A comparative Study of Ridge, LASSO and Principal components Regression

By

Franck A. OLILO

An Alternate Paper Plan Submitted in Partial Fulfillment of

The Requirements for the Degree of

Master of Science

In

Applied Statistics

Minnesota State University

Mankato, Minnesota

November 2022

A comparative study of Ridge, LASSO and Principal components Regression

Applied Statistics

Minnesota State University Mankato,

November 2022

Franck A. Olilo

ABSTRACT

One of the statistical techniques that is often employed and has applications in all aspects of daily life is linear regression. In regression, the goal is to correlate the variation in one or more response variables with proportional change in one or more explanatory factors to explain the variation in the response variables. They are deemed to be orthogonal if there is no linear relationship between these explanatory variables. Several of the explanatory variables will fluctuate in quite comparable ways if the variables are not orthogonal. This issue, known as multicollinearity, is one that frequently arises in regression analysis. When two or more explanatory variables are highly (but not perfectly) correlated with one another, it makes challenging to interpret the strength of each variable's effect because in the presence of multicollinearity the OLS estimators are not precisely estimated.

In the first part of this paper, we discuss the multicollinearity problem in linear regression model, present the technique to identify the problem, look for its causes and consequences. After that we explore ways to handle multicollinearity such as Ridge Regression, Lasso Regression and Principal Components regression and discuss the theory beyond them.

In addition, we attempted a case study and applied those methods, and we compare which among the OLS, RR, LAS, and PCR should be an alternative when fitting a model with multicollinearity. MSE, RMSE and R squared being the comparison factor, the results showed that RR, LAS and PCR have mean square error less than the OLS while RR and LASSO performs well than PCR

CONTENTS

LIST OF TABLES

LIST OF FIGURES

I- INTRODUCTION

One of the first methods for statistical analysis is the least squares (LS) approach, which is used to fit data under certain assumptions with one or more explanatory variables to select the optimal regression line that minimizes the sum of the squared errors. However, if these presumptions are broken, the LS approach cannot guarantee the desirable outcomes.
There are several assumptions that must be fulfilled for multiple regression analysis, one of which is the absence of multicollinearity. The presence of multicollinearity will cause a problem in the regression modeling by affecting the LS estimation. Therefore, it would be ideal to manage the problem before fitting our model. The most popular model for resolving the multicollinearity issue is both regularization method L1/L2 regularization (LAS and RR) and PCR,

Several studies have been done on estimating regression parameters when multicollinearity affects a dataset. (Eledum H.,2011) used simulation technique to compare between the three biased estimation methods RR, PCR, and Latent Root LR. (Yazid M. & Mowafaq M., 2009) and (Kianoush F., 2013) were used Monte Carlo simulation to estimate the regression coefficients by RR and PCR. (Moawad E., 2014) presented and compared the partial least squares (PLS) regression as an alternative procedure for handling multicollinearity problems with two RR and PCR. (Ali G. et al, 2014) studied the prediction of rangeland biomass using different methods including Multiple regression, Principal Component Analysis, Partial Least Square regression, and Ridge regression and compared them. (Piyush K. et al, 2013) compared RR and PCR estimators with the r-k class estimator, which is composed by combining the RR estimator and the PCR estimator into a single estimator [6]

This paper is organized as follows. We introduce multiple linear regression, section 2 discusses multicollinearity, section 3 pertains Ridge Regression, section 4 discusses the theory of LAS and PCR, in section 5 we attempt a case study the application of Ridge Regression in comparison of LASSO and PCR.

1.1- Multiple Linear Regression

Multiple linear regression model is a model that involves more than one regressor variable with a linear relationship between a dependent variable and independent variables. In others word the response variable y may be related k regressors $x_1, x_2, \dots, x_k$

$$y = \beta_0 + \beta_1 x_1 + \beta_1 x_1 + \cdots + \beta_k x_k + \varepsilon \qquad (1.1)$$

β_0 , β_1, …, β_k are the parameters or regression coefficients.
ε is the random error term
In matrix form, the model in (1.1) can be written as:

$$y = X\beta + \varepsilon \qquad (1.2)$$

where $\boldsymbol{y}$ is an $n * 1$ observation (dependent variable) matrix, X is $n * p + 1$ regressor variable, $\boldsymbol{\beta}$ is $p * 1$ parameter regression vector and $\boldsymbol{\varepsilon}$ is $n * 1$ random vector.
However before fitting a linear regression model, we must check the followings assumption

- The errors terms are independent and have constant variance
- The errors terms are normally distributed with mean vector equal to zero and variance covariance matrix ‘
- Matrix of $\boldsymbol{X}$ is a full column rank
- The errors are uncorrelated

After the assumptions have been checked, we must now estimate the regression estimates and the error.

1.2- Ordinary Least Square Estimation

The regression coefficient can be estimated using the Least Square methods
We can use equation (1.2) to find the vector of the least square estimators by minimizing

$$\begin{aligned} s(\beta) &= (y - X\beta)'(y - X\beta) \\ &= y'y - 2\beta'X'y + 2\beta'X'X\beta \end{aligned} \qquad (1.3)$$

$\beta'X'y$ is a 1×1 matrix with its transpose being $y'X\beta$.
The derivative of $s(\beta)$ on β must equal to 0

$$\frac{\partial S}{\partial \beta} = -2X'y + 2X'X\hat{\beta} = 0 \qquad (1.4)$$

The least square normal equation is found by simplifying equation (1.4), which give

$$X'X\hat{\beta} = X'y$$

And finally, we solve for the least square estimator of β

$$\hat{\beta} = (X'X)^{-1}X'y \qquad (1.5)$$

Both LS estimators and its covariance matrix depend on characteristics of the matrix X^TX.
LS estimators are sensitive to a number of errors which may result in wide confidence interval, make regressors statistically insignificant and have the regression coefficient have a wrong sign

II- MULTICOLLINEARITY

The multicollinearity terms come from Ranger Frisch (1993). It is a phenomenon in which two or more predictors in linear regression are correlated, if the degree of correlation is high enough, it can have an impact on the whole analysis. in this section we will talk about the different type of multicollinearity problem, their sources, consequences, and identification

2.1- Sources

There exist several factors of multicollinearity

- *Data Collection*

When collecting data, it may happen data sampling is done only over a small region of a design space and/or a range of value taken by the explanatory variable

- *Variable Selection*

In many cases, as the numbers of variables increases, each variable tends to measure the different nuances of some factor and each highly correlated variable only has little information content.

There may also exist a relationship among variable which may be due to their definition or any identity relation

- *Model formulation*

The model formulation may be too complicated such as adding polynomial terms to regression model, or the number of variables is larger than the observations

2.2- Consequences

The presence of multicollinearity has a serious effect on the least square estimate.

Let illustrate it using this model: $y = \beta_1 x_1 + \beta_2 x_2 + \varepsilon$

The least square equation:

$$(X'X)\hat{\beta} = X'y$$

can be written as:

$$\begin{bmatrix} 1 & r_{12} \\ r_{12} & 1 \end{bmatrix} \begin{bmatrix} \hat{\beta}_1 \\ \hat{\beta}_2 \end{bmatrix} = \begin{bmatrix} r_{1y} \\ r_{2y} \end{bmatrix}$$

The inverse of $(X^T X)$ is:

$$(X'X)^{-1} = \begin{bmatrix} \frac{1}{(1+r_{12})^2} & \frac{-r_{12}}{(1+r_{12})^2} \\ \frac{-r_{12}}{(1+r_{12})^2} & \frac{1}{(1+r_{12})^2} \end{bmatrix}$$

Which is used to estimate the regression coefficients by computing

$$\hat{\beta}_1 = \frac{r_{1y} - r_{12} r_{2y}}{(1 + r_{12})^2} \qquad \hat{\beta}_2 = \frac{r_{2y} - r_{12} r_{1y}}{(1 + r_{12})^2}$$

with the variance:

$$V(\hat{\beta}) = \sigma^2 (X'X)^{-1} = \frac{\sigma^2}{1 - r^2} \quad (2.7)$$

The effects of multicollinearity can be limited to:

- *Inflation of Variance*: using equation (2.7), as r tend to 1 or -1, variance tend to infinity, therefore if two variables are perfectly collinear, the variance become large

-*Estimation instability:* Since $X^T X$ is an ill-condition, the variance and OLS estimates are affected by the inverse matrix the instability of the of the inverse matrix $X^T X$ might make the estimation of the OLS numerically impossible to obtain and may be sensitive to minor change

-*Statistical Insignificance*: A high correlation in two variables affect their p-value so reduction of significance of a variable compared to the other. Therefore, the regression coefficients may be statistically insignificant because of large standard errors

2.3- Identification

With all consequences that multicollinearity can cause, it is a clever idea to identify its degree before fitting the model. The question we need to ask ourselves is how we diagnose the presence of multicollinearity. In this section we will go over the different techniques used to detect multicollinearity in a dataset.

- *Correlation Matrix*
The inspection of the diagonal elements in $X'X$ is one step of the measure of multicollinearity, because if two variables are nearly linear dependent the $|r|$ will be near 1. A rule of thumb requires the correlation coefficient of the explanatory variables should be less than 0.8. Any correlation coefficients greater than 0.8 indicate the presence of multicollinearity

-*Eigen system and Condition Number*
The eigenvalues of $X'X$, may be used to diagnose multicollinearity, the average eigenvalue 1 because the total sum is constant. The presence of multicollinearity is indicated when the eigenvalue is close to 0.
The condition number is calculated by the ratio of the largest to the smallest singular value from singular value decomposition

$$\text{Condition Number} = \frac{\text{MAX}\,\lambda_i}{\text{MIN}\,\lambda_i}$$

The condition number indicated how ill-condition the matrix $X'X$ is, the matrix is ill-condition when condition number is large.

-*Variance Inflation Factor*

The coefficient of determination R_j^2 will be helpful in computing VIF,

The diagonal element of C is $C_{jj} = \frac{1}{1-R_j^2}$ when X_j is regressed on $k-1$ variables. The values of C_{jj} depend on the orthogonality of X_j , two case may occurs

1st case: When X_j is nearly orthogonal to other variables, the coefficient of determination is small and make C_{jj} tends to 1

2nd case: When X_j is nearly linear independent to other variables, the coefficient of determination is close t 1 and make C_{jj} large.

In results (2.7), we can see that C_{jj} is a factor of the variance formula, which also affect the increase of the coefficients variance that is why C_{jj} is called the Variance Inflation Factor

$$\text{VIF} = C_{jj} = \frac{1}{1-R_j^2}$$

The presence of multicollinearity is indicated when the value of the VIF exceeds 5 or 10

III- RIDGE REGRESSION

3.1- Ridge estimator

In the previous section, we learned about the consequences of multicollinearity. One of the consequences being a considerable inflation of the variance of the least squares estimates. We learned that the variance inflation factor is defined as the diagonal elements of $(X'X)^{-1}$, which imply that a large variance inflation factor will result in large variance of the parameter estimates. This problem may come from the unbiased requirement of the estimators $\hat{\beta}$ []. Hoerl and Kinnard (1970,1975) while working on obtaining a biased estimator β introduced Ridge Regression.
The ridge estimator is found by solving the least square estimator equation with an added constant k, the new modified equation is

$$(\mathrm{X'X} + \mathrm{kI})\hat{\beta}_{\mathrm{R}} = \mathrm{X'y}$$

and the Ridge estimator being $\hat{\beta}_{\mathrm{R}} = (\mathrm{X'X} + \mathrm{kI})^{-1}\mathrm{X'y}$
Hoerl and Kinnard show that there exists a constant k for which MSE of $\hat{\beta}_{\mathrm{R}}$ is less than the variance of the least square estimator.

$$\mathrm{SS}_{\mathrm{Res}} = \left(\mathrm{y} - \mathrm{X}\widehat{\hat{\beta}_{\mathrm{R}}}\right)' \left(\mathrm{y} - \mathrm{X}\hat{\beta}_{\mathrm{R}}\right)$$

$$= \left(\mathrm{y} - \mathrm{X}\hat{\beta}\right)'\left(\mathrm{y} - \mathrm{X}\hat{\beta}\right) + \left(\hat{\beta}_{\mathrm{R}} - \hat{\beta}\right)'\mathrm{X'X}\left(\hat{\beta}_{\mathrm{R}} - \hat{\beta}\right)$$

The mean square error of the ridge estimator is:

$$\mathrm{MSE}\left(\hat{\beta}_{\mathrm{R}}\right) = \mathrm{Var}\left(\hat{\beta}_{\mathrm{R}}\right) + \left[\mathrm{bias}\left(\hat{\beta}_{\mathrm{R}}\right)\right]^2$$

$$= \mathrm{tr}\left[\mathrm{V}\left(\hat{\beta}_{\mathrm{R}}\right)\right] + \left[\mathrm{bias}\left(\hat{\beta}_{\mathrm{R}}\right)\right]^2 , \mathrm{V}\left(\hat{\beta}_{\mathrm{R}}\right) \text{ being the covariance}$$

$$= \sigma^2\mathrm{tr}[(\mathrm{X'X} + \mathrm{kI})^{-1}\mathrm{X'X}(\mathrm{X'X} + \mathrm{kI})^{-1}] + \mathrm{k}^2\beta'(\mathrm{X'X} + \mathrm{kI})^{-2}$$

$$\mathrm{MSE}\left(\hat{\beta}_{\mathrm{R}}\right) = \sigma^2 \sum_{\mathrm{j}=1}^{\mathrm{k}} \frac{\lambda_{\mathrm{j}}}{(\lambda_{\mathrm{j}}+\mathrm{k})^2} \mathrm{k}^2\beta'(\mathrm{X'X} + \mathrm{kI})^{-2}\beta$$

$\lambda_1, \lambda_2, \ldots, \lambda_p$ are the eigenvalues of $X'X$
While computation method exits to choose the best k value, Hoerl and Kinnard suggested a graphical method called Ridge Trace.

3.2- Ridge Trace and choice of k

The constant k is the biasing parameter and should be greater or equal to zero. If $k = 0$, the ridge estimator is the least square estimator. From equation (3.2) for an increase in k the RSS increase in oppositive of R^2 who decrease, it seems obvious that the objective is to select a value k of which stabilize the ridge estimator and produce a smaller MSE than the OLS.
Hoerl and Kinnard (1970,1975) suggested that Ridge Trace is an appropriate method to select k, The ridge trace plots the elements of the ridge estimator for different value k and from the plot we can observe at which value of k our ridge estimate stabilizes.

Using ridge regression can be advantageous, however it also has some disadvantages. Below is a limited list

Advantage: Avoids Overfitting
Does not required unbiased estimators
Improve least square estimate in case of multicollinearity

Disadvantage: Include all predictors
No feature selection
Trade variance for bias

IV- LASSO & Principal Components Regression

4.1- LASSO REGRESSION

LASSO was implemented by Toshigami 1996 as a solution of quadratic programming problem and has been used for model showing high levels of multicollinearity. LASSO is a regularization technique used over regression for a better prediction accuracy.
LASSO stands for Least Absolute Shrinkage and Selection Operator. The lasso model uses a shrinkage method which shrunk the data toward the mean, minimized the residual sum of square and mostly shrunk the least important feature coefficient to zero that reduce the number of explanatory variables in the final model which differs from Ridge that punishes all coefficient.
Lasso regression uses a regulation technique called L1 regularization compared to ridge regression that uses L2 regularization. L1 regularization add a penalty called the absolute value of magnitude to the loss function that is,

$$\min \sum_{i=1}^{n} (y_i - x_i\beta)^2 + \lambda \|\beta\|_1 \quad (4.1)$$

The second term in the equation (4.1) is the shrinkage penalty, as ridge regression we need to select a λ that produce the MSE
Similarly, to the ridge regression the lasso deals with overfitting by deriving a biased estimator but with possibly lower variance than the variance of the OLS estimator. It can lead to a lower expected prediction error in contrast to the result which we obtain when we use the OLS method. However, this does not hold in each situation. The examples are mentioned in (Hansen, 2016). Secondly, the lasso is better than OLS for the purpose of interpretation. With many independent variables, we often would like to identify a smaller subset of these variables that exhibit the strongest effects. The sparsity of the lasso is mainly counted as an advantage due to a simpler interpretation, but it is important to highlight that the lasso is not able to select more than n variables.

Advantages: Feature selection
Reduce overfitting
Easy interpretation

Disadvantages: Biased Coefficient
Prediction performance less than ridge

4.2- Principal Components Regression

PCA is a method that reduces the dimensionality of a dataset, by finding a new set of variables, smaller than the original set of variables. This efficient reduction of the number of variables is achieved by obtaining orthogonal linear combinations of the original variables—the so-called Principal Components (PCs). PCA was introduced by (Hotelling 1933) and is useful for the compression of data and to find patterns in high-dimensional data.
When dealing with multicollinearity, the first options to solve the problem are ridge and lasso regression, however we learned that one way to handle multicollinearity is dimension reduction. Principal components regression is a statistical method known for dimension reduction using the PCA process to estimate regression coefficients but instead of using the original feature it involves creating principal component uses them as predictors. The key idea is to explain the variability of the data using a smaller number of principal components.
PCR can be divided into four steps: the first step is to standardize the variables which subtract their means and divide their standard deviation
The second step is to run a principal components analysis on the table of the explanatory variables. We write it as

$$\mathrm{X'X} = \mathrm{PDP'} = \mathrm{Z'Z}$$

P: eigenvector matrix of $X'X$, D: is a diagonal matrix of eigenvalues of $X'X$, Z: data matrix of principal components
PC create new variable Z, so let say we start with variables X_1 , X_2 we will end with variables Z_1 , Z_2. Our new variables Z_1 and Z_2 will regress with Y which results in new biased estimates β
Therefore, the third step is to run an ordinary least squares regression (linear regression) on the selected components and obtain their estimate: the factors that are most correlated with the dependent variable will be selected. Finally, the parameters of the model are computed for the selected explanatory variables.

Advantages: Delete most correlated features
Avoids overfitting

Disadvantages: Information may get lost

V- APPLICATION / CASE STUDY

There are no direct methods to measure accurately the body fat of a person. The method used right now uses the body density through full water immersion and uses the Siri equation to calculate body fat. Our case study is to write a linear regression model with body fat as given different explanatory variables. However, we discovered that our dataset suffers from multicollinearity problem, which we will try to handle using Ridge Regression, LASSO Regression, Principal Components Analysis and compare which of those three options are a better fit for prediction.

5.1- **Data**

The data used in this paper was obtained from Kaggle, it originally has 15 variables and 25 observations. We are keeping a total of 7 variables for this case study after we found out some variables are not significance using backward p-value and AIC and/or does not match with the body fat prediction. The variables are listed below, and *Table 5.1* shows a partial view of the dataset.

Y : Body Fat
X_1: Weight
X_2: Chest
X_3: Abdomen
X_4: Hip
X_5: Thigh
X_6: Biceps

Body Fat	Weight	Chest	Abdomen	Hip	Thigh	Biceps
12.3	154.25	93.1	85.2	94.5	59	32
6.1	173.25	93.6	83	98.7	58.7	30.5
25.3	154	95.8	87.9	99.2	59.6	28.8
…	…	…	…	…	…	…
29.3	186.75	111.1	111.5	101.7	60.3	31.3
26	190.75	108.3	101.3	97.8	56	30.5
31.9	207.5	112.4	108.5	107.1	59.3	33.7

Table 5.1- Partial view of dataset

Before fitting our model, we are going to identify if pout dataset suffers from multicollinearity using method learned in section 2

5.2- **Measures of Multicollinearity**

The measures of multicollinearity were investigated using correlation matrix, tolerance, eigen values and variance inflations factor (VIF)

a- Correlation coefficient

The Pearson Correlation results is one of the methods used to check multicollinearity in the dataset and it gives the value at which each variable is collinear. While checking the Pearson Correlation results **(table5.2)** we are looking for value greater than 0.8, which indicate a high correlation

Variables	Y	X_1	X_2	X_3	X_4	X_5	X_6
Y	**1**						
X_1	**0.6199**	**1**					
X_2	**0.7009**	**0.8913**	**1**				
X_3	**0.8253**	**0.8742**	**0.9098**	**1**			
X_4	**-0.6384**	**0.92944**	**0.8207**	**0.8593**	**1**		
X_5	**0.556**	**0.8505**	**0.7055**	**0.7379**	**0.8837**	**1**	
X_6	**0.4821**	**0.7851**	**0.7072**	**0.6564**	**0.7173**	**0.7401**	**1**

Table5.2 – Pearson Correlation

In **Table 5.2** we observed that several variables are highly correlated to each other's such as (Body fat and Abdomen: 0.8253); (Chest and Abdomen:0.9098); (Chest and Thigh:0.8207) ;(Abdomen and hip:0.8593); (Hip and Thigh:0.8837) ; finally Weight and all others variables except Biceps

b- Eigen value & condition number

Using the matrix of the standardized explanatory variables, the eigen value can be used to diagnose multicollinearity. The average of a the eigen value is equal 1, any value greater than 1 or close to 0 indicates the presence of multicollinearity.

Eigen value	5.57160	0.65073	0.316316	0.27289	0.08840	0.053869	0.046181

Table 5.3-Eigen value

The square root of the ratio between the maximum of the eigenvalues is called the condition index and the largest among them is the condition number. A condition number between 10 and 30 indicates the presence of multicollinearity and when a value is larger than 30, we can mark it as strong

Condition number: $kappa = 120.6449$

Our condition number is way larger than 30 which confirms the presence of multicollinearity.

c- Tolerance & variance inflation factor

Another method to measure multicollinearity in a dataset is the Tolerance and Variance Inflation factor. When considering tolerance, we are looking for values that are far below 0 which indicate a problem. As for variance inflation factor, we are looking for values larger than 5 and 10

From table 4.3 we can see that two variable tolerance are below zero and show that 3 variables have VIF larger than 5 and 2 have VIF larger than 10 which again confirms multicollinearity.

Variables	Tolerance	VIF
X_1	0.07203	13.8813
X_2	0.11851	8.43771
X_3	0.13067	7.65247
X_4	0.09050	11.0487
X_5	0.18405	5.43320
X_6	0.34921	2.91613

Table 5.4-Tolerance and VIF

5.3- **Ridge regression and parameter estimation**

In section 5.2 we found out that our data suffer from a multicollinearity problem, To resolve the problem, applying the ridge regression concept which is the L2 regularization may be a good option to fit our model.
On section 3 we went over the theory of ridge regression and learned that finding an optimal value for lambda or k is the key aspect in ridge regression, there are various approach to use to choose or compute k, however in our case study we will be using the Ridge Trace and K-fold Cross Validation to determine the ideal parameter for our model.

a- Ridge Trace

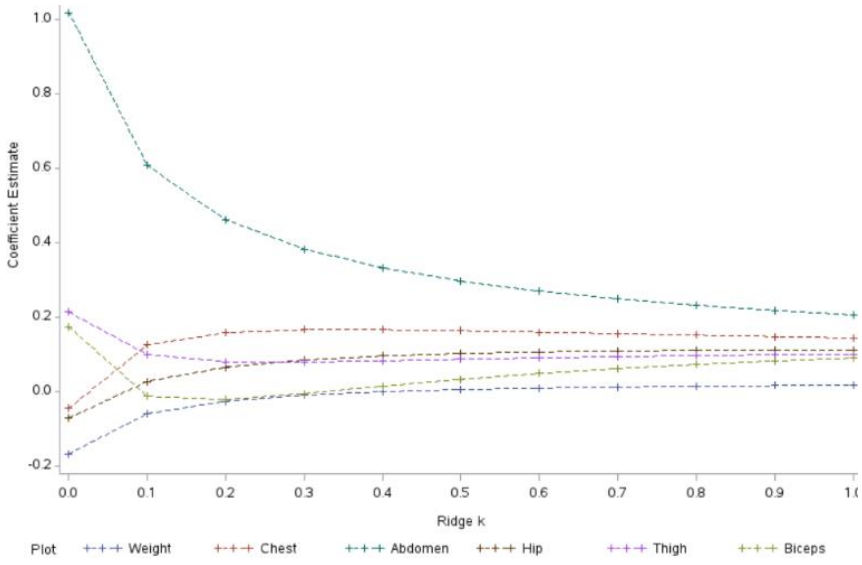

Figure 5.1

RStudio and SAS can be used to plot the ridge trace. Both SAS and R code can be found in page
Figure 5.1 display the trace for ridge regression for $0 \leq k \leq 1$.
To select the value of k, we are looking for the value we believe that stabilize all coefficient and lead coefficients to reasonable values

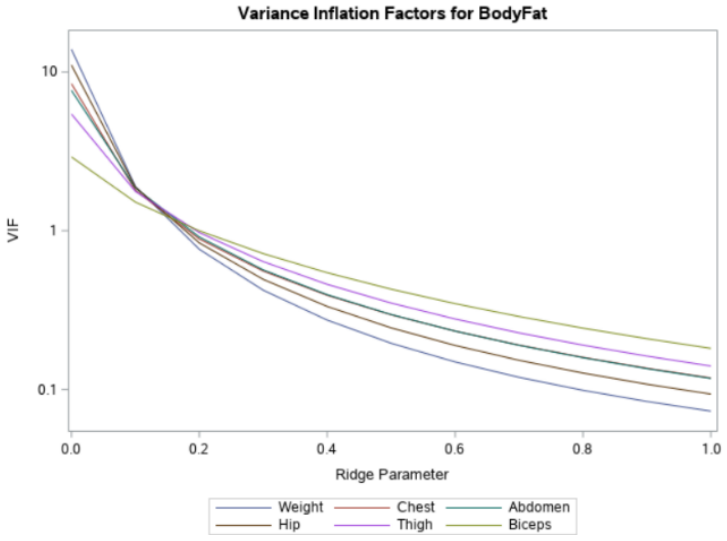

Figure 5.2

Figure 5.2 display the variation inflation factor for each variable over different value of the ridge parameter k. In this approach to find k is to look for the parameter that ensure all the VIF values are less than 5 or 10 and/or equal a desired value.

Let now have a look at figure **Table 5.5** which display partial ridge regression coefficient and vid score at value of k between 0 and 1

Type	Ridge k	RMSE	Intercept	Weight	Chest	Abdomen	Hip	Thigh	Biceps
PARMS		0.5243	-1.99E-15	-0.538	-0.042	1.247	-0.054	0.126	0.060
Ridge VIF	0.0			13.88	8.437	7.652	11.04	5.433	2.916
Ridge	0.0	0.5243	-1.99E-15	-0.538	-0.042	1.247	-0.054	0.126	0.060
Ridge VIF	0.02			7.459	5.207	5.065	6.336	3.962	2.439
Ridge	0.02	0.5288	-1.53E-15	-0.418	0.020	1.081	-0.032	0.101	0.033
Ridge VIF	0.04			4.737	3.655	3.676	4.227	3.075	2.115
Ridge	0.04	0.5374	-1.23E-15	-0.335	0.061	0.962	-0.013	0.085	0.016
Ridge VIF	0.08			2.454	2.189	2.257	2.352	2.075	1.675
Ridge	0.08	0.555	-8.82E-16	-0.226	0.108	0.801	0.012	0.065	-00019
…	…	…	…	…	…	…	…	…	…
Ridge VIF	0.50			0.1955	0.2955	0.296	0.244	0.350	0.428
Ridge	0.50	0.6435	-2.40E-16	0.020	0.160	0.364	0.080	0.051	0.011
…	…	…	…	…	…	…	…	…	…
Ridge VIF	1			0.073	0.118	0.117	0.093	0.140	0.181
Ridge	1	0.6805	-1.5E-16	0.005	0.141	0.252	0.087	0.060	0.031

Table 5.5- Partial ridge regression coefficient estimates and VIF score

Using all the information above we can guess which value is the ideal parameter. Someone can decode to pick any value after 0.2 since the VIF score of each variable is less than 5 another person can pick up 0.5 ask because most of the VIF are below 1. However, we are also trying to avoid additional problems such as overfitting therefore, we will use the k-fold cross validation technique to help us find the best parameter value

b- Cross Validation

K-fold cross validation is a technique that randomly splits the data into k approximately equal group (RStudio use default 10 times). The data is randomly split into two group, 9/10 go for training 1/10 for validation

Let consider a dataset consisting of 100 rows divided into ten folds each containg 10 rows. The first fold will represent the validation dataset and the rest will represent the training dataset.As next step we use this dataset to train our model and calculate its accuracy or loss, nand then repeat ths process for different validation set .An illustration can be found below

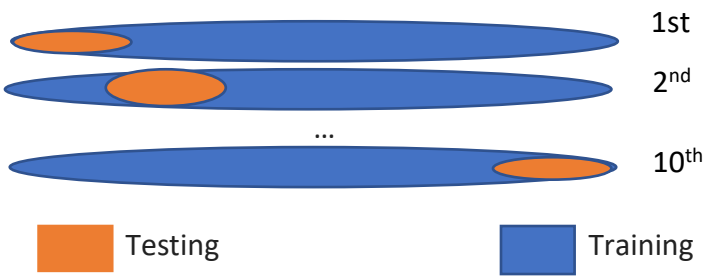

Figure 5.3- Illustration

The CV technique finds the optimal value of lambda or k by computing the Ridge regression coefficients based on the data for all the partitions except for the testing partition during each step. We then use these coefficients to forecast the y values of the data in the testing partition and calculate the residuals for each data element. The implementation of the cross validation using RStudio is under the build in function glmnet() and cv.glmnt().
The optimal lambda or k is determined via cross validation by minimizing the mean squared error

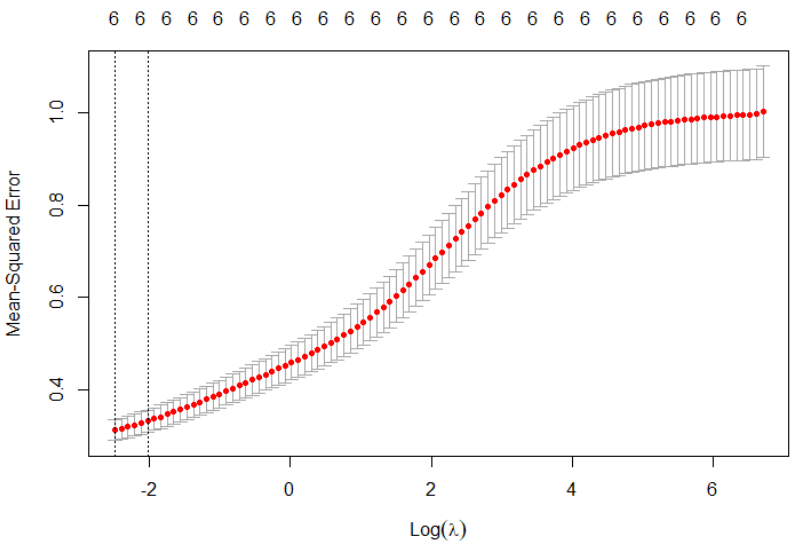

Figure 5.4

```
ridge3<-cv.glmnet(x,y,alpha=0)
ridge3$lambda.min
[1] 0.08237252
coef(ridge3)
7 x 1 sparse Matrix of class
"dgCMatrix"
                       s0
(Intercept) -2.431808e-16
Weight      -2.208937e-01
Chest        1.106214e-01
Abdomen      7.941868e-01
Hip          1.314701e-02
Thigh        6.421203e-02
Biceps      -8.189247e-04
```

Figure 5.5

The cross-validation technique estimates our optimal ridge parameter as 0.8327252, which would have been difficult to choose. In the next section, we will fit our ridge regression model and do a prediction

c- Prediction

The prediction made in this case study uses 90% of the data as training and 10% percent as testing. We are sing those percentage because our data set is not large enough

```
pred<-predict(ridge,newx=x,s=ridgeo$lambda.min)
eval_results(y, pred, data2)
       RMSE   Rsquare
1 0.524316 0.73173784
```

Figure 5.6

The results in figure 5.6 show that our ridge regression have a R^2 of 73.13% and 0.52 as Root of MSE

5.4- **LASSO**.

Previously, we introduced L2 regularization, we learned that we have two types of regularization. The difference between L1 and L2 is by how they manage the penalty. Ridge regression(L2) adds a square magnitude as penalty term to the loss function and LASSO (L1) adds an absolute value of the magnitude coefficient to the loss function. Another difference is that LASSO regression intends to shrink the coefficient of the less important/significant variable to zero. In other words, with a good Lasso parameter some coefficients will be exactly zero. Ridge regression and LASSO regression use the same R Studio build in function "glmnet", however the value of alpha differ, when alpha =0, no parameters are eliminated (Ridge Regression), when alpha=1, same parameters may be eliminated (LASSO) and when alpha is between 0 and 1, we have an elastic regression.
In addition to RStudio, SAS use the build in function "PROC GLMELECT" used to perform our LASSO regression
Similarly, to ridge regression, the function cv. glmnet will perform a k-cross validation which will help us identify the lambda or k that has the lowest MSE.

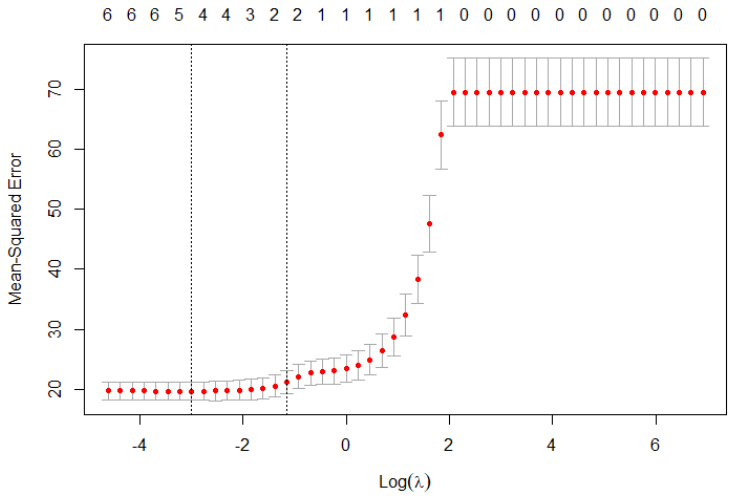

Figure 5.7

```
lasso$lambda.min
[1] 0.005408755
lasso<-
glmnet(x,y,alpha=1,lambda=0
.005408755)
coef(lasso)
7 x 1 sparse Matrix of
class "dgCMatrix"
                          s0
(Intercept) -1.830331e-16
Weight      -4.955335e-01
Chest        .
Abdomen      1.166109e+00
Hip          .
Thigh        8.420877e-02
Biceps       3.805854e-02
```

Figure 5.8

From figure 5.8 we observe that two coefficients are set t 0, the optimal k value or lambda value is 0.0054, which will be used to fit our final model

a- Prediction

After fitting the model with 0.0054 as value of k, our prediction give an R^2 of 73% and Root of mean square of 0.5185687

```
pred<-predict(lasso,newx=x,s=lasso$lambda.min)
eval_results(y, pred, data2)
```

```
      RMSE   Rsquare
1 0.52336    0.7305
```

Figure 5.9

5.5- **Principal Components Regression**

For this case study, we will RStudio library “pls”and function “prompt” to perform principal components regression while following the four steps in section 4. In addition to RStudio, SAS feature will also be used for verification of the PCR development.
We did first standardize our dataset before performing PCR

Body Fat	Weight	Chest	Abdomen	Hip	Thigh	Biceps
-0.814	-0.885	-0.931	-0.698	-0.801	-0.059	-0.076
-1.558	-0.182	-0.869	-0.913	-0.155	-0.12	-0.59
0.744	-0.894	-0.599	-0.434	-0.07	0.061	-1.173
…	…	…	…	…	…	…
1.223	0.316	1.279	1.876	0.305	0.202	-0.316
0.828	0.464	0.935	0.877	-0.294	-0.665	-0.59
1.535	1.084	1.438	1.582	1.134	0.001	0.506

Table 5.6- partial view of the standardize data

Principal components analysis is compute using “prcomp” function to illustrate the percent variation by different numbers of principal components

	PC1	PC2	PC3	PC4	PC5	PC6
Standard Deviation	2.244	0.651	0.558	0.311	0.29	0.214
Proportion of Variance	0.839	0.07	0.052	0.0161	0.0143	0.007
Cumulative Proportion	0.839	0.909	0.9618	0.977	0.992	1

Table 5.7 – Variance explained

We can see from table () that if all 6 principal components are used the percentage of variance is 100% however the model might be overfitted. Therefore, we will try to locate an optimal number of principal components to use.

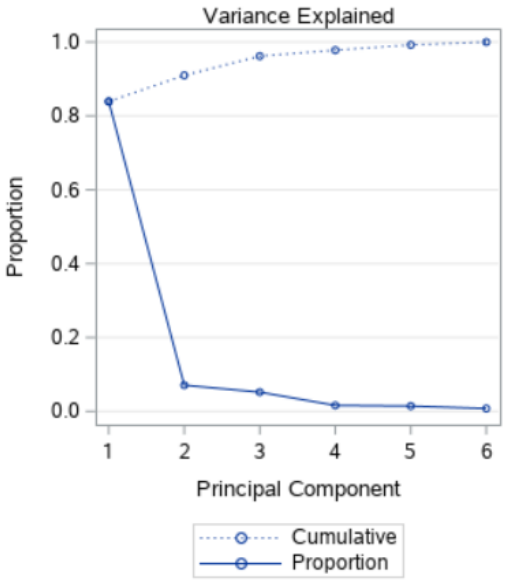

Figure 5.10

We can have a better look of the table using figure (5.10), we can observe that when adding first three principal components, there were a consequent gain in the percent of variance however adding PC4 and PC5 have less gain in PV

The choice of the optimal number of principal components can be hard to make, therefore we will compute the MSE while adding PC4, PC5 and PC6 to find which have the lower MSE and RMSE and we use the k-fold as additional resources for the optimal number of principal components. The k-fold cross validation technique has been explained in ridge

a- Prediction

The MSE and RMSE of the PCR model adding PC4, PC5 and PC6 are illustrated below (Figure 5.11, We observe that the MSE and RMSE have a large enough decrease from adding PC5 than adding PC6. Therefore, we are sure to pick up the model containing all PC except PC6

```
## 4 compemets
pcr_pred <- predict(pcr_model, test, ncomp = 4)
mean((pcr_pred - y_test)^2)
0.312036
sqrt(mean((pcr_pred - y_test)^2))
0.558602
##5 components
pcr_pred <- predict(pcr_model, test, ncomp = 5)
mean((pcr_pred - y_test)^2)
0.281248
sqrt(mean((pcr_pred - y_test)^2))
0.530329
## 6 componennts
pcr_pred <- predict(pcr_model, test, ncomp = 6)
mean((pcr_pred - y_test)^2)
0.283546
sqrt(mean((pcr_pred - y_test)^2))
0.532491
```

Figure 5.11

The prediction of principal components analysis output the result of all components analysis, which differ from ridge and Lasso which give the the prediction of the given optimal value.

In Figure 5.12 we can see that the percentage of variance explained does not increase from PC5 to PC6 , which confirms our expectation to stop at PC5.

```
pcrm<-
pcr(BodyFat~Weight+Chest+Abdomen+Hip+Thigh+Biceps,data=data1,sca
le=TRUE , validation="CV")
summary(pcrm)
VALIDATION: RMSEP
Cross-validated using 10 random segments.

TRAINING: % variance explained
         1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
X          83.90    90.98    96.18    97.80    99.23   100.00
BodyFat    48.69    58.44    58.65    70.14    73.17    73.17
```

Figure 5.12

Both graph in figure 5.13 illustrate the curve of the MSE and R^2 respectively
Numerically, the R^2 of the 5 principal components is less than or equal to 0.73 and the MSE is 0.530329

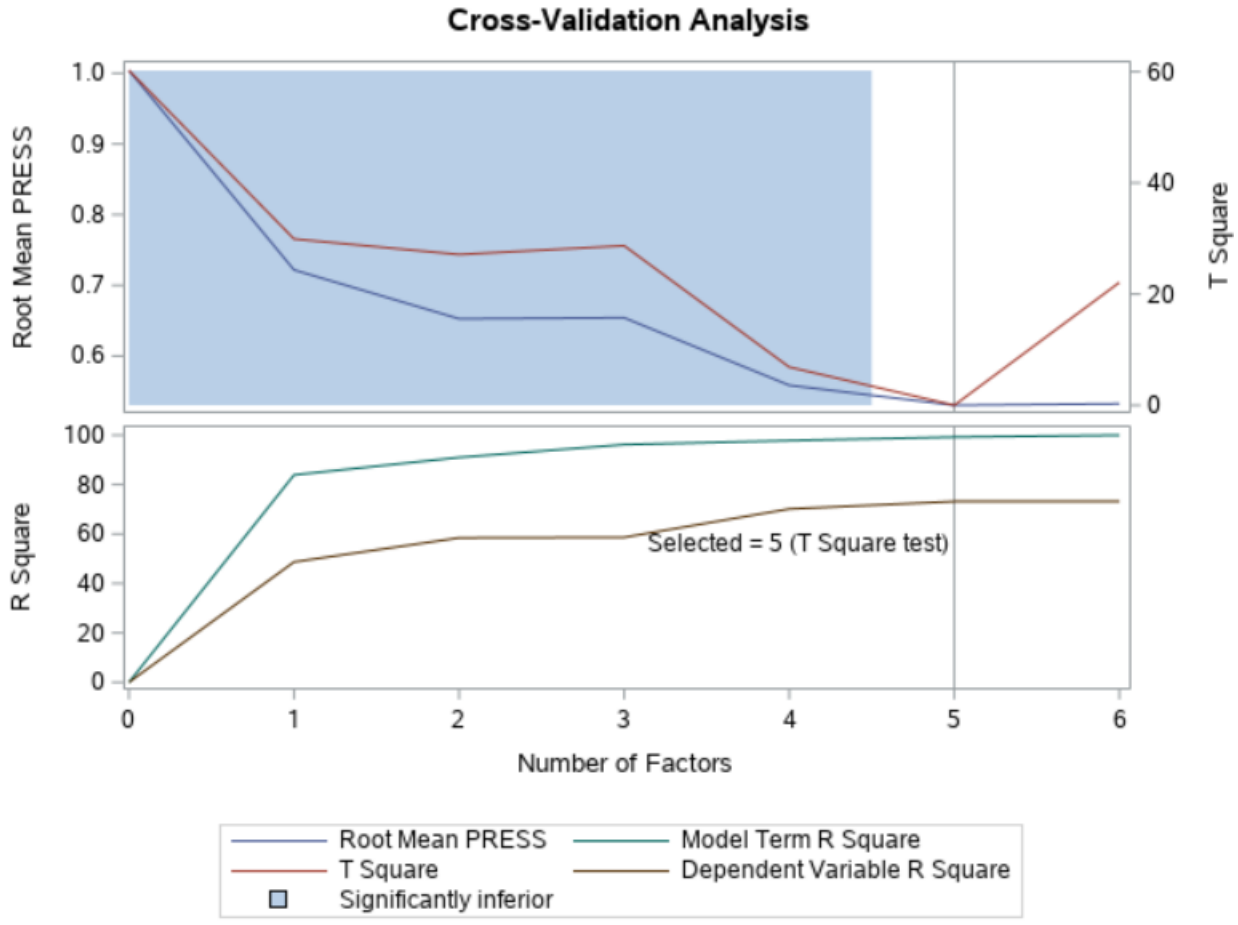

Figure 5.13

5.6- **Model Comparison**

So far in this case study, we identified the problem of multicollinearity, we decided to handle the problem with Ridge, LASSO and Principal components regression. The final step in our case study is to compare our three model is determine which of them work well in our data.
Our choice will be the model with the lowest MSE, RMSE and highest R^2 .

Model	RMSE	MSE	R-squared
OLS	4.310906	18.58391	0.6973
Ridge Regression	0.52432	0.27492	0.7317
Lasso Regression	0.52336	0.2739	0.7305
PCA	0.530329	0.28124	$0.7 < R^2 < 0.73$

Table 6.1

Table 6.1 list each of the model and their corresponding RMSE, MSE and R^2. The model with the lowest MSE and RMSE is LASSO regression, while ridge regression has the highest R^2
We observed that all three model have 0.52<RSME < 0.54, 0.27 < MSE< 0.29.

VI- CONCLUSION

If left untouched, multicollinearity can negatively impact the interpretation and accuracy of the model. Therefore, it is essential to diagnose and solve the problem of multicollinearity before fitting the model. Detecting multicollinearity can be done using several simple procedures such as correlation matrix, VIF, Tolerance and VIF etc., which may or may not confirm the presence of multicollinearity. If it is confirmed to be present, we should take the step to correct it through the implementation of techniques of Ridge, Lasso and Principal Component Regression. Those technique correct multicollinearity differently so they may have different impact and different prediction accuracy. Therefore, the analyst may use one technique depending on his goal.

Our goal was to select the technique with the lowest MSE and highest R^2 which is this case was

For future research, we would add as comparison method AIC

R-CODE

```
##TESTING MULTICOLLINEARITY
data1<-read.csv("bodyfat.csv")
attach(data1)
cor(data1)
eigen(cor(data1))$values
kappa(cor(data1),exact=TRUE)
mymodel<-lm(BodyFat~Weight+Chest+Abdomen+Hip+Thigh+Biceps)
ols_vif_tol(mymodel)

##rmse rsquare function
eval_results <- function(true, predicted, df) {
    SSE <- sum((predicted - true)^2)
    SST <- sum((true - mean(true))^2)
    R_square <- 1 - SSE / SST
    RMSE = sqrt(SSE/nrow(df))
data.frame(
    RMSE = RMSE,
    Rsquare = R_square
)

}

data1$BodyFat
linmod<-lm(BodyFat~Weight+Chest+Abdomen+Hip+Thigh+Biceps)
pred<-predict(linmod,newx=data1)
eval_results(y, pred, data1)


## RIDGE REGRESSION
data2=data.frame(scale(data1))
y<-data1$BodyFat
x<-
data.matrix(data1[,c('Weight','Chest','Abdomen','Hip','Thigh','B
iceps')])
ridgeo<-cv.glmnet(x,y,alpha=0,standardize=FALSE)
blambda<-ridge3$lambda.min
blambda
ridge3<-glmnet(x,y,alpha=0,lambda=0.08220827)
coef(ridge3)

## ridge regression PREDICTTION option 1
ridgen<-glmnet(x,y,alpha=0)
plot(ridgen,xvar="lambda",label=TRUE)
ridgeo<-cv.glmnet(x,y,alpha=0)
plot(ridgeo)
```

```
ridge3<-cv.glmnet(BodyFat~.,alpha=0,data=data1)
plot(ridge3)
coef(ridge3)
lm_seq=seq(0,1,0.01)
ridgem<-
lm.ridge(BodyFat~Weight+Chest+Abdomen+Hip+Thigh+Biceps,data=data
2,lambda=lm_seq)
select(ridgem)
## ridge regression PREDICTTION option 2
y<-data2$BodyFat
x<-
data.matrix(data2[,c('Weight','Chest','Abdomen','Hip','Thigh','B
iceps')])
ridgeo<-cv.glmnet(x,y,alpha=0,standardize=FALSE)
blambda<-ridge3$lambda.min
blambda
ridge<-glmnet(x,y,alpha=0)
pred<-predict(ridge,newx=data2,s=ridgeo$lambda.min)



##LASSO Regression rcode

library(mlbench)
library(tidyverse)
library(broom)
library(glmnet)
lasso1<-cv.glmnet(x,y,alpha=1,standardize=FALSE)
blambda<-lasso1$lambda.min
lasso<-glmnet(x,y,alpha=1)
pred<-predict(lasso,newx=data2,s=lasso1$lambda.min)

##PCA CODE

library(pls)
bodyfatx<-data1[,-1]
bodyfatpca<-prcomp(bodyfatx, center = TRUE,scale=TRUE)
summary(bodyfatpca)
plot(bodyfatpca,type="l")
train<-
data1[1:200,c("BodyFAt","Weight","Chest","Abdomen","Hip","Thigh"
,"Biceps")]
y_test<-data1[201:nrow(data1),c("BodyFat")]
test<-
data1[201:nrow(data1),c("Weight","Chest","Abdomen","Hip","Thigh"
,"Biceps")]

pcr_model <- pcr(BodyFat~., data = train,scale =TRUE, validation
= "CV")
```

```
pcr.fit <- train(BodyFat ~., data=train,preProc = c('center',
'scale'),method='pcr')
pcr.fit <- train(BodyFat ~., data=train,
                 preProc = c('center', 'scale'),
                 method='pcr')
```

SAS CODE

```
proc import datafile='/home/u50429604/bodyfat.csv' /*import xls
file*/
     dbms='csv' /* specify the type of EXCEL file to read*/
     out=final1 replace /*name of the data set*/;
     getnames=yes;
     run;

/*Standardize data*/
PROC STANDARD DATA=final1 MEAN=0 STD=1 OUT=final;
  VAR BodyFat Weight Chest Abdomen Hip Thigh Biceps;
RUN;

/*LASSO 1st option*/
proc glmselect data=final plots=all;
     model BodyFat = Weight Chest Abdomen Hip Thigh Biceps
           /selection=lar (stop=none choose=aic);
run;

/*LASSO 2nd option*/
proc glmselect data=final plots=all;
     model BodyFat = Weight Chest Abdomen Hip Thigh Biceps
           /selection=lasso;
run;

/*ridge regression*/
PROC STANDARD DATA=final1 MEAN=0 STD=1 OUT=final;
  VAR BodyFat Weight Chest Abdomen Hip Thigh Biceps;
RUN;
proc reg data=final;
model BodyFat = Weight Chest Abdomen Hip Thigh Biceps / vif tol
collin;
run;
/* 0<ridge<1 - Ridge Trace*/
proc reg data=final outvif plots(only)=ridge(unpack VIFaxis=log)
outest=pfinal ridge=0 to 1 by 0.01;
model BodyFat = Weight Chest Abdomen Hip Thigh Biceps ;
plot / ridgeplot nomodel nostat;
run;
proc print data=pfinal;
```

```
run;
/* Ridge : k=0.1*/
proc reg data=final outvif plots(only)=ridge(unpack VIFaxis=log)
outest=pfinal ridge=0.08237252;
model BodyFat = Weight Chest Abdomen Hip Thigh Biceps ;
plot / ridgeplot nomodel nostat;
run;
proc print data=pfinal;
run;
/* Ridge : Outseb*/
proc reg data=final outvif plots(only)=ridge(unpack VIFaxis=log)

/*PCR Principal components analysis */
proc princomp data=final1;
var Weight Chest Abdomen Hip Thigh Biceps;
run;
proc pls data=final1 method=pcr cv=one cvtest(stat=press) ;
model BodyFat = Weight Chest Abdomen Hip Thigh Biceps;
run;
proc pls data=final1 method=pcr nfac=6 ;
model BodyFat = Weight Chest Abdomen Hip Thigh Biceps;
run;
proc factor data=final1 rotate=varimax scree n=5 ;
var BodyFat Weight Chest Abdomen Hip Thigh Biceps;
run;
```

REFERENCES

[1] Yichao Wu (2019) Taylor & Francis Online
Can't Ridge Regression Perform Variables Selection?
https://www-tandfonline-com.ezproxy.mnsu.edu/doi/full/10.1080/00401706.2020.1791254

[2] S. Najarian, M. Arashi & B.M. Golam (2012) Taylor & Francis Online
A simulation Study on some Restricted Ridge Regression Estimators
https://www-tandfonline-com.ezproxy.mnsu.edu/doi/full/10.1080/03610918.2012.659953

[3] Deanna (2018) SAS Global Forum
Ridge Regression and Multicollinearity: An In Depth Review
https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2018/2825-2018.pdf

[4] Konstantinos T. Florida Atlantic University Digital Library
Ridge Regression
http://fau.digital.flvc.org/islandora/object/fau%3A11013/datastream/OBJ/view/RIDGE_REGRESSION.pdf

[5] Jackie Jen-Chy (1977) University of British Columbia
Multicollinearity, Autocorrelation , And Ridge Regression
https://open.library.ubc.ca/soa/cIRcle/collections/ubctheses/831/items/1.0094775

[6] Hussein Yousif (2016) International Journal of Research
A Comparison Study of Ridge Regression and Principle Components Regression with Application
https://www.researchgate.net/publication/301657827_A_Comparison_Study_of_Ridge_Regression_and_Principle_Component_Regression_with_Application

[7] Pasha and Muhammad Akbar (2004) Journal of Research
Application of Ridge Regression To Multicollinearity Data
https://www.bzu.edu.pk/jrscience/vol15no1/15.pdf

[8] Singh R.(2010) The IUP Journal of Computational Mathematics
A Survey of Ridge Regression for Improvement Over Ordinary Least Squares
https://papers.ssrn.com/sol3/papers.cfm?abstract_id=1750091

[9] Mitchell Claudius (1980) University Microfilms International
A Study of Ridge Regression and Some Examples of Its Application In Multivariate Analysis
https://auislandora.wrlc.org/islandora/object/thesesdissertations%3A894