Introduction

This report addresses an important question in statistical learning: how can we best predict body fat percentage using various physical measurements? The dataset "fat.csv" from the faraway library in R contains 252 observations with 18 variables, including body fat percentage calculated via Brozek's equation as the response variable brozek.

Predicting body fat percentage is critical in medical, fitness, and health-related fields where body composition plays a vital role in determining an individual's health status. However, identifying the most relevant predictors and selecting appropriate modeling techniques is key to ensuring accurate and interpretable predictions. Throughout the assignment, various regression techniques will be employed, including:

- Linear regression with all predictors.
- Linear regression with the best subset of k = 5 predictor variables
- Linear regression with variables (stepwise) selected using AIC
- Regularization methods such as Ridge and LASSO regression
- Principal Component Regression (PCR) and Partial Least Squares (PLS), both dimensionality reduction techniques

Why should we care about this problem? Body fat percentage is a crucial metric for understanding a person's overall health. Effective prediction models can lead to more accurate assessments in clinical and fitness settings. Moreover, the methodological approaches tested here go beyond standard linear regression, offering insights into when and how to apply advanced techniques to improve prediction accuracy and model stability, especially when multicollinearity or dimensionality issues are present.

This report provides valuable insights by comparing multiple modeling techniques, demonstrating when complex methods like LASSO or PCR outperform traditional regression, and revealing the practical importance of cross-validation in model evaluation. The findings can be extended to broader fields, enhancing how we approach regression problems in various real-world applications, from healthcare to finance.

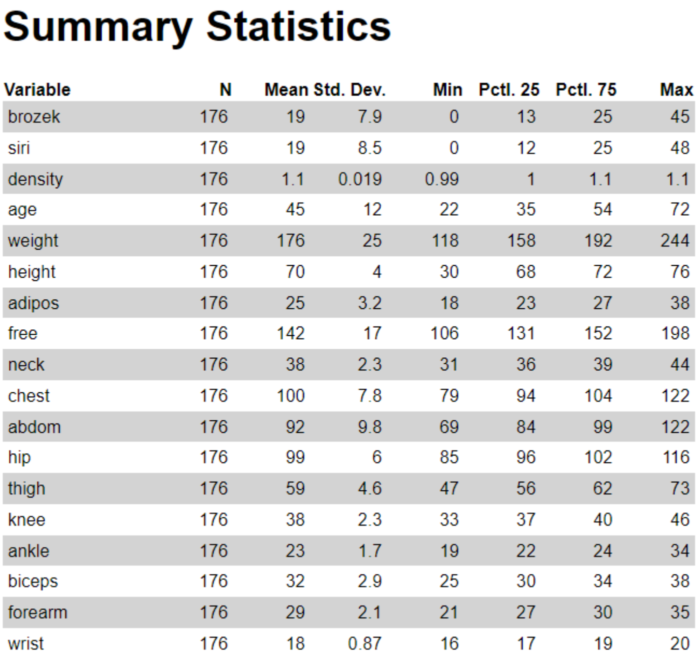
By incorporating these advanced methods and validation techniques, this analysis highlights the significance of model selection and performance evaluation, offering a comprehensive framework for tackling complex prediction problems.

Exploratory Data Analysis

The "fat" dataset contains 252 observations and 18 variables. The response variable **brozek** represents the body fat percentage calculated using **Brozek's equation**, with the remaining 17 variables as potential predictors. From basic exploratory data analysis, all variables are in numerical format and have no missing values. Next, the dataset is split into 70% training and 30% testing data. The training data now consists of 176 observations and 76 observations in the testing data. The variable names and their data types are shown below using the **str()** method on the full dataset:

```
'data.frame': 252 obs. of  18 variables:

 $ brozek : num  12.6 6.9 24.6 10.9 27.8 20.6 19 12.8 5.1 12 ...

 $ siri: num  12.3 6.1 25.3 10.4 28.7 20.9 19.2 12.4 4.1 11.7 ...

 $ density: num  1.07 1.09 1.04 1.08 1.03 ...

 $ age: int  23 22 22 26 24 24 26 25 25 23 ...

 $ weight: num  154 173 154 185 184 ...

 $ height: num  67.8 72.2 66.2 72.2 71.2 ...

 $ adipos: num  23.7 23.4 24.7 24.9 25.6 26.5 26.2 23.6 24.6 25.8 ...

 $ free: num  135 161 116 165 133 ...

 $ neck: num  36.2 38.5 34 37.4 34.4 39 36.4 37.8 38.1 42.1 ...

 $ chest: num  93.1 93.6 95.8 101.8 97.3 ...

 $ abdom: num  85.2 83 87.9 86.4 100 94.4 90.7 88.5 82.5 88.6 ...

 $ hip: num  94.5 98.7 99.2 101.2 101.9 ...

 $ thigh: num  59 58.7 59.6 60.1 63.2 66 58.4 60 62.9 63.1 ...

 $ knee: num  37.3 37.3 38.9 37.3 42.2 42 38.3 39.4 38.3 41.7 ...

 $ ankle: num  21.9 23.4 24 22.8 24 25.6 22.9 23.2 23.8 25 ...

 $ biceps: num  32 30.5 28.8 32.4 32.2 35.7 31.9 30.5 35.9 35.6 ...

 $ forearm: num  27.4 28.9 25.2 29.4 27.7 30.6 27.8 29 31.1 30 ...

 $ wrist: num  17.1 18.2 16.6 18.2 17.7 18.8 17.7 18.8 18.2 19.2 ...
```

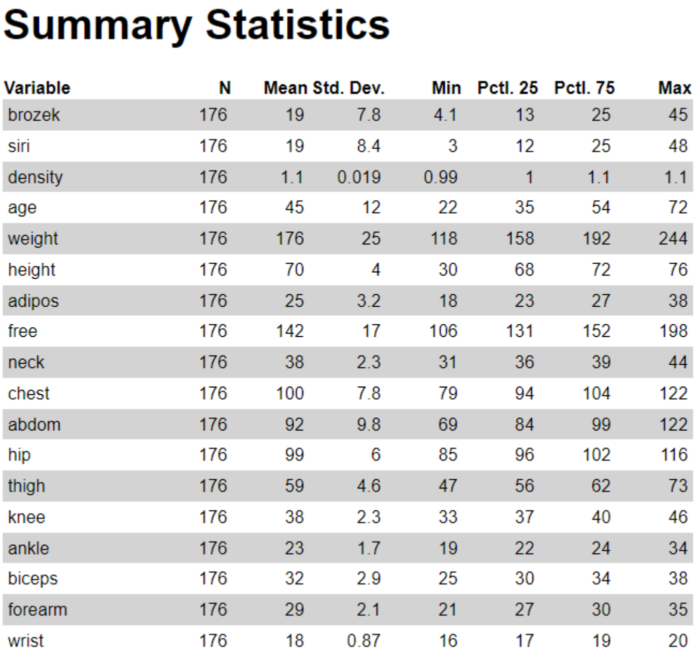
Since all variables are numerical, the statistics summary of the training dataset might be useful for finding any weird patterns in the data. After printing the statistics summary using summary(), one interesting fact is that there are values 0 in both brozek and siri, which shouldn't happen because these variables represent the body fat percentage. From both a biological and dataset-driven perspective, body fat percentage cannot be 0. There will always be some essential fat in the body. Since both brozek = 0 and siri = 0 happened in the same observation of row 182, this could be a potential data entry error. Hence, it is safe to fix it using mean imputation, which replaces the 0 with the mean of the column. See Tables 1 and 2 for a full statistical summary of the training dataset before and after the mean imputation.



Summary Statistics

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
brozek	176	19	7.9	0	13	25	45
siri	176	19	8.5	0	12	25	48
density	176	1.1	0.019	0.99	1	1.1	1.1
age	176	45	12	22	35	54	72
weight	176	176	25	118	158	192	244
height	176	70	4	30	68	72	76
adipos	176	25	3.2	18	23	27	38
free	176	142	17	106	131	152	198
neck	176	38	2.3	31	36	39	44
chest	176	100	7.8	79	94	104	122
abdom	176	92	9.8	69	84	99	122
hip	176	99	6	85	96	102	116
thigh	176	59	4.6	47	56	62	73
knee	176	38	2.3	33	37	40	46
ankle	176	23	1.7	19	22	24	34
biceps	176	32	2.9	25	30	34	38
forearm	176	29	2.1	21	27	30	35
wrist	176	18	0.87	16	17	19	20

Table 1: Summary Statistics Before Mean Imputation



Summary Statistics

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
brozek	176	19	7.8	4.1	13	25	45
siri	176	19	8.4	3	12	25	48
density	176	1.1	0.019	0.99	1	1.1	1.1
age	176	45	12	22	35	54	72
weight	176	176	25	118	158	192	244
height	176	70	4	30	68	72	76
adipos	176	25	3.2	18	23	27	38
free	176	142	17	106	131	152	198
neck	176	38	2.3	31	36	39	44
chest	176	100	7.8	79	94	104	122
abdom	176	92	9.8	69	84	99	122
hip	176	99	6	85	96	102	116
thigh	176	59	4.6	47	56	62	73
knee	176	38	2.3	33	37	40	46
ankle	176	23	1.7	19	22	24	34
biceps	176	32	2.9	25	30	34	38
forearm	176	29	2.1	21	27	30	35
wrist	176	18	0.87	16	17	19	20

Table 2: Summary Statistics After Mean Imputation

Figure 2 below displays the shape of the distribution. The training data is visualized as multiple box plots across all variables. From the plot, there is an extreme outlier with height = 29.5 inches. After carefully inspecting this unusual data point, the data showed that this 44-year-old individual is roughly 74.93 cm tall and weighed 205 lbs, with a neck of 36.6 cm long and a hip at 115.5 cm high, which is taller than the individual itself. There is enough evidence to conclude that this data point is due to some data entry error. After careful consideration, it is best to remove this data point from the data. See Figure 2 below for the updated box plots.

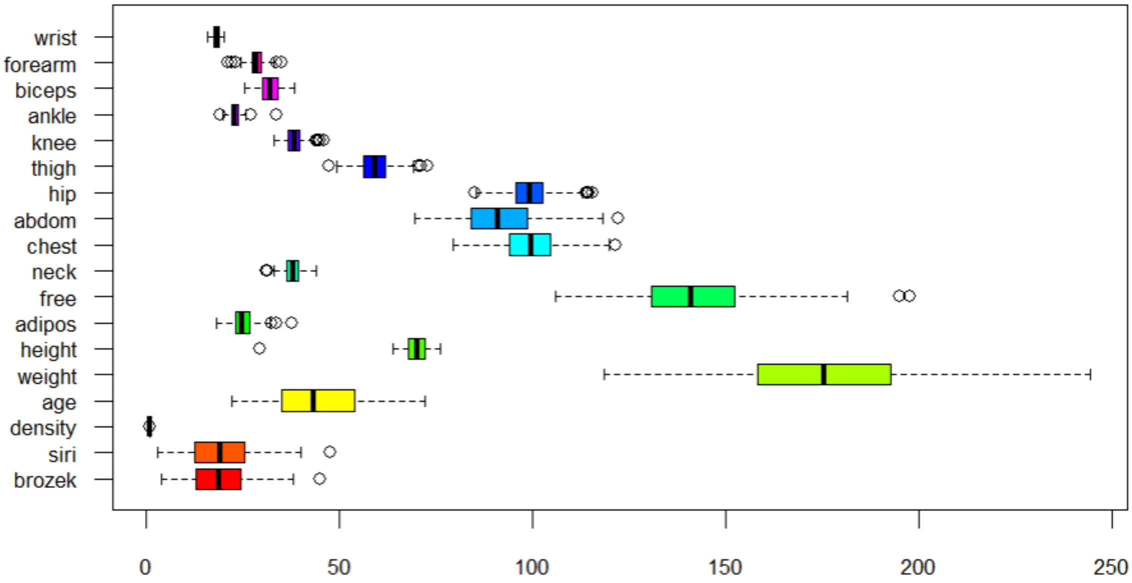


Figure 1: Multiple Boxplots Across All Features

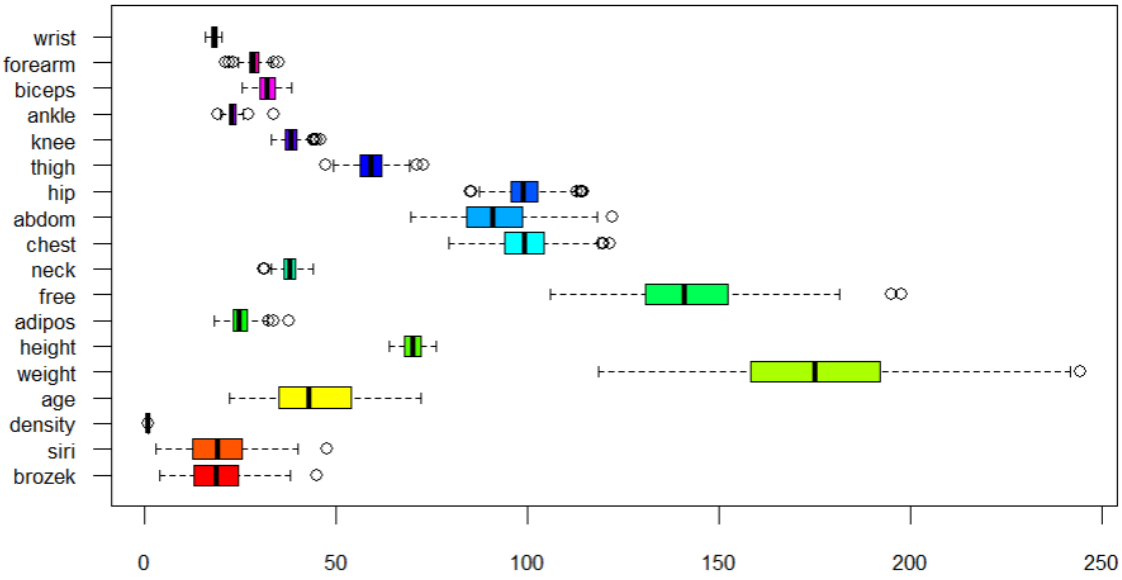


Figure 2: Updated Multiple Boxplots Across All Features

All other observations seem reasonable. In the case of body fat analysis, extreme values may not always be invalid. If the dataset contains individuals with very high or very low body fat percentages, these outliers could represent specific population groups (e.g., athletes or individuals with specific medical conditions). Hence, it is reasonable to keep the rest of the data points for further analysis.

Furthermore, some highly correlated variables are observed in the training data from the correlation matrix. The response variable brozek is positively correlated with variables like siri, weight, adipos, chest, and abdom. This suggests that, as these variables increase, brozek also tends to increase, indicating that they likely have a strong impact on predicting brozek (body fat percentage). Similarly, brozek is negatively correlated with density, which suggests that as density increases, brozek decreases. This is expected since density and body fat percentage are often inversely related. The correlation matrix is shown below in Figure 3:

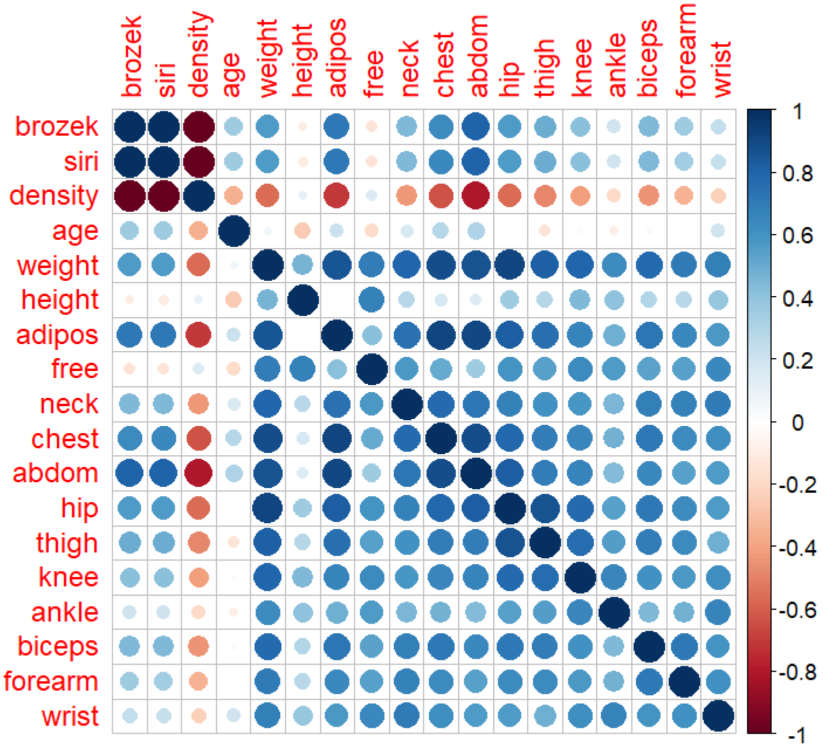


Figure 3: Correlation Matrix

One interesting fact is that the correlation pattern of siri mirrors that of brozek with other variables. This is because the Siri equation is another method for calculating body fat percentage, and its values closely follow those of brozek. Hence, the correlation patterns are almost identical. There is potential issue of multicollinearity among the predictor variables, such as weight, adipos, chest, abdom, and hip, which are highly correlated with each other. This could lead to multicollinearity problems in regression models, where it becomes difficult to separate the individual effects of these variables on the response variable. Conducting a Variance Inflation Factor (VIF) test is an appropriate way to assess multicollinearity. If multicollinearity is truly a problem in our model, we can perform regularization or dimensional reduction techniques to deal with it.

Methods

The following models are implemented to predict the body fat percentage (response variable brozek). The mean squared error (MSE) was calculated to assess the predictive accuracy of the model on the test data. Monte Carlo Cross-Validation algorithm was repeated 100 times to compute and compare the “average” performances of each model mentioned. See the performance of each model and cross-validation results in the Results Section.

(i) Linear Regression with All Predictors

A standard linear regression model is built to include all predictor variables. This served as a baseline model, allowing us to understand the relationships between the response variable (brozek) and all predictors without regularization or feature selection.

(ii) Linear Regression with the Best Subset of k = 5 Predictors

The second model used the best subset selection method to find the optimal subset of 5 predictors. This method exhaustively searches through all combinations of predictor variables and selects the subset that minimizes the model's AIC. The regsubsets() function from the leaps package was used for subset selection, the parameter nvmax was set to 5 to choose the best subset of k = 5 predictors. The 5 predictors chosen were siri+knee+wrist+thigh+age.

(iii) Linear regression with variables (stepwise) selected using AIC

The stepwise selection, which iteratively adds or removes variables based on their contribution to the model, reduces the number of predictors and builds a parsimonious model. Both forward and backward selection methods were tried, and the one with the lowest AIC was selected. The stepwise selection was carried out using the stepAIC function from the MASS package with direction = "both".

(iv) Ridge Regression

Ridge regression was used to mitigate multicollinearity by applying L2 regularization. The regularization parameter, lambda (λ), was selected to minimize the Generalized Cross Validation (GCV) score. The lm.ridge() function from the MASS package was used to fit the model, and the optimal λ was chosen from a grid of values ranging from 0 to 100.

(v) LASSO Regression

LASSO applies L1 regularization to shrink coefficients and perform variable selection. The lars() function was used for fitting the LASSO model.

(vi) Principal Component Regression (PCR)

Principal Component Regression (PCR) was applied to address multicollinearity by transforming the original predictors into a set of orthogonal principal components. The PCR model was fitted using cross-validation to determine the optimal number of components that minimize the adjusted validation error. All 17 components were selected.

(vii) Partial Least Squares (PLS) Regression

PLS regression was also implemented to deal with multicollinearity by reducing the predictor space while keeping the components relevant to the response variable. The number of PLS components was selected via cross-validation. Like PCR, all 17 components were selected.

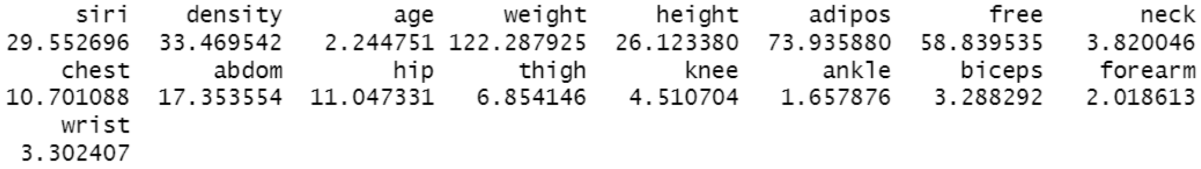
The above 7 models were repeated 100 times in the cross-validation to find the average MSE performance.

Results

The table below provides a summary of the performance of various regression models implemented to predict body fat percentage. The models were evaluated using Mean Squared Error (MSE), Average MSE (across 100 Monte Carlo Cross Validations), and the variance of MSE, as follows:

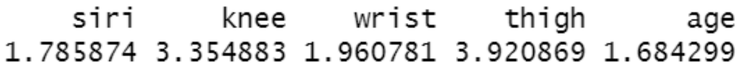
Model	MSE	AVG MSE	Variance of MSE	Predictors Selected
Linear Reg Full	0.04650	0.09794975	0.031525815	All 17 Predictors
Linear Reg k = 5	0.05060	0.03490135	0.0004733885	siri+knee+wrist+thigh+age
Linear Reg Stepwise	0.04671	0.10173742	0.038722597	siri + weight + height + free + knee + wrist
Ridge Reg	0.04639	0.10079577	0.035145436	All 17 Predictors
Lasso Reg	0.04923	0.08642128	0.027429771	All 17 Predictors
PCR	0.04650	0.09794975	0.031525815	All 17 Components
PLSR	0.04650	0.09794975	0.031525815	All 17 Components

Table 3: Model Performance



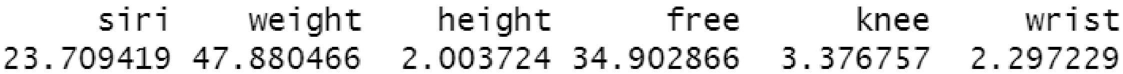
```
     siri   density       age    weight    height    adipos      free      neck 
29.552696 33.469542  2.244751 122.287925 26.123380 73.935880 58.839535  3.820046 
    chest     abdom       hip     thigh      knee     ankle    biceps   forearm 
10.701088 17.353554 11.047331  6.854146  4.510704  1.657876  3.288292  2.018613 
    wrist 
 3.302407 
```

Figure 4: Model 1 Full Linear Reg VIF



```
    siri     knee    wrist    thigh      age 
1.785874 3.354883 1.960781 3.920869 1.684299 
```

Figure 5: Model 2 Linear Reg with k = 5 VIF



```
     siri    weight    height      free      knee     wrist 
23.709419 47.880466  2.003724 34.902866  3.376757  2.297229 
```

Figure 6: Model 3: Stepwise Regression VIF

- **Full Linear Regression & PCR & PLSR**: Due to all 17 predictors being selected in the three models, the PCR and PLSR model is equivalent to the full regression model and the three models yielded the same results. Since the full model exhibits multicollinearity, PCR and PLSR here will exhibit the same issues.
- **Linear Regression (k = 5 Subset Model)**: The subset model with only 5 predictors (siri, knee, wrist, thigh, and age) showed the lowest AVG MSE and least variance across cross-validation runs, indicating a more stable model. The model does not exhibit multicollinearity issues.
- **Stepwise Linear Regression**: The stepwise model selected variables using AIC, has the highest AVG MSE (0.10173742), and showed the largest variance (0.038722597) across cross-validations, highlighting some instability in performance. Thus, not the best model. The model also has multicollinearity issues checked using VIF method. The predictors siri (VIF = 23.71), weight (VIF = 47.88), and free (VIF = 34.90) have very high VIF values, indicating substantial multicollinearity.
- **Lasso Regression**: Lasso regression has a balanced predictive performance with a slightly reduced average MSE and variance MSE compared to the full and Ridge model. However, the model may not eliminate multicollinearity, since all 17 predictors were selected in the Lasso Model.

Findings/Conclusions

The analysis revealed several key insights about the predictive modeling of body fat percentage based on the dataset. Initially, the Exploratory Data Analysis (EDA) suggested potential multicollinearity between the predictor variables, which motivated the use of regularized regression models like Ridge and LASSO, as well as dimensionality reduction techniques such as Principal Component Regression (PCR) and Partial Least Squares (PLS). The inclusion of multicollinearity reduction approaches was necessary, given the correlation between variables like weight, adipos, chest, abdom, and hip.

Since the number of components used in PCR or PLSR is equal to the number of predictors (k = 17), then the model becomes equivalent to the full linear regression model, and thus it will exhibit the same issues, including multicollinearity. This is because no dimensionality reduction or regularization occurs when all the original predictors are retained. In other words, PCR and PLSR lose their advantage of handling multicollinearity when k = p, and the collinearity among the predictors remains an issue in the same way as in the full regression model.

While the Ridge model did not outperform the full model in terms of average MSE, it is designed to handle multicollinearity effectively. LASSO was successful in reducing the average MSE during the cross-validation. However, LASSO selected all 17 variables which could raise the same problem in multicollinearity as appeared in the full model.

Variable selection proved essential for improving model stability. The best subset selection for k = 5 predictors was notably more stable across cross-validations, with the lowest average MSE and variance of MSE. The results suggest that the best subset model offers the best trade-off between prediction accuracy and model stability for this dataset. Moreover, the best subset model with k = 5 predictors demonstrated that a simpler model can achieve comparable accuracy while enhancing stability, which could be preferable for more interpretable models. Multicollinearity issues do not exhibit in the best subset model.

In terms of the worst performer, stepwise regression models had the highest average MSE and high variance of MSE, highlighting some instability in performance. Multicollinearity does exist in the stepwise regression model.

One caveat of this analysis is that it assumes the regression assumptions hold true for all models, including linearity, homoscedasticity, independence of errors, and normality of residuals. Thus, no tests or diagnostics, such as residual analysis, normality tests (e.g., Shapiro-Wilk), or checks for homoscedasticity were conducted after the models were built. This may affect the validity of the models, as unverified assumptions could lead to biased estimates or inaccurate predictions in real-world datasets.

From a real-world perspective, the findings highlight the importance of selecting appropriate modeling techniques when dealing with high-dimensional and correlated data, such as body measurements. The analysis suggests that when building predictive models for body fat estimation, both regularization techniques and variable selection methods should be considered to balance model complexity with predictive performance.

Appendix

```
library(car)
library(caTools)
library(corrplot)
library(vtable)
library(dplyr)
library(leaps)
library(MASS)
## EDA
data <- read.table(file = "fat.csv", sep=",", header=TRUE)
head(data)
str(data)
#make this example reproducible
set.seed(1)
#use 70% of dataset as training set and 30% as test set
sample <- sample.split(data$brozek, SplitRatio = 0.7)
train  <- subset(data, sample == TRUE)
test   <- subset(data, sample == FALSE)
head(train)
#check null values
sum(is.na(train))
## Correlation Matrix
mydata.cor = cor(train, method = c("spearman"))
corrplot(mydata.cor)
## Statistical Summary table
st(train)
# Impute mean values for brozek and siri wherever they are 0
```

```r
train <- train %>%
  mutate(brozek = ifelse(brozek == 0, mean(brozek[brozek != 0], na.rm = TRUE), brozek),
       siri = ifelse(siri == 0, mean(siri[siri != 0], na.rm = TRUE), siri))
## Multiple Boxplot
stacked_df <- stack(train)
head(stacked_df)
par(mar = c(5, 7, 4, 2))
boxplot(train, col = rainbow(ncol(train)), horizontal = TRUE,las = 1, cex.axis = 0.7)
## Remove extreme data point
train <- train[train$height != 29.5, ]
## Linear Regression
mod1 <- lm(brozek~., data = train)
summary(mod1)
pred1 = predict(mod1,test[,2:18])
MSE1 = mean((pred1-test[,1])^2)
MSE1
vif(mod1)
## Subset Regression with k = 5
mod2 <- regsubsets(brozek~., data = train, nvmax = 5)
summary(mod2)
mod2 <- lm(brozek~siri+knee+wrist+thigh+age, data = train)
pred2 = predict(mod2,test[,2:18])
MSE2 = mean((pred2-test[,1])^2)
MSE2
vif(mod2)
## StepAIC
mod3 <- stepAIC(mod1, direction = "both")
pred3 = predict(mod3,test[,2:18])
MSE3 = mean((pred3-test[,1])^2)
```

```
MSE3
vif(mod3)
## Ridge Regression
mod4 <- lm.ridge(brozek~., data = train, lambda = seq(0,100,.001))
select(mod4)
coef <-mod4$coef[,which.min(mod4$GCV)]
# scale(test[,2:18], center = F, scale = mod4$scales) : The scale parameter is used to standardize the test predictors using the scaling factors (mod4$scales) obtained from the lm.ridge model on the training data. This ensures that the test data is scaled in the same way as the training data was.
# %*%coef: This multiplies the scaled test predictors by the regression coefficients (coef) obtained from the lm.ridge model. This gives the predicted values before adjusting for the intercept.
# The entire expression mod4$ym - sum(mod4$xm * (coef / mod4$scales)) computes the intercept term for the predictions, adjusting for the scaling.
pred4 <- scale(test[,2:18], center = F, scale = mod4$scales) %*% coef + (mod4$ym-sum(mod4$xm*(coef/mod4$scales)))
MSE4 <- mean((pred4-test[,1])^2)
MSE4
## Lasso Regression
library(lars)
mod5 <- lars(as.matrix(train[,2:18]),train[,1], type = "lasso", trace = TRUE)
lamb <- mod5$lambda[which.min(summary(mod5)$Cp)]
pred5 <- predict(mod5,test[,-1],s=lamb,type="fit", mode="lambda")
MSE5 <- mean((pred5$fit - test[,1])^2)
MSE5
# Extract the coefficients
coef(mod5)
## PCR
library(pls)
mod6 <- pcr(brozek~., data=train, validation="CV")
which.min(mod6$validation$adj)
```

```
summary(mod6)
pred6 = predict(mod6, test[,2:18], ncomp = 17)
MSE6 <- mean((pred6 - test[,1])^2)
MSE6


## PLS
mod7 <- plsr(brozek~., data=train, validation="CV")
which.min(mod7$validation$adj)
summary(mod7)
pred7 <- predict(mod7, test[,2:18], ncomp=17)
MSE7 <- mean((pred7 - test[,1])^2)
MSE7
# Combine all MSE
TEALL1<- round(c(MSE1,MSE2,MSE3,MSE4,MSE5,MSE6,MSE7), 5)
TEALL1
# Cross Validation
set.seed(1)
TEALL = NULL;
B= 100
for (b in 1:B){
  sample <- sample.split(data$brozek, SplitRatio = 0.7)
  train  <- subset(data, sample == TRUE)
  test   <- subset(data, sample == FALSE)
  # Linear regression with all predictors
  model1 = lm(brozek~., data=train)
  pred1 = predict(model1,test[,2:18])
  te1 <- mean((pred1 - test[,1])^2)
  # Linear regression with best subset k = 5
  model2 <- regsubsets(brozek~., data = train, nvmax = 5)
```

```
 summary(model2)
 model2 <- lm(brozek~siri+knee+wrist+thigh+age, data = train)
 pred2 = predict(model2, test[,2:18])
 te2 <- mean((pred2 - test[,1])^2)

 # Stepwise
 model3 <- step(model1)
 pred3 = predict(model3, test[,2:18])
 te3 <- mean((pred3 - test[,1])^2)
 # Ridge Regression
 model4 <- lm.ridge(brozek~., data=train, lambda=seq(0,100,.001))
 select(model4)
 coef <-model4$coef[,which.min(model4$GCV)]
 pred4 <- scale(test[,2:18], center=F, scale=model4$scales)%*%coef+
(model4$ym - sum(model4$xm *(coef/model4$scales) ))
 te4 <- mean((pred4 - test[,1])^2)
 # LASSO
 model5 <- lars(as.matrix(train[,2:18]),train[,1],type="lasso",trace=TRUE)
 lambdaX <- model5$lambda[which.min(summary(model5)$Cp)]
 pred5 <-predict(model5, test[,-1],s=lambdaX,type="fit", mode="lambda")
te5 <- mean((pred5$fit - test[,1])^2)
 # Principal Component Regression
 model6 <- pcr(brozek~., data=train, validation="CV")
 pred6 = predict(model6, test[,2:18], ncomp = 17)
 te6 <- mean((pred6 - test[,1])^2)
 # Partial Least Squares
 model7 <- plsr(brozek~., data=train, validation="CV")
 which.min(model7$validation$adj)
 # resuilts in 17
 pred7 <- predict(model7, test[,2:18], ncomp=17)

 te7 <- mean((pred7 - test[,1])^2)

 TEALL = rbind(TEALL, cbind(te1, te2, te3, te4, te5, te6, te7) );

 }

dim(TEALL)

colnames(TEALL) <- c("mod1", "mod2", "mod3", "mod4", "mod5", "mod6", "mod7")

apply(TEALL, 2, mean)

apply(TEALL, 2, var)
```