Code Appendix

Taylor Keating, Zichen Liu, Niki Petrakos

December 17, 2021

Prepare the data

Read in the NHANES 2015-2016 biomarker and outcome data from .XPT files.

```
biomarkers <- read_xpt(paste0(getwd(), "/BIOPRO_I.XPT"))
outcome <- read_xpt(paste0(getwd(), "/BMX_I.XPT"))
```

Subset the outcome to only include ID and obesity variables, then create indicator variable for obesity.

```
outcome <- outcome %>% select("SEQN", "BMXBMI")
outcome$Obesity <- ifelse(outcome$BMXBMI >= 30, 1, 0)
```

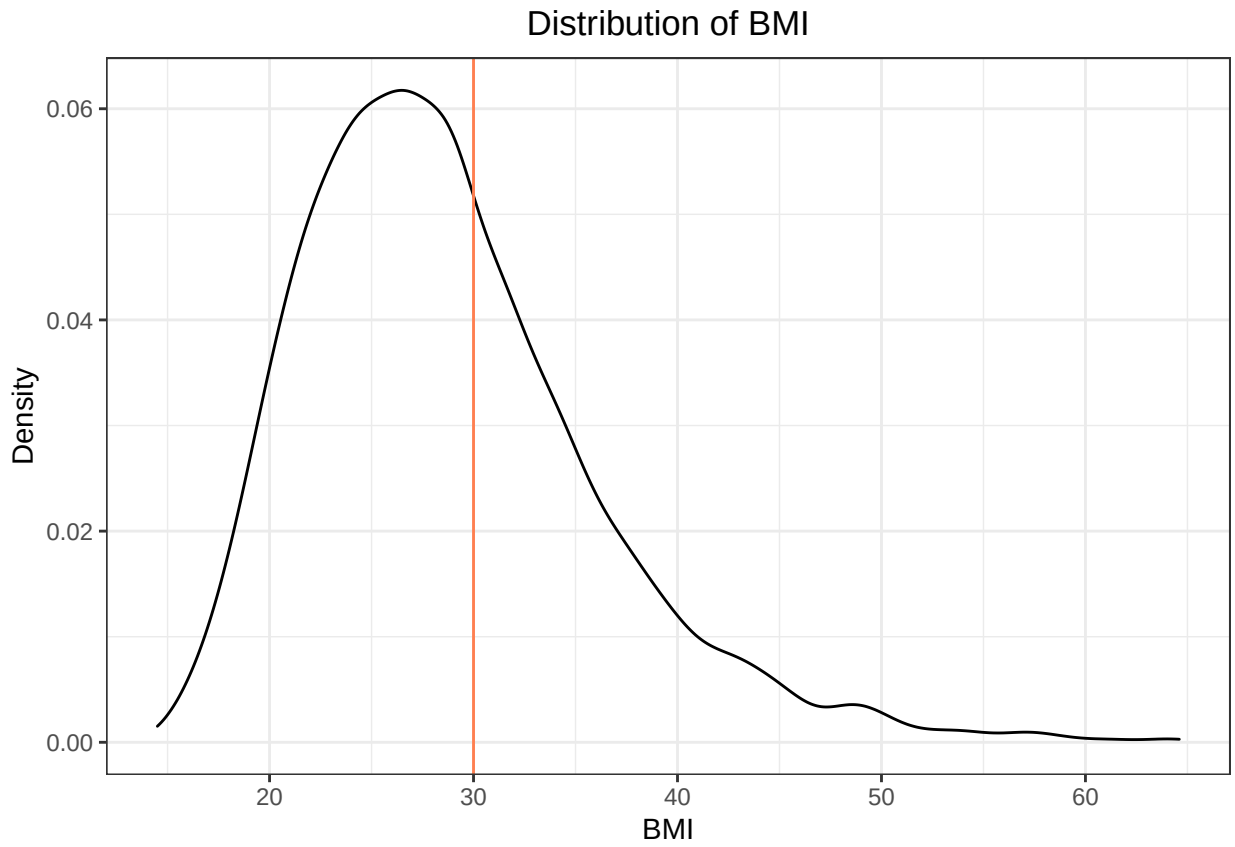
Merge outcome data and biomarker data, then exclude incomplete cases.

```
data <- inner_join(outcome, biomarkers, by = "SEQN")
data <- data[complete.cases(data),]
```

Exploratory analysis

Plot the distribution of BMI in our data.

```
ggplot(data = data, aes(x = BMXBMI)) + geom_density() +
  geom_vline(xintercept = 30, col = "coral") +
  labs(x = "BMI", y = "Density", title = "Distribution of BMI") +
  theme_bw() + theme(plot.title = element_text(hjust = 0.5))
```




```
prop_obese <- sum(data$Obesity == 1) / nrow(data)
```

In our dataset, 0.36 of the population are obese based on BMI. We can view a detailed breakdown of the population by weight status, determined by BMI cut-offs.

```
data <- data %>% mutate("Weight_Status" = case_when(BMXBMI < 18.5 ~ "Under",
                                                    BMXBMI >= 18.5 & BMXBMI < 25 ~ "Healthy",
                                                    BMXBMI >= 25 & BMXBMI < 30 ~ "Over",
                                                    BMXBMI >= 30 & BMXBMI < 40 ~ "Obese",
                                                    BMXBMI >= 40 ~ "Severe",
                                                    TRUE ~ "NA"))
data %>% group_by(Weight_Status) %>% summarize(N = n()) %>% kable()
```

Weight_Status	N
Healthy	1850
Obese	1788
Over	1867
Severe	454
Under	210

Screening-based method

Further subset the data to only columns required for implementing the screening-based method.

```
data_clean <- data %>% select(-c("SEQN", "BMXBMI", "Weight_Status"))
data_clean <- data_clean %>% select(Obesity, everything())
```

Rename all biomarker columns to the proper labels.

```
labels <- get_label(biomarkers)
labels <- c(Obesity = "Obesity", labels)
colnames(data_clean) <- lapply(1:ncol(data_clean), function(x) labels[names(data_clean)[x]][[1]])
```

Some biomarker columns are repeats but with different units in parentheses; remove these repeated columns.

```
colnames(data_clean) <- str_remove(colnames(data_clean), "\\s\\((.*)\\)")
data_clean <- data_clean[, !duplicated(colnames(data_clean))]
```

Additionally, the commas and spaces in the biomarker names must be removed for the linear models to be fit without errors.

```
colnames(data_clean) <- colnames(data_clean) %>%
  str_remove(",") %>%
  str_replace_all(" ", "_")
```

Next, split the dataset 50/50 into training and validation sets.

```
set.seed(1)
n <- nrow(data_clean)
train <- sample(n, size = floor(0.5*n), replace = FALSE)
data_train <- data_clean[train, ]
data_test <- data_clean[-train, ]
```

Define a function `calc_corr` to calculate the spearman correlation between obesity and one biomarker.

```
calc_corr <- function(biomarker, data){
  return(cor(data[biomarker+1], data[1], method="spearman"))
}
```

Apply the function to calculate pairwise-correlations between obesity and all biomarkers in the training set. Then, rank the biomarkers from largest to smallest correlations.

```
corrs <- do.call(rbind, lapply(1:(ncol(data_train)-1), calc_corr, data = data_train))
colnames(corrs)[1] <- "Correlations"
corrs_sort <- data.frame(correlation = corrs[order(-abs(corrs)),])
```

Define a function `calc_mse` that fits a linear model with the N top ranked biomarkers on the training set, then calculates the mean squared error when predicting the validation set.

```
calc_mse <- function(n, data_train, data_test){
  biomarkers <- paste0(rownames(corrs_sort)[1:n], collapse = "+")
  model <- lm(eval(parse(text = paste0("Obesity~", biomarkers))), data = data_train)
  pred <- predict(model, data_test)
  errors <- data_test$Obesity - pred
  return(mean(errors^2))
}
```

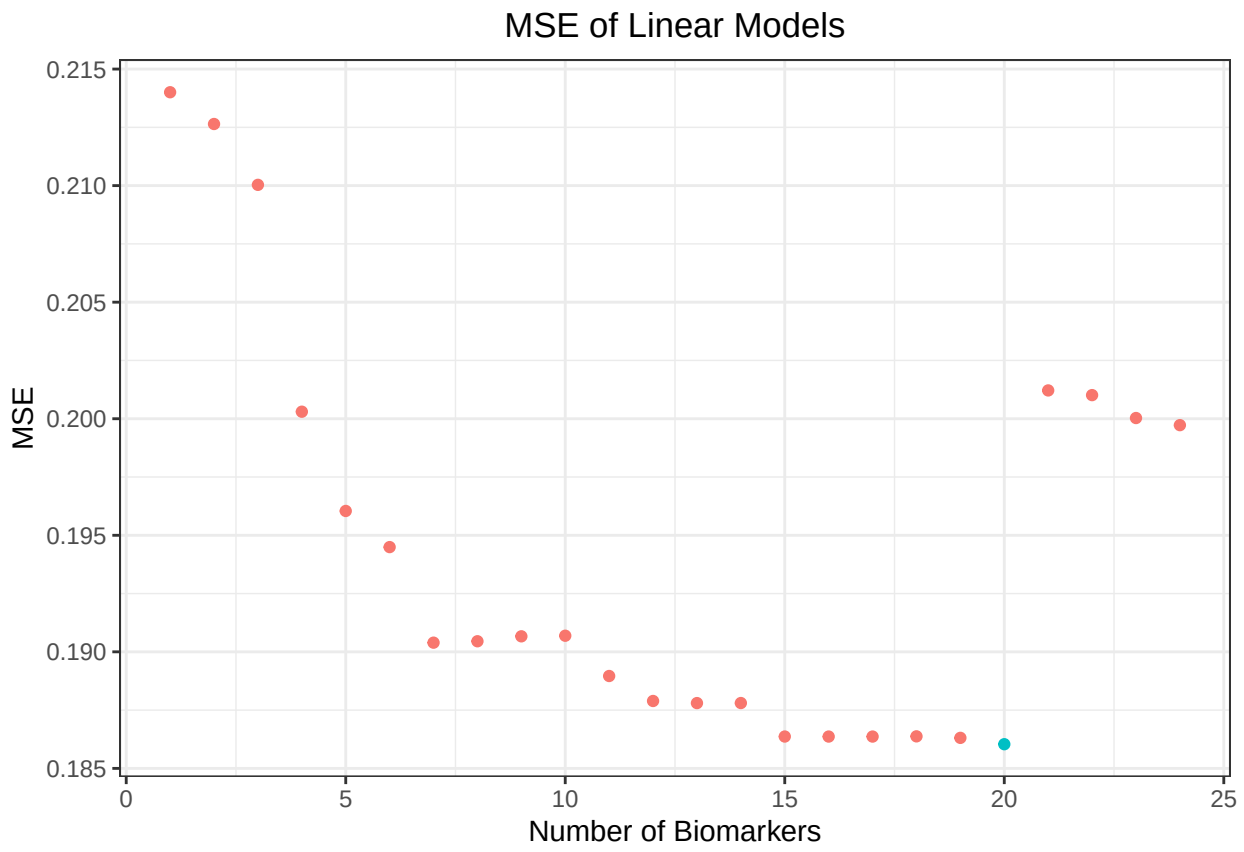
Apply the function to calculate the MSE for each fitted model from one biomarker to all biomarkers included.

```
mses <- do.call(rbind, lapply(1:(ncol(data_test) - 1), calc_mse,
                              data_train = data_train, data_test = data_test))
```

Display the calculated MSE's from each linear model graphically.

```
mse_df <- data.frame("Param" = 1:nrow(mses), "MSE" = mses)
mse_min <- which.min(mse_df$MSE)
mse_df$Min <- ifelse(mse_df$Param == mse_min, 1, 0)

ggplot(data = mse_df, aes(x = Param, y = MSE, col = as.factor(Min))) + geom_point() +
  labs(x = "Number of Biomarkers", y = "MSE", title = "MSE of Linear Models") +
  theme_bw() + theme(plot.title = element_text(hjust = 0.5), legend.position = "none")
```



The number of biomarkers that minimizes the MSE is 20.

```
final_biomarkers_screen <- paste(rownames(corrs_sort)[1:mse_min], collapse = ", ")
```

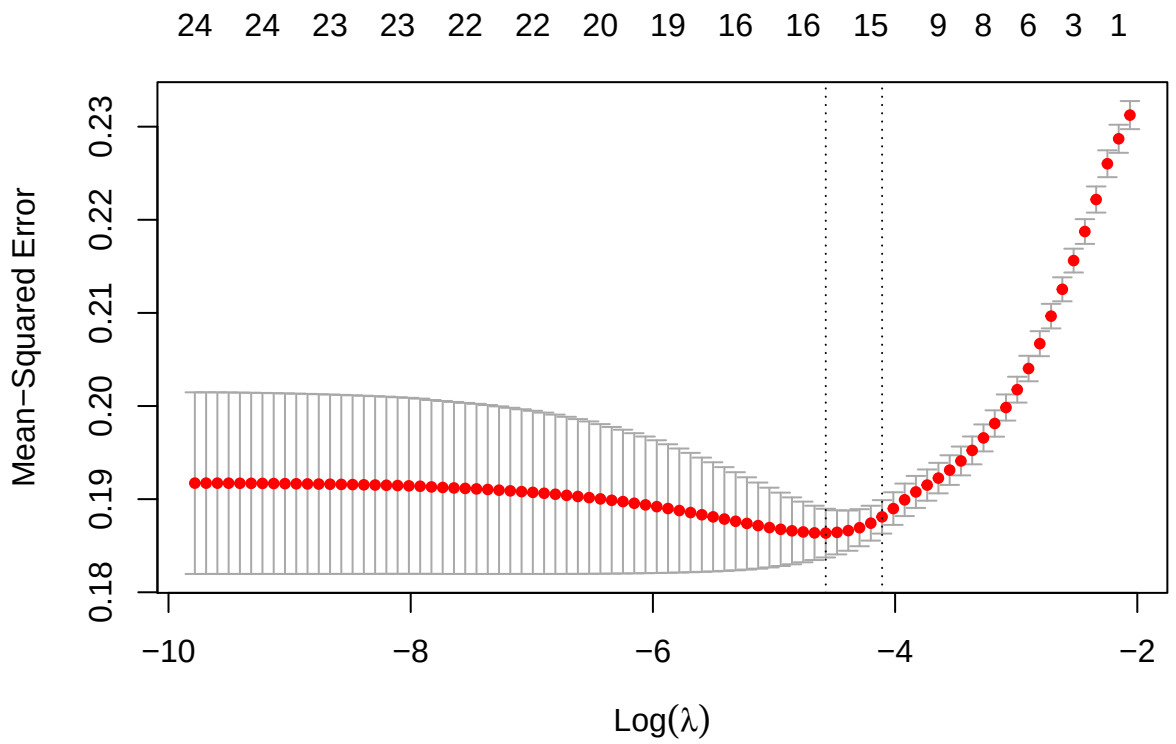
The 20 biomarkers that were chosen from the screening-based method to be most correlated with the outcome of obesity are Albumin_refrigerated_serum, Gamma_Glutamyl_Transferase, Triglycerides_refrig_serum, Uric_acid, Alanine_Aminotransferase, Glucose_refrigerated_serum, Iron_refrigerated_serum, Globulin, Lactate_Dehydrogenase, Total_Calcium, Phosphorus, Total_Bilirubin, Bicarbonate, Cholesterol_refrigerated_serum, Creatinine_refrigerated_serum, Osmolality, Creatine_Phosphokinase, Total_Protein, Chloride, Sodium.

Prediction-based methods

Lasso regression with linear models

We use `glmnet` for 10-fold cross-validation on 100 λ values to determine the tuning parameter that minimizes the MSE.

```
set.seed(456)
nlambda <- 100
fit_lasso <- cv.glmnet(x = as.matrix(data_clean)[,-1], y = as.matrix(data_clean)[,1],
                       alpha = 1, nlambda = nlambda)
plot(fit_lasso)
```



Get the coefficients produced by this λ to see which biomarkers were selected for.

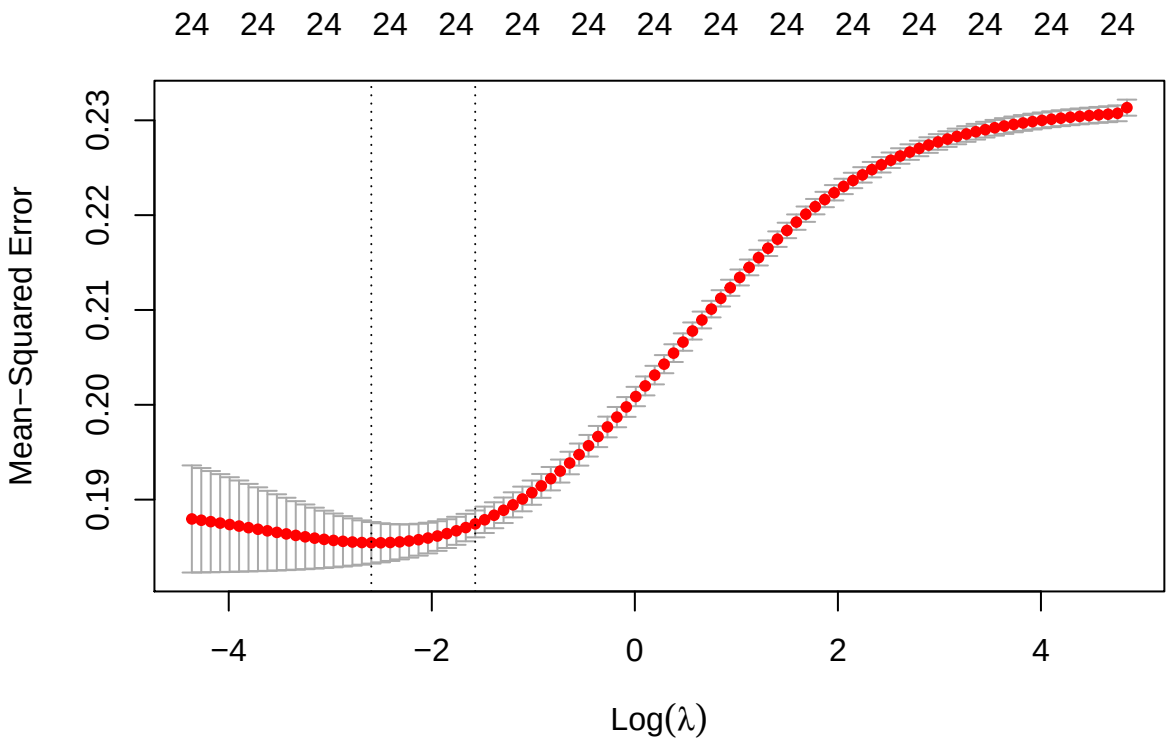
```
coef_lasso <- summary(coef(fit_lasso, s = "lambda.min"))
results_lasso <- lapply(list(coef_lasso$i), function(x) names(data_clean)[x])[[1]]
final_biomarkers_lasso <- paste(results_lasso[2:length(results_lasso)], collapse = ", ")
```

The 15 biomarkers that were chosen from using lasso regression with linear models to be most correlated with the outcome of obesity are Albumin_refrigerated_serum, Alkaline_Phosphatase, Aspartate_Aminotransferase, Alanine_Aminotransferase, Blood_Urea_Nitrogen, Bicarbonate, Creatinine_refrigerated_serum, Globulin, Glucose_refrigerated_serum, Iron_refrigerated_serum, Lactate_Dehydrogenase, Phosphorus, Total_Bilirubin, Triglycerides_refrig_serum, Uric_acid.

Ridge regression with linear models

We use `glmnet` for 10-fold cross-validation on 100 λ values to determine the tuning parameter that minimizes the MSE.

```
set.seed(789)
nlambda <- 100
fit_ridge <- cv.glmnet(x = as.matrix(data_clean)[,-1], y = as.matrix(data_clean)[,1],
                       alpha = 0, nlambda = nlambda)
plot(fit_ridge)
```



Get the coefficients produced by this λ to see which biomarkers were selected for.

```
coef_ridge <- summary(coef(fit_ridge, s = "lambda.min"))
results_ridge <- lapply(list(coef_ridge$i), function(x) names(data_clean)[x])[[1]]
final_biomarkers_ridge <- paste(results_ridge[2:length(results_ridge)], collapse = ", ")
```

The 24 biomarkers that were chosen from using ridge regression with linear models to be most correlated with the outcome of obesity are Albumin_refrigerated_serum, Alkaline_Phosphatase, Aspartate_Aminotransferase, Alanine_Aminotransferase, Blood_Urea_Nitrogen, Bicarbonate, Total_Calcium, Cholesterol_refrigerated_serum, Creatine_Phosphokinase, Chloride, Creatinine_refrigerated_serum, Globulin, Glucose_refrigerated_serum, Gamma_Glutamyl_Transferase, Iron_refrigerated_serum, Potassium, Lactate_Dehydrogenase, Sodium, Osmolality, Phosphorus, Total_Bilirubin, Total_Protein, Triglycerides_refrig_serum, Uric_acid.

Lasso regression with polynomial models

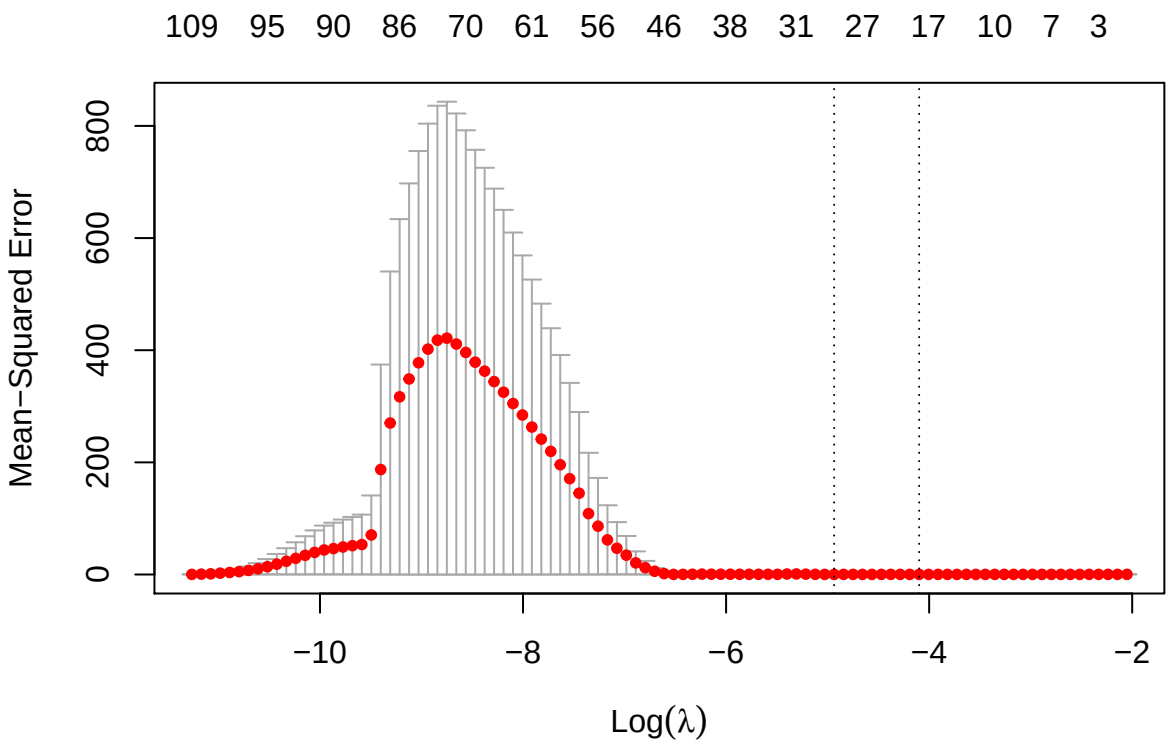
Add polynomial terms up to 5 degree for each biomarker.

```
max_degree <- 5
selected_matrix <- cbind(data_clean$Obesity,
                         matrix(apply(data_clean[, -1], 2,
                                      FUN = poly, degree = max_degree, raw = T),
                                nrow = nrow(data_clean),
                                byrow = F))

colnames(selected_matrix)<- c("Obesity", sapply(names(data_clean[, -1]),
                                                FUN = paste, 1:max_degree, sep = "_poly_"))
```

We use `glmnet` for 10-fold cross-validation on 100 λ values to determine the tuning parameter that minimizes the MSE.

```
set.seed(123)
nlambda <- 100
fit_poly <- cv.glmnet(x = selected_matrix[, -1], y = selected_matrix[, 1],
                      alpha = 1, nlambda = nlambda)
plot(fit_poly)
```



Get the coefficients produced by this λ to see which biomarkers were selected for.

```
coef_poly <- summary(coef(fit_poly, s = "lambda.min"))
results_poly <- lapply(list(coef_poly$i), function(x) colnames(selected_matrix)[x])[[1]]
final_biomarkers_poly <- paste(results_poly[2:length(results_poly)], collapse = ", ")
```

The 26 biomarkers and their polynomials that were chosen from using lasso regression with polyomials models to be most correlated with the outcome of obesity are Albumin_refrigerated_serum_poly_2, Albumin_refrigerated_serum_poly_3, Alkaline_Phosphatase_poly_1, Alkaline_Phosphatase_poly_5, Aspartate_Aminotransferase_poly_1, Aspartate_Aminotransferase_poly_3, Aspartate_Aminotransferase_poly_4, Aspartate_Aminotransferase_poly_5, Alanine_Aminotransferase_poly_1, Alanine_Aminotransferase_poly_3, Blood_Urea_Nitrogen_poly_1, Bicarbonate_poly_1, Cholesterol_refrigerated_serum_poly_5, Creatinine_refrigerated_serum_poly_1, Globulin_poly_1, Glucose_refrigerated_serum_poly_1, Glucose_refrigerated_serum_poly_4, Gamma_Glutamyl_Transferase_poly_5, Iron_refrigerated_serum_poly_1, Potassium_poly_1, Lactate_Dehydrogenase_poly_1, Phosphorus_poly_1, Total_Bilirubin_poly_1, Triglycerides_refrig_serum_poly_1, Triglycerides_refrig_serum_poly_2, Uric_acid_poly_1.