasso.bw)

## glmnet variable importance  
##   
## only 20 most important variables shown (out of 31)  
##   
## Overall  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cobalt\_nanogram\_per\_gram\_pregnancy 100.0000  
## infant\_growth\_zscore 92.0265  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_vanadium\_nanogram\_per\_gram\_pregnancy 78.2329  
## birth\_length 45.4455  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_thallium\_nanogram\_per\_gram\_pregnancy 43.2583  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_molybdenum\_nanogram\_per\_gram\_pregnancy 28.2822  
## income40000-70000 19.1258  
## educationSome high school or less 15.2562  
## educationCollege 12.9755  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_chromium\_nanogram\_per\_gram\_pregnancy 11.3451  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cadmium\_nanogram\_per\_gram\_pregnancy 11.0508  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_tin\_nanogram\_per\_gram\_pregnancy 8.9177  
## income>70000 8.5392  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_nickel\_nanogram\_per\_gram\_pregnancy 7.3596  
## smoking81 6.7963  
## educationSome college 6.6209  
## smoking2 6.2198  
## income20000-40000 3.1333  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_antimony\_nanogram\_per\_gram\_pregnancy 2.9761  
## mother\_bmi\_pre\_preg 0.7803

# Make predictions  
pred.bw = predict(lasso.bw, training)  
pred.bw.prob = predict(lasso.bw, training, type = "prob")  
  
# Model prediction performance  
eval.results = confusionMatrix(pred.bw, training$obesity, positive = "1")  
print(eval.results)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 673 8  
## 1 101 232  
##   
## Accuracy : 0.8925   
## 95% CI : (0.8718, 0.9109)  
## No Information Rate : 0.7633   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.7376   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.9667   
## Specificity : 0.8695   
## Pos Pred Value : 0.6967   
## Neg Pred Value : 0.9883   
## Prevalence : 0.2367   
## Detection Rate : 0.2288   
## Detection Prevalence : 0.3284   
## Balanced Accuracy : 0.9181   
##   
## 'Positive' Class : 1   
##

#Accuracy of this model is 0.89

The variable that is of the most important is the concentration of cobalt, followed by infant growth score and concentration of vanadium. Levels of income and education are both among the ten most important variables.

### Step 4: Test your final model in the testing dataset

Use the implementation of your model in the testing set to obtain final performance metrics and perform the inference needed to address your research question.

Question 7: Summarize your final conclusion in 2-3 sentences

set.seed(100)  
  
# Using best fit model from above with testing data  
pred.lasso.f = predict(lasso.bw, testing)  
pred.lasso.f.prob = predict(lasso.bw, testing, type = "prob")  
  
# Evaluating in testing data with confusion matrix  
eval.results = confusionMatrix(pred.lasso.f, testing$obesity, positive = "1")  
print(eval.results)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 293 7  
## 1 38 95  
##   
## Accuracy : 0.8961   
## 95% CI : (0.8634, 0.9232)  
## No Information Rate : 0.7644   
## P-Value [Acc > NIR] : 1.648e-12   
##   
## Kappa : 0.7389   
##   
## Mcnemar's Test P-Value : 7.744e-06   
##   
## Sensitivity : 0.9314   
## Specificity : 0.8852   
## Pos Pred Value : 0.7143   
## Neg Pred Value : 0.9767   
## Prevalence : 0.2356   
## Detection Rate : 0.2194   
## Detection Prevalence : 0.3072   
## Balanced Accuracy : 0.9083   
##   
## 'Positive' Class : 1   
##

The final accuracy of this model is **89.6**. The variables of importance suggest that concentrations of some metals and sociodemographic factors are both important when considering risk factors for childhood obesity. Modifiable variables, including infant growth and birth length, suggest that intervening on maternal and child nutrition are both important when ultimately considering childhood obesity.

### Step 5: Construct another model, making a different choice of variable inclusion.

Redo the above, but now make the opposite choice about variable inclusion. That is, if you did not include the features that themselves might not be modifiable contributors to childhod overweight/obesity, but you would typically include in an explanatory model, include them now. Conversely, if you include those variables previously, exclude them now

Question 7: Do the “important” features change when you make a different choice about the other features? Do the hyperparameters that optimize the model change when the additional variables are included? What about model performance?

set.seed(100)  
  
train.indices = createDataPartition(y = bw.data$obesity,p = 0.7,list = FALSE)  
  
training.2 = bw.data[train.indices,]  
testing.2 = bw.data[-train.indices,]  
  
#Create grid to search lambda  
lambda = 10^seq(-3,3, length = 100)  
  
lasso.2 = train(  
 obesity ~.,   
 data = training.2,   
 method = "glmnet",   
 trControl = trainControl("cv", number = 10, sampling = "down"),   
 tuneGrid = expand.grid(alpha = 0, lambda = lambda)  
)  
  
#Print the values of alpha and lambda that gave best prediction  
lasso.2$bestTune

## alpha lambda  
## 22 0 0.01873817

#Print all of the options examined  
lasso.2$results

## alpha lambda Accuracy Kappa AccuracySD KappaSD  
## 1 0 1.000000e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 2 0 1.149757e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 3 0 1.321941e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 4 0 1.519911e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 5 0 1.747528e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 6 0 2.009233e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 7 0 2.310130e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 8 0 2.656088e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 9 0 3.053856e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 10 0 3.511192e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 11 0 4.037017e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 12 0 4.641589e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 13 0 5.336699e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 14 0 6.135907e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 15 0 7.054802e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 16 0 8.111308e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 17 0 9.326033e-03 0.8541837 0.6534224 0.039314201 0.08344424  
## 18 0 1.072267e-02 0.8541837 0.6534224 0.039314201 0.08344424  
## 19 0 1.232847e-02 0.8541837 0.6534224 0.039314201 0.08344424  
## 20 0 1.417474e-02 0.8541837 0.6534224 0.039314201 0.08344424  
## 21 0 1.629751e-02 0.8541837 0.6534224 0.039314201 0.08344424  
## 22 0 1.873817e-02 0.8541837 0.6534224 0.039314201 0.08344424  
## 23 0 2.154435e-02 0.8512328 0.6468157 0.041049423 0.08769747  
## 24 0 2.477076e-02 0.8472821 0.6384820 0.041279306 0.08720364  
## 25 0 2.848036e-02 0.8384100 0.6202824 0.044453950 0.08940323  
## 26 0 3.274549e-02 0.8354494 0.6128001 0.043924289 0.08753434  
## 27 0 3.764936e-02 0.8265774 0.5943288 0.045350185 0.08924513  
## 28 0 4.328761e-02 0.8147447 0.5709473 0.050818201 0.10111282  
## 29 0 4.977024e-02 0.8078431 0.5579215 0.052751548 0.10310979  
## 30 0 5.722368e-02 0.7999418 0.5404699 0.052738821 0.10406890  
## 31 0 6.579332e-02 0.7911085 0.5249667 0.062539292 0.11562792  
## 32 0 7.564633e-02 0.7812464 0.5058709 0.064723389 0.11967933  
## 33 0 8.697490e-02 0.7674335 0.4771475 0.060683118 0.11292468  
## 34 0 1.000000e-01 0.7615026 0.4671865 0.059417571 0.11135987  
## 35 0 1.149757e-01 0.7565424 0.4564932 0.058110495 0.10962205  
## 36 0 1.321941e-01 0.7486605 0.4380118 0.056001991 0.10784459  
## 37 0 1.519911e-01 0.7417783 0.4222330 0.059354631 0.11344569  
## 38 0 1.747528e-01 0.7299456 0.3989934 0.058430211 0.11476916  
## 39 0 2.009233e-01 0.7279752 0.3959750 0.058251044 0.11438931  
## 40 0 2.310130e-01 0.7220637 0.3859745 0.061225822 0.11916463  
## 41 0 2.656088e-01 0.7210542 0.3826872 0.058696246 0.11555956  
## 42 0 3.053856e-01 0.7190837 0.3785902 0.060332584 0.11671439  
## 43 0 3.511192e-01 0.7171326 0.3766321 0.066253797 0.12185899  
## 44 0 4.037017e-01 0.7112211 0.3629944 0.063728642 0.11777297  
## 45 0 4.641589e-01 0.7033197 0.3499965 0.061747865 0.11035055  
## 46 0 5.336699e-01 0.6983984 0.3377959 0.060934618 0.11183008  
## 47 0 6.135907e-01 0.6905164 0.3252889 0.061775510 0.11295422  
## 48 0 7.054802e-01 0.6875461 0.3226031 0.061647806 0.11329300  
## 49 0 8.111308e-01 0.6845758 0.3188232 0.063276080 0.11525138  
## 50 0 9.326033e-01 0.6835857 0.3172774 0.064161965 0.11680980  
## 51 0 1.072267e+00 0.6806348 0.3100547 0.067366080 0.12553466  
## 52 0 1.232847e+00 0.6796641 0.3091618 0.070108503 0.12968638  
## 53 0 1.417474e+00 0.6786838 0.3064042 0.071456605 0.13221588  
## 54 0 1.629751e+00 0.6796738 0.3091938 0.071456681 0.13335855  
## 55 0 1.873817e+00 0.6786935 0.3047089 0.071888223 0.13417793  
## 56 0 2.154435e+00 0.6806639 0.3078730 0.074120731 0.13783220  
## 57 0 2.477076e+00 0.6786935 0.3048752 0.072774119 0.13545366  
## 58 0 2.848036e+00 0.6786935 0.3047703 0.072172952 0.13411285  
## 59 0 3.274549e+00 0.6777034 0.3031370 0.070647355 0.13139405  
## 60 0 3.764936e+00 0.6757426 0.3008013 0.072740489 0.13427782  
## 61 0 4.328761e+00 0.6737721 0.2981760 0.073160540 0.13456023  
## 62 0 4.977024e+00 0.6737624 0.2962924 0.070520025 0.13079995  
## 63 0 5.722368e+00 0.6747428 0.2975784 0.070174838 0.13004254  
## 64 0 6.579332e+00 0.6727723 0.2949106 0.070287960 0.13042681  
## 65 0 7.564633e+00 0.6727723 0.2949106 0.070287960 0.13042681  
## 66 0 8.697490e+00 0.6727723 0.2949106 0.070287960 0.13042681  
## 67 0 1.000000e+01 0.6737527 0.2960401 0.069035010 0.12889329  
## 68 0 1.149757e+01 0.6727626 0.2944732 0.067841201 0.12706116  
## 69 0 1.321941e+01 0.6717822 0.2913991 0.069100190 0.13143814  
## 70 0 1.519911e+01 0.6717822 0.2913991 0.069100190 0.13143814  
## 71 0 1.747528e+01 0.6707921 0.2901609 0.069774702 0.13239310  
## 72 0 2.009233e+01 0.6698020 0.2887151 0.069178960 0.13158764  
## 73 0 2.310130e+01 0.6688216 0.2877004 0.071123010 0.13332995  
## 74 0 2.656088e+01 0.6688216 0.2877004 0.071123010 0.13332995  
## 75 0 3.053856e+01 0.6688216 0.2877004 0.071123010 0.13332995  
## 76 0 3.511192e+01 0.6678315 0.2861696 0.070042940 0.13169331  
## 77 0 4.037017e+01 0.6678315 0.2861696 0.070042940 0.13169331  
## 78 0 4.641589e+01 0.6678315 0.2861696 0.070042940 0.13169331  
## 79 0 5.336699e+01 0.6678315 0.2861696 0.070042940 0.13169331  
## 80 0 6.135907e+01 0.6678315 0.2861696 0.070042940 0.13169331  
## 81 0 7.054802e+01 0.6678315 0.2861696 0.070042940 0.13169331  
## 82 0 8.111308e+01 0.6678315 0.2861696 0.070042940 0.13169331  
## 83 0 9.326033e+01 0.6678315 0.2861696 0.070042940 0.13169331  
## 84 0 1.072267e+02 0.6678315 0.2861696 0.070042940 0.13169331  
## 85 0 1.232847e+02 0.6678315 0.2861696 0.070042940 0.13169331  
## 86 0 1.417474e+02 0.6678315 0.2861696 0.070042940 0.13169331  
## 87 0 1.629751e+02 0.6678315 0.2861696 0.070042940 0.13169331  
## 88 0 1.873817e+02 0.6678315 0.2861696 0.070042940 0.13169331  
## 89 0 2.154435e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 90 0 2.477076e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 91 0 2.848036e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 92 0 3.274549e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 93 0 3.764936e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 94 0 4.328761e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 95 0 4.977024e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 96 0 5.722368e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 97 0 6.579332e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 98 0 7.564633e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 99 0 8.697490e+02 0.2366919 0.0000000 0.001203023 0.00000000  
## 100 0 1.000000e+03 0.2366919 0.0000000 0.001203023 0.00000000

# Model coefficients  
coef(lasso.2$finalModel, lasso.2$bestTune$lambda)

## 40 x 1 sparse Matrix of class "dgCMatrix"  
## 1  
## (Intercept) 2.044697e+01  
## birth\_weight 2.829627e-03  
## infant\_growth\_zscore 1.420458e+00  
## gestational\_length 1.175121e-01  
## birth\_length -6.678169e-01  
## child\_height -3.345940e-02  
## mother\_age 8.138836e-03  
## mother\_bmi\_pre\_preg 3.202887e-02  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_aluminum\_nanogram\_per\_gram\_pregnancy -6.390628e-04  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_antimony\_nanogram\_per\_gram\_pregnancy -1.089766e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_arsenic\_nanogram\_per\_gram\_pregnancy 8.217284e-02  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_barium\_nanogram\_per\_gram\_pregnancy -3.200546e-04  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cadmium\_nanogram\_per\_gram\_pregnancy 2.609329e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cesium\_nanogram\_per\_gram\_pregnancy -1.003616e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_chromium\_nanogram\_per\_gram\_pregnancy -2.459067e-02  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cobalt\_nanogram\_per\_gram\_pregnancy -5.830896e-02  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_copper\_nanogram\_per\_gram\_pregnancy 6.876937e-04  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_lead\_nanogram\_per\_gram\_pregnancy 3.490462e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_magnesium\_nanogram\_per\_gram\_pregnancy 6.521873e-06  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_manganese\_nanogram\_per\_gram\_pregnancy -1.640328e-02  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_molybdenum\_nanogram\_per\_gram\_pregnancy -4.941954e-02  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_nickel\_nanogram\_per\_gram\_pregnancy -2.648440e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_selenium\_nanogram\_per\_gram\_pregnancy -3.715960e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_thallium\_nanogram\_per\_gram\_pregnancy -5.835143e-01  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_tin\_nanogram\_per\_gram\_pregnancy -3.893722e-03  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_vanadium\_nanogram\_per\_gram\_pregnancy 1.542673e+00  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_zinc\_nanogram\_per\_gram\_pregnancy -8.148127e-05  
## sex111 1.427965e-01  
## income20000-40000 1.672576e-01  
## income40000-70000 -2.919680e-01  
## income>70000 -1.904294e-01  
## raceOther 4.873621e-01  
## raceBlack 4.916352e-01  
## raceWhite 1.066963e-01  
## raceAsian 5.363181e-01  
## educationCollege -1.648613e-01  
## educationSome high school or less -5.362663e-02  
## educationSome college -1.283377e-01  
## smoking2 -2.192863e-01  
## smoking81 -3.272285e-02

varImp(lasso.2)

## glmnet variable importance  
##   
## only 20 most important variables shown (out of 39)  
##   
## Overall  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_vanadium\_nanogram\_per\_gram\_pregnancy 100.000  
## infant\_growth\_zscore 92.078  
## birth\_length 43.289  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_thallium\_nanogram\_per\_gram\_pregnancy 37.825  
## raceAsian 34.765  
## raceBlack 31.869  
## raceOther 31.592  
## income40000-70000 18.926  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_nickel\_nanogram\_per\_gram\_pregnancy 17.168  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_cadmium\_nanogram\_per\_gram\_pregnancy 16.914  
## smoking2 14.214  
## income>70000 12.344  
## income20000-40000 10.842  
## educationCollege 10.686  
## sex111 9.256  
## educationSome college 8.319  
## gestational\_length 7.617  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_antimony\_nanogram\_per\_gram\_pregnancy 7.064  
## raceWhite 6.916  
## pregnancy\_sample\_red\_blood\_cell\_concentration\_arsenic\_nanogram\_per\_gram\_pregnancy 5.326

# Make predictions  
pred.bw.2 = predict(lasso.2, training.2)  
pred.bw.prob.2 = predict(lasso.2, training.2, type = "prob")  
  
# Model prediction performance  
eval.results = confusionMatrix(pred.bw.2, training.2$obesity, positive = "1")  
print(eval.results)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 662 16  
## 1 112 224  
##   
## Accuracy : 0.8738   
## 95% CI : (0.8517, 0.8936)  
## No Information Rate : 0.7633   
## P-Value [Acc > NIR] : < 2.2e-16   
##   
## Kappa : 0.693   
##   
## Mcnemar's Test P-Value : < 2.2e-16   
##   
## Sensitivity : 0.9333   
## Specificity : 0.8553   
## Pos Pred Value : 0.6667   
## Neg Pred Value : 0.9764   
## Prevalence : 0.2367   
## Detection Rate : 0.2209   
## Detection Prevalence : 0.3314   
## Balanced Accuracy : 0.8943   
##   
## 'Positive' Class : 1   
##

#Accuracy within training dataset => 0.87  
  
# Using best fit model from above with testing data  
pred.lasso.f.2 = predict(lasso.2, testing.2)  
pred.lasso.f.prob.2 = predict(lasso.2, testing.2, type = "prob")  
  
# Evaluating in testing data with confusion matrix  
eval.results = confusionMatrix(pred.lasso.f.2, testing.2$obesity, positive = "1")  
print(eval.results)

## Confusion Matrix and Statistics  
##   
## Reference  
## Prediction 0 1  
## 0 294 11  
## 1 37 91  
##   
## Accuracy : 0.8891   
## 95% CI : (0.8557, 0.9171)  
## No Information Rate : 0.7644   
## P-Value [Acc > NIR] : 2.816e-11   
##   
## Kappa : 0.7171   
##   
## Mcnemar's Test P-Value : 0.000308   
##   
## Sensitivity : 0.8922   
## Specificity : 0.8882   
## Pos Pred Value : 0.7109   
## Neg Pred Value : 0.9639   
## Prevalence : 0.2356   
## Detection Rate : 0.2102   
## Detection Prevalence : 0.2956   
## Balanced Accuracy : 0.8902   
##   
## 'Positive' Class : 1   
##

#Accuracy within testing dataset => 0.89

The important features change with inclusion of the other variables; the concentration of cobalt becomes the most important variable. When the additional variables are included, lambda does change from **0.016 to 0.019**. The model performance, as evaluated through assessing accuracy, is marginally better in the model with non-modifiable features excluded.

## Exercise 2: Creating more refined phenotypes for an explanatory analysis.

This exercise is *loosely* based on the following paper: Deploying unsupervised clustering analysis to derive clinical phenotypes and risk factors associated with mortality risk in 2022 critically ill patients with COVID-19 in Spain <doi:10.1186/s13054-021-03487-8>. Data were simulated and are not true COVID data.

Researchers are interested in understanding the factors associated with ICU mortality among COVID-19 patients. They hypothesize there are different clinical phenotypes that could be at different risks for mortality and require different medical interventions. The goal of this research is to determine if patient features including demographics and clinical data at ICU admission could be used to separate COVID-19 patients into distinct phenotypic clusters. The secondary aim was to determine if identified phenotypic clusters had different risk of mortality.

You are provided with the following dataset for a subset of 178 COVID-19 patients, with the instructions to conduct an unsupervised analysis to identify phenotypic clusters within the patient population, describe the clusters in terms of the input features and then determine if there are differences in mortality rate across the clusters. All feature data in the dataset have been centered and scaled. The outcome, mortality, is a binary indicator.

Variables in the dataset (covid.csv) are:

Ageyr: the age of the patient at admission to the ICU APACHE:APACHE II Score, Acute Physiology and Chronic Health Evaluation II Score SOFA: SOFA Score Sequential Organ Failure Assessment DDimer: D dimer, a fibrin degradation product, can indicate blood clot formation and breakdown SerumLactate: measures level of lactic acid in the blood, can be indicator of hypoxia Ferritin: measure of iron sufficiency in blood CRP: C-reactive protein, measure of inflammation Creatinine: product of protein metabolism, high levels can indicate impaired kidney function WBC: concentration of white blood cells, marker of infection DBP: diastolic blood pressure Procalcitonin: marker of bacterial infection IGA: immunoglobulin A, measure of antibodies found in mucous membranes Oxmetric.1: measure of oxygen saturation mortality: 1=Died in ICU, 0=Survived

### Step 1: Load needed libraries and implement the unsupervised analysis

Question 1: Name one potential risk/concern of obtaining data that has already been centered and scaled.

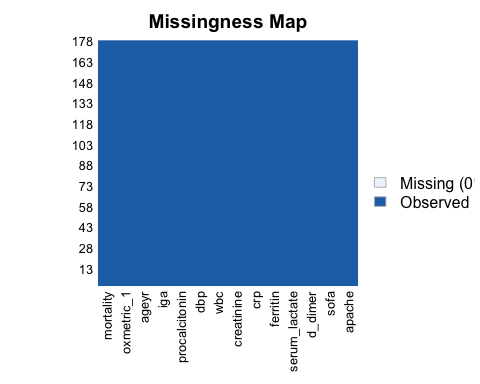
One potential concern with obtaining clean data is that we would be trusting whoever cleaned it that they had done this properly. I would want to see the code that they used to center and scale the data, to ensure that this needed to be done and that it was done appropriately.

Question 2: What unsupervised analysis do you think is appropriate for this research question? Justify your answer.

Hierarchical clustering would be the most appropriate for this particular research question, as we are trying to identify clusters. Other unsupervised methods, such as PCA, are inappropriate as we are not focused on dimension reduction.

Implement the unsupervised analysis you chose in Question 2, implementing appropriate analyses to determine the number of phenotypic clusters to keep within the analysis.

library(stats)  
library(factoextra)  
library(cluster)  
  
#Importing and recoding the data to be appropriate for the variable type  
  
covid.data = read\_csv("./data/covid.csv") %>%   
 janitor::clean\_names()  
  
#Checking whether there is missingness  
missmap(covid.data)



#Making sure data is actually scaled  
colMeans(covid.data, na.rm = TRUE)

## apache sofa d\_dimer serum\_lactate ferritin   
## -2.810252e-16 3.442939e-16 -2.836175e-16 6.928953e-16 2.583130e-16   
## crp creatinine wbc dbp procalcitonin   
## 1.949712e-16 1.304941e-17 -5.551895e-16 -3.983432e-16 -3.229314e-16   
## iga ageyr oxmetric\_1 mortality   
## -6.786862e-17 -7.906831e-16 2.583130e-16 1.741573e-01

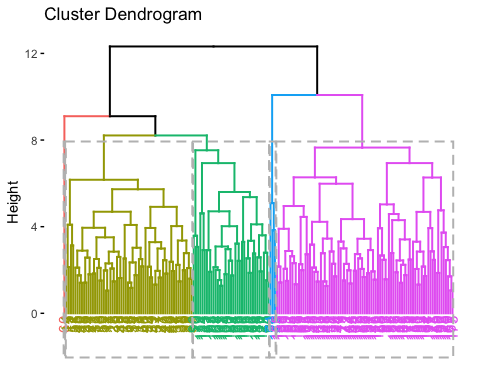
apply(covid.data, 2, sd, na.rm = TRUE)

## apache sofa d\_dimer serum\_lactate ferritin   
## 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000   
## crp creatinine wbc dbp procalcitonin   
## 1.0000000 1.0000000 1.0000000 1.0000000 1.0000000   
## iga ageyr oxmetric\_1 mortality   
## 1.0000000 1.0000000 1.0000000 0.3803145

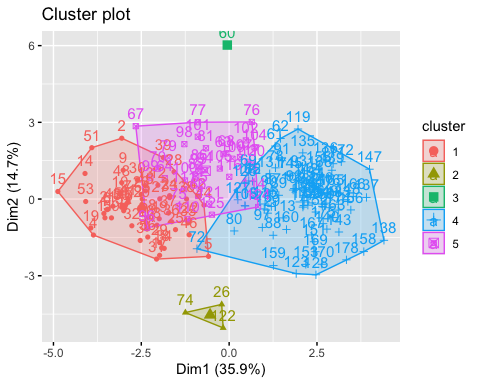
set.seed(100)  
  
clusters.hcut = hcut(covid.data, k = 5, hc\_func = "hclust", hc\_method = "complete", hc\_metric = "euclidian")  
  
clusters.hcut$size

## [1] 58 3 1 81 35

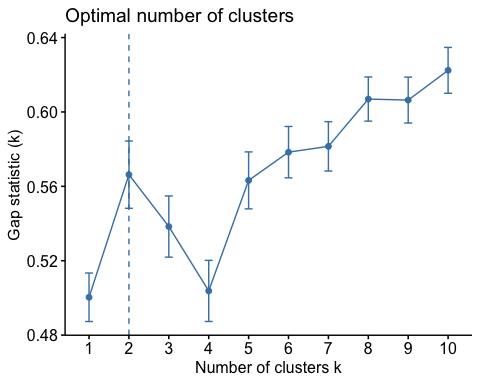
fviz\_dend(clusters.hcut, rect = TRUE)



fviz\_cluster(clusters.hcut)



gap\_stat = clusGap(covid.data, FUN = hcut, hc\_method = "complete", K.max = 10, B = 50)  
fviz\_gap\_stat(gap\_stat)



As per the gap statistic, the optimal number of clusters is **2 clusters**.

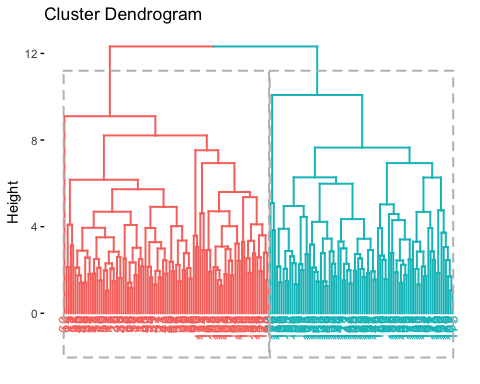
### Step 2: Interpret the clusters

Describe the clusters in terms of both their input features and the incidence of mortality within the cluster.

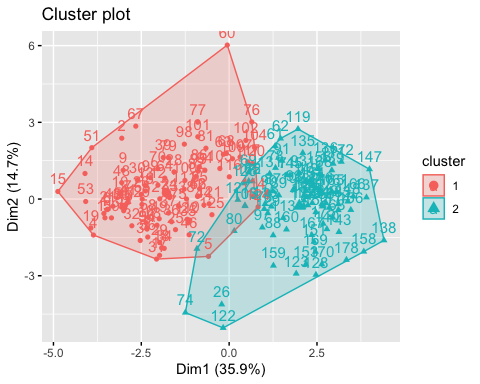
#Characterizing the clusters  
clusters.c = hcut(covid.data, k = 2, hc\_func = "hclust", hc\_method = "complete", hc\_metric = "euclidian")  
  
clusters.c$size

## [1] 94 84

fviz\_dend(clusters.c, rect = TRUE)



fviz\_cluster(clusters.c)



input.feature.vals = cbind(covid.data, cluster = clusters.c$cluster)  
input.feature.vals %>%  
 group\_by(cluster) %>%  
 summarise\_all(mean) %>%   
 knitr::kable()

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| cluster | apache | sofa | d\_dimer | serum\_lactate | ferritin | crp | creatinine | wbc | dbp | procalcitonin | iga | ageyr | oxmetric\_1 | mortality |
| 1 | 0.5154620 | -0.3715702 | 0.4715748 | -0.2273347 | -0.5983223 | 0.1350353 | 0.5007861 | 0.618920 | 0.6341271 | -0.5739372 | 0.6624438 | 0.1968494 | -0.5983223 | 0.2127660 |
| 2 | -0.5768265 | 0.4158047 | -0.5277147 | 0.2543984 | 0.6695511 | -0.1511110 | -0.5604035 | -0.692601 | -0.7096184 | 0.6422630 | -0.7413062 | -0.2202838 | 0.6695511 | 0.1309524 |

There is a higher incidence of mortality within the first cluster. Within this first cluster, APACHE II scores are higher on average (which is in line with what the scale assesses, so this finding in conjunction with higher mortality is reasonable). D-dimer levels, C-reactive protein levels, creatinine levels, white blood cell levels, diastolic blood pressure levels, immunoglobulin A levels, and age are all greater in the first cluster in comparison to the second cluster. In the second cluster, SOFA score, serum lactate levels, ferritin, procalcitonin, and oxygen saturation are all greater than in the first cluster.

Question 3: A researcher at a different medical institution has heard about your analysis and is interested in using your results to determine risk of mortality within their ICU. What are some limitations or concerns of using the results from your unsupervised analysis in a different setting?

They would need to consider some serious concerns and limitations. Firstly, the measurement tools used to collect this data may differ from their own measurement tools, and this could cause discrepancies. Additionally, they would need to consider differing patient compositions; seeing as this analysis was done with COVID patients, they would need to think about whether this is an appropriately similar patient pool to their own ICU. As this is a purely data-driven technique, the researchers need to consider the same data preparation steps that I’ve taken but apply it to their own data in a logical way.