A NEW 5-STEP PERTURBATION-BASED MULTICOLLINEARITY DIAGNOSTIC PACKAGE

by

Ryan Zamora, B.S.

A thesis submitted to the Graduate Council of
Texas State University in partial fulfillment
of the requirements for the degree of
Master of Science
with a Major in Mathematics
May 2019

Committee Members:

Shuying Sun, Chair

Alex White

Qiang Zhao

FAIR USE AND AUTHOR'S PERMISSION STATEMENT

Fair Use

Duplication Permission

DEDICATION

I would like to dedicate this thesis to my family, friends, and professors who have supported my personal and educational advancement.

ACKNOWLEDGEMENTS

I would like to acknowledge my committee members, Dr. Sun, Dr. White, and Dr. Zhao. Thank you for always having an open door for guidance and questions, and for helping me complete this thesis. I would like to especially acknowledge Dr. Sun. Through your guidance and strength, I am able to complete this project. I am forever grateful for the time and effort you put into helping me complete this thesis.

TABLE OF CONTENTS

Page

ACKNOWLEDGEMENTS v

LIST OF TABLES viii

LIST OF FIGURES ix

LIST OF ABREVIATIONS x

ABSTRACT xi

CHAPTER

I. INTRODUCTION 1

1.1 Background for multiple linear regression 1
1.2 What is multicollinearity and why is it a problem? 2
1.3 The structure of this thesis 5

II. REVIEW 6

2.1 Multicollinearity introduction 6
2.2 The *perturb* package 10
2.2.1 Example of the *perturb* package 11
2.3 The *mctest* package 15
2.3.1 Example of the *mctest* package 15
2.4 Strengths, weaknesses, and motivation for a new software package 19

III. A NEW PACKAGE 23

3.1 Introduction 23
3.2 R function overview and the 5-step chart 24
3.3 Body dimension dataset 27
3.4 Step 1. Perform observational analysis 29
3.5 Step 2. Perform perturbation analysis 38
3.6 Step 3. Calculate the overall or individual diagnostic measures and plot their distributions 39

3.7 Step 4. Conduct summary analysis for each regressor and calculate the rate of change 44
3.8 Step 5. Rank the overall diagnostics and/or identify coupling regressors 46

IV. DISCUSSION 51

4.1 Application to a larger dataset 51
4.2 Comparison to a variable selection method 57
4.3 Dealing with multicollinearity 59
4.4 Limitations 60

V. CONCLUSION 62

REFERENCES 63

LIST OF TABLES

Table **Page**

1. Multicollinearity diagnostic summary table 8
2. The summary of an MLR model for the Consumption dataset 11
3. Output of *colldiag* function....14
4. Output of *omcdiag* function....16
5. Output of *imcdiag* function....17
6. Output of *eigprop* function 17
7. Diagnostics checklist by packages....20
8. 5-steps of the *mcperturb* package 26
9. Summary statistics for a subset of the body dimension dataset....28
10. Summary table of implausible coefficients & standard errors....35
11. Output table of minimum, maximum, and max/min difference for the determinant...44
12. Summary table of VIF diagnostics with the variable shoulder perturbed 46
13. Overall diagnostic ranks by differences....47
14. Least squares best fit values for VIF....48
15. Rate of change values for VIF 48
16. Least squares best fit values for VIF with larger dataset....53
17. Rate of change values for VIF with larger dataset....55
18. Summary table of the forward selection method 58

LIST OF FIGURES

Figure **Page**

1. Perturbation analysis of Consumption dataset 13

2. VIF and eigenvalues plots from the *mc.plot* function 18

3. Correlation matrix of the regressors 29

4. The mean-centered density plots for the body dimension dataset 31

5. The mean-centered density function for shoulder diameter and chest girth 33

6. MLR adjusted R^2 values 37

7. Boxplots of the determinant of $\boldsymbol{X'X}$ with respect to the noise variable 40

8. Boxplots of variables' vifList with noise added to the shoulder regressor 42

9. Boxplots of the VIF's for all the regressors at multiple noise levels 43

LIST OF ABBREVIATIONS

Abbreviation	Description
MLR	Multiple linear regression
OLS	Ordinary least squares
VIF	Variable inflation factors
TOL	Tolerance
SLR	Simple linear regression
AIC	Akaike information criteria
PCR	Principal component regression

ABSTRACT

The ordinary least squares method, for estimating unknown parameters of a multiple linear regression (MLR) model, produces an idealistic solution if the column vectors (regressors) of the design matrix **X** are linearly independent. However, in a typical MLR setting, true linear independence of the regressors is often an unrealistic situation. Multicollinearity arises as two or more predictor variables departure from linear independence, thus, providing the model with redundant information and causing problems in the MLR parameter estimation and inaccurate statistical inference. The degree of multicollinearity directly reflects the amount of redundancy or interdependence among regressors and the inaccuracy of the MLR inference. Several statistical and analytical detection methods exist and are commonly used to diagnose multicollinearity. These diagnostic methods often produce a measure that reflects the degree of multicollinearity present in the overall model or among the individual regressors. However, these diagnostic methods generally fail to breakdown complex multicollinearity relationships among the regressors. There is also lack of a methodology that combines perturbation analysis with the available diagnostic measures. In addition, several observational strategies are often overlooked and underutilized for diagnosing multicollinearity. Therefore, we develop a new R package, *mcperturb*, that encompasses several multicollinearity observational strategies and employs a new 5-step perturbation-based method. This package can identify the regressors that may be the main source of the multicollinearity problem. The outputs from the *mcperturb* package provide a comprehendible opportunity to observe the

relatedness between two or more variables on a deeper level than the currently available multicollinearity diagnostic packages.

I. INTRODUCTION

1.1 Background for multiple linear regression

Across many disciplines, multiple linear regression (MLR) analysis is one of the most commonly practiced modeling techniques. It is often used for the investigation of linear relationships among variables and making statistical inference. An MLR model can be written in matrix form as:

$$\boldsymbol{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

where **y** is an $n \times 1$ vector of responses, **X** is an $n \times p$ matrix ($p = k + 1$), k is the number of regressors, $\boldsymbol{\beta}$ is an $p \times 1$ vector of coefficients or parameters, and $\boldsymbol{\varepsilon}$ is an $n \times 1$ vector of random errors. $\boldsymbol{\beta}$ is a vector of unknown parameters and is estimated using the ordinary least-squares (OLS) method. The OLS estimator of $\boldsymbol{\beta}$ can be written as:

$$\widehat{\boldsymbol{\beta}} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\boldsymbol{X}'\boldsymbol{y},$$

where $\boldsymbol{X}'\boldsymbol{X}$ is a square $p \times p$ matrix. Therefore, the estimated MLR model is written as:

$$\widehat{\boldsymbol{y}} = \mathbf{X}\widehat{\boldsymbol{\beta}} + \hat{\boldsymbol{\varepsilon}}.$$

Like any good modeling technique, key assumptions about the errors must be met to achieve reliable results. The key assumptions for the ordinary least squares (OLS) estimator used for estimating the unknown parameters of an MLR model are (Montgomery et al., 2012):

a. The errors are normally distributed.

b. The error terms are uncorrelated.

c. The expectation for the error terms is zero E($\boldsymbol{\varepsilon}$) = **0**.

d. The variance for the error terms is constant Var($\boldsymbol{\varepsilon}$) = $\sigma^2 \boldsymbol{I}_n$, where $\boldsymbol{I}_n$ is the identity matrix.

The Gauss-Markov theorem states that the OLS estimators are best linear unbiased estimators if the last three assumptions are satisfied (Montgomery et al., 2012; Ott et al., 2015). There are often no explicit assumptions for the predictor variables of the **X**-matrix. However, to derive the least-squares estimator for $\boldsymbol{\beta}$, one implicit assumption is that the predictor variables are linearly independent of each other. If linear dependence exists among the predictor variables, the **X**-matrix will not be of full rank, therefore singular. If the **X**-matrix is not full rank, then the $\boldsymbol{X}'\boldsymbol{X}$-square matrix will not be of full rank. The $\boldsymbol{X}'\boldsymbol{X}$-square matrix cannot be inverted if it is not of full rank, thus the OLS estimator for $\boldsymbol{\beta}$ is not solvable (Curto & Pinto, 2007).

When predictor variables are highly correlated or are approaching linear dependence, the least-squares estimator for $\boldsymbol{\beta}$ will become unstable because the **X**-matrix is ill-conditioned. This problem is often referred to as multicollinearity. In the rest of this thesis project, we will discuss multicollinearity in greater detail and introduce a new perturbation strategy that employs existing diagnostic measures to identify the regressor variables that may associate with a multicollinearity issue.

1.2 What is multicollinearity and why is it a problem?

Before defining multicollinearity and discussing why multicollinearity is a problem in greater detail, it's important to introduce some key terms that are often used throughout the

multicollinearity discussion. *Relatedness* is a term used to describe a relationship between variables. Two variables can be linearly, exponentially, quadratically, or otherwise related or associated with each other. Generally, the more two variables are related to each other, the more they are thought to provide similar information to the model. The term *orthogonal* is used for describing no linear relationships between regressors or when the regressors are uncorrelated (Montgomery et al., 2012; Mertler & Reinhart, 2016). It has been stated that complete independence among variables means that the variables are orthogonal to each other (Mertler & Reinhart, 2016). The column vectors of a matrix are considered *linearly dependent* if and only if there is a set of constants $a_1, a_2, \dots, a_p$, not all zero such that

$$\sum_{1}^{p} a_i \boldsymbol{x}_i = \mathbf{0}.$$

The column vectors are considered *linearly independent* if

$$\sum_{i=1}^{p} a_i \boldsymbol{x}_i = \mathbf{0}$$

can only be satisfied by the trivial solution, $a_i = 0$ for $i = 1, 2, .. p$. The *condition number* of a square matrix can be found by calculating the ratio between the largest eigenvalue λ_{max} and smallest eigenvalue λ_{min} (Montgomery et al., 2012; Belsley et al., 1980).

$$\kappa = \frac{\lambda_{max}}{\lambda_{min}}$$

A *well-conditioned* matrix has column vectors that have low dependencies on each other; resulting in a small condition number. An *ill-conditioned* matrix occurs when columns of a matrix have large dependencies of each other. Thus, the condition number of the matrix is large (Belsley et al., 1980). Condition indices of a square matrix are defined as

$$\kappa_j = \frac{\lambda_{max}}{\lambda_j}, j = 1,2, \dots, p.$$

The largest condition index of a square matrix is the condition number (Montgomery et al., 2012; Belsley et al., 1980). With a large number of column vectors, a common occurrence is the *redundancy* of variables. The term redundancy of variables is used when two or more variables are highly related to each other, thus providing the same or similar information to the model. Finally, a square matrix is considered a *singular matrix* if the rank of the matrix is less than the number of parameters, thus, the determinant is 0. A square matrix that has linearly dependent columns is considered a singular matrix, which cannot be inverted.

In literature, collinearity and multicollinearity are often used interchangeably as shown below. The term *collinearity* has been defined as the degree of linear dependence between two regressor variables, and perfect collinearity occurs when one of the predictors is an exact linear combination of one or more other regressor variables (Kleinbaum et al., 2007). Collinearity has also been referred to as the near-linear relation among the predictors (Hocking, 2013). The term *multicollinearity* can be defined as the existence of near-linear dependence between the regressors (Montgomery et al., 2012). Multicollinearity has been described as a mathematical problem that arises when high intercorrelations exist among predictor variables (Mertler & Reinhart, 2016). To avoid confusion throughout the rest of this paper, multicollinearity will be used henceforth.

If perfect multicollinearity exists, the **X**-matrix will not be full rank. Therefore, the least-squares estimator will not be solvable because the $\boldsymbol{X}'\boldsymbol{X}$-square matrix is singular and

consequently is not invertible. The **X**-matrix can approach singularity if the predictor variables are highly related or interdependent on each other. As the $\boldsymbol{X'X}$-square matrix approaches singularity, multicollinearity problems arise in an MLR model. These problems can be seen as large in magnitude standard errors for the regressors, as well as implausible and unstable coefficient estimates. Therefore, an MLR model that suffers from multicollinearity will often have inaccurate and unstable coefficients and should be perceived as unreliable for statistical inference (Montgomery et al., 2012; Ott et al., 2015).

1.3 The structure of this thesis

The structure of this thesis is described below. In chapter 2, we'll be reviewing two available R-packages used for diagnosing multicollinearity, *perturb* and *mctest* (Hendrickx, 2012; Imdadullah et al, 2016). We'll discuss the strengths and weaknesses of these two packages, the output files from their main functions with interpretation, and how each package inspires the idea of this thesis. In chapter 3 we will introduce a new multicollinearity diagnostics package called *mcperturb*. The *mcperturb* package is based on a 5-step perturbation strategy (algorithm). Detailed examples will be provided to illustrate the capabilities of this new package. The observational strategy/analysis from the *mcperturb* package will be explored first. The overall diagnostics perturbation strategy will follow with example output files from the package. Finally, the individual diagnostic perturbation strategy will be explored using an example and the output files of the *mcperturb* package.

II. REVIEW

2.1 Multicollinearity introduction

Two aspects are explored when diagnosing a multicollinearity problem. The first aspect is about identifying the source and measuring the degree. Identifying the source of a multicollinearity problem often encompasses the following four primary origins (Montgomery et al., 2012).

a. The sampling technique or data collection method
b. The model or population constraints
c. The model and variable specification
d. The "curse of dimensionality" or an over fitted model

Essential multicollinearity is defined as the multicollinearity that is inherent to the model and cannot be easily fixed, e.g., a and b, as shown above. Montgomery et. al. suggest that identifying essential multicollinearities is the primary concern because they cannot be easily fixed and require advanced detection methods (Montgomery et al., 2012). Nonessential multicollinearity can be defined as the multicollinearity that is introduced to the model by the researcher, e.g., c and d, as listed above. An example of nonessential multicollinearity is the multicollinearity that arises from the inclusion of higher ordered terms in the model. Most authors agree that the nonessential multicollinearity causes can be easily fixed and often prevented (Montgomery et al., 2012; Iacobucci et al., 2016).

The second aspect of diagnosing the multicollinearity problem is about measuring the

degree of multicollinearity that exists in the overall model and among the individual regressors. Several multicollinearity diagnostics have been developed to measure the degree of multicollinearity. In Table 1, we summarize several diagnostic measures by their name, type, focus, and root. A more detailed summary of multicollinearity diagnostics can be found in Imdadullah et al. (2016).

Table 1 column 2 is the "type" of multicollinearity diagnostic. It categorizes based on whether the diagnostic is an observational strategy, a statistical measure, or a numerical analysis technique. Unlike a statistical measure or numerical analysis technique, an observational strategy/analysis does not measure the degree of multicollinearity. Instead, an observation analysis may display key information that are related to the source of the multicollinearity problem. Table 1 column 3 is the "focus" of each multicollinearity diagnostic. It classifies whether a diagnostic focuses on providing information about the individual regressor's or the overall model's multicollinearity problem, or just the source of the multicollinearity problem. The statistical measures and numerical techniques quantify the degree of multicollinearity. The "cutoff" values for the statistical and numerical diagnostic measures are suggested by Imdadullah et al. (2016). Table 1 column 4 is the "root" of the multicollinearity diagnostic. It summarizes according to the root analysis performed. The root of a diagnostic is based on the variance (σ^2) of the model or individual regressors, coefficient of determination (R^2) for the model of individual regressors, eigenvalues (μ, λ) of the **X**-matrix and $\boldsymbol{X'X}$-matrix respectively, and the determinant $|Det|$ of the $\boldsymbol{X'X}$-matrix.

Table 1. Multicollinearity diagnostic summary table

Diagnostic Name	**Type**	**Focus**	**Root**
Plotting the regressors density functions	Observational	Source	σ^2
Identifying implausible coefficients & standard errors	Observational	Source	σ^2
Perturbation analysis	Observational	Source	σ^2
Identifying insignificant t-stats when overall F-test is significant	Observational	Source	σ^2
Change in the overall models R^2	Observational	Source	R^2
Pair-wise correlation matrix	Statistical Measure	Individual	R^2
VIFs	Statistical Measure	Individual	$\sigma^2/R^2/\lambda$, μ
TOL limit	Statistical Measure	Individual	$\sigma^2/R^2/\lambda$, μ
Fi method	Statistical Measure	Individual	R^2
Leamer's method	Statistical Measure	Individual	σ^2
Corrected VIF	Statistical Measure	Individual	$\sigma^2/R^2/\lambda$, μ
Kleins rule	Statistical Measure	Individual	R^2
Variance decomposition proportions	Statistical Measure	Individual	$\sigma^2/R^2/\lambda$, μ
Determinant of $\boldsymbol{X'X}$ matrix	Numerical Technique	Overall	|Det|
Farrar's $\boldsymbol{\chi^2}$	Statistical Measure	Overall	σ^2
Red's indicator	Numerical Technique	Overall	λ
Sum of the inverse λ	Numerical Technique	Overall	λ
Theil's indicator	Statistical Measure	Overall	R^2
Condition number & condition indices	Numerical Technique	Overall	λ, μ
Eigenvalues	Numerical Technique	Overall	λ, μ

A majority if not all of the diagnostic measures listed in Table 1 have direct or indirect relationships in their equations. For example, the VIF is equal to the inverse of the Tolerance Limit. That is

$$VIF = \frac{1}{1 - R_j^2} = (TOL)^{-1},$$

(Montgomery et al., 2012; Imdadullah et al., 2016), where R_j^2 is the coefficient of determination obtained when x_j is regressed on the remaining $p - 1$ regressors. Also,

$$VIF = \sum_{i=1}^{p} \frac{t_{ji}^2}{\mu_i^2} = \sum_{i=1}^{p} \frac{t_{ji}^2}{\lambda_i^2},$$

where λ_i are the eigenvalues of the $\boldsymbol{X'X}$ matrix, μ_i are the singular values of the **X**-matrix (note, $\boldsymbol{X} = \boldsymbol{UDT'}$ is called the singular-value decomposition of **X**), and t_j is the associated eigenvector of the of the $\boldsymbol{X'X}$ matrix (Montgomery et al., 2012). Finally, the variance decomposition proportion matrix can be defined as

$$\pi_{ij} = \frac{\frac{t_{ji}^2}{\mu_i^2}}{VIF_j}, \qquad j = 1,2,\dots,p \text{ (Montgomery et al., 2012).}$$

Plotting the density functions of the regressors is not a typical multicollinearity observational strategy in the current literature. This is one of the new contributions of this thesis project and the strategy will be discussed in detail in Chapter 3.

Recent R-packages, *perturb* and *mctest*, have been created for the purpose of diagnosing multicollinearity (Hendrickx, 2012; Imdadullah et al., 2016). These existing R-packages have different strategies for diagnosing multicollinearity. Their strengths and weaknesses will be discussed in the following sections as well as how they motivate the creation of the *mcperturb* package. We will start the review with the *perturb* package and its two main functions. Then we switch our focus to the *mctest* package and its diagnostic measures.

2.2 The *perturb* package

For evaluating multicollinearity, Belsley proposed the original perturbation strategy in 1980 (Belsley, 1980). He introduced a small amount of random normally distributed noise into the X-matrix and performed an MLR model. He performed this procedure for multiple iterations. Then he observed and analyzed the variability of the coefficients. If multicollinearity exists in an MLR model, adding a small amount of noise to the X-matrix can expose unstable coefficients (Belsley, 1980; Hendrickx, 2012). Thus, multicollinearity diagnostics uses perturbation as an observational strategy.

The *perturb* package provides two main functions, *perturb* and *colldiag*. The methodology behind the *perturb* function is to introduce random noise (interference) into the X-matrix for multiple iterations and calculate the MLR coefficients for each predictor variable. The *colldiag* function calculates the condition number, condition indices, and the variance decomposition proportions for the regressors in the **X**-matrix (Hendrickx, 2012).

The *perturb* package uses the "consumption" dataset, a dataframe with 5 variables and 28 observations. The 5 variables are year (1947-1974), cons (total consumption dollars), rate (interest rate), dpi (deposit income), and d_dpi (change in deposit income) (Belsley, 1991). The cons variable is the response variable and the predictor variables are year, rate, dpi, and d_dpi. The *perturb* package provides source code that generates the output files for the *perturb* and *colldiag* functions in the following example.

2.2.1 Example of the *perturb* package

We start off the example by first providing the summary table for the MLR model in Table 2, which is based on the MLR model for the raw data without perturbation. In Table 2, the regressors with large standard errors can be identified.

```
R code used for generating Table 2
library("perturb")
data(consumption)
attach(consumption)
ct1 <- with(consumption, c(NA, cons[-length(cons)]))
C_regmod <- lm(cons~ct1+dpi+rate+d_dpi,data = consumption)
summary(C_regmod)
```

Table 2. The summary of an MLR model for the Consumption dataset

```
Call:
lm(formula = cons ~ ct1 + dpi + rate + d_dpi, data = consumption)

Residuals:
    Min      1Q  Median      3Q     Max
-6.5567 -2.5185 -0.8726  2.2804  5.8832

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   6.7242     3.8271   1.757  0.09283 .
ct1           0.2454     0.2375   1.033  0.31271
dpi           0.6984     0.2077   3.363  0.00281 **
rate         -2.2097     1.8384  -1.202  0.24217
d_dpi         0.1608     0.1834   0.877  0.39016
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.557 on 22 degrees of freedom
  (1 observation deleted due to missingness)
Multiple R-squared:  0.9991,   Adjusted R-squared:  0.9989
F-statistic:  5964 on 4 and 22 DF,  p-value: < 2.2e-16
```

From the summary table in Table 2 we can identify large standard errors for the intercept term and the rate regressor. This may be due to the regressors that provide the same information to the model. This redundancy among the variables is a sign that multicollinearity may exist among the regressors. Another sign that this model may have a multicollinearity issue is that the overall model's p-value for the F statistic is highly significant, while only the dpi regressor has a significant p-value. The intercept term and the rate term are not significant coefficients. This might be because they have large standard errors relative to their estimates. Thus, multicollinearity may be the reason for these regressors to be insignificant to the model.

Figure 1 displays the boxplots of the coefficients calculated using the *perturb* function. The *perturb* function allows random normal or uniform noise to be added to selected predictor variables. The *perturb* function takes in a model, a list of noise variables, and list of noise amounts. It then runs the model with the perturbation variables for n iterations and outputs a summary table of the calculated coefficients. We performed n = 100 iterations with noise added to the "dpi", "rate", and "d_dpi" regressors. In Figure 1, the robustness of the coefficients should be observed. From the boxplots in Figure 1, we can identify large variations in the coefficients for the intercept term and the rate regressor. These variations show that small perturbations in the dataset lead to large variation of their coefficients. This is a sign that multicollinearity may exist among the regressors.

R code used to generate Figure 1

```
perturb1 <- perturb(C_regmod, pvars = c("dpi", "rate", "d_dpi"), prange = c(1, 1, 1))
boxplot(perturb1$coeff.table,  ylab = "Coefficients", xlab = "Regressors")
```

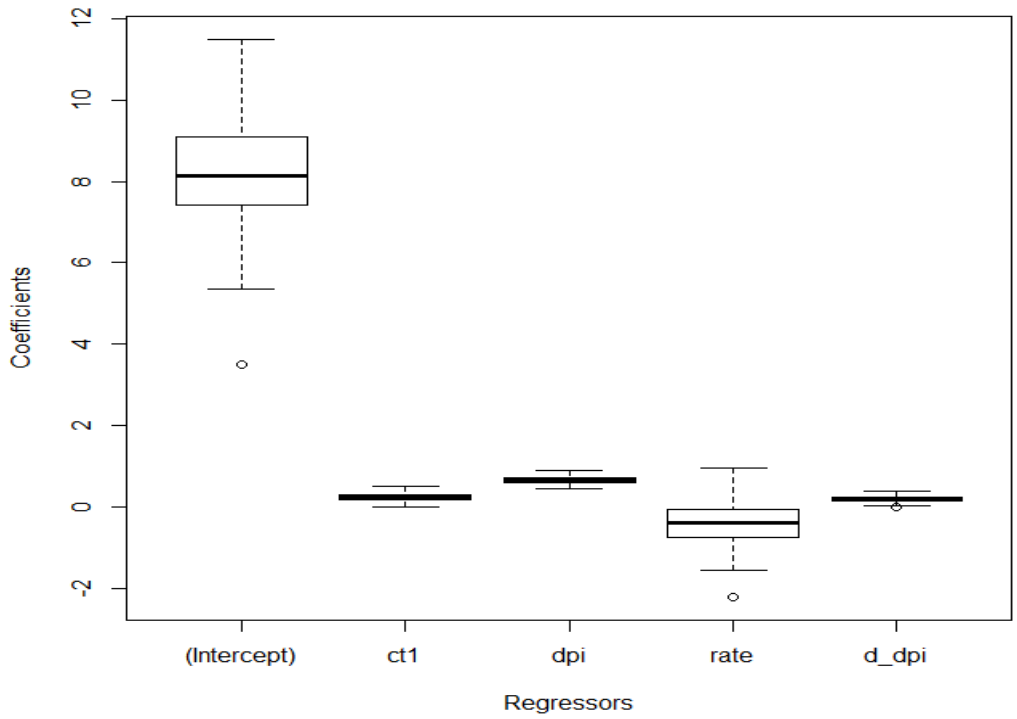

Figure 1. Perturbation analysis of Consumption dataset.

To conclude the example, we perform the *colldiag* function. The *colldiag* function takes in a model object or a dataframe and calculates the condition indexes and variance decomposition proportions. The variance decomposition proportions are the proportions of the variance for each coefficient contributed by its eigenvalue. Table 3 shows the output from the *colldiag* function. In Table 3, large condition indexes and variance decomposition proportions should be identified. The procedure for performing the *colldiag* function is as follows. First, identify the condition indices greater than 30, then identify the variance decomposition proportions greater than 0.5 (Montgomery et al., 2012; Hendrickx, 2012). The variables that have variance decomposition proportions greater than 0.5 may be causing a multicollinearity problem.

```
R code used to generate Table 3
cd = colldiag(C_regmod)
print(cd, fuzz = 0.3)
```

Table 3. Output of *colldiag* function

```
Condition
Index   Variance Decomposition Proportions
           intercept ct1   dpi     rate  d_dpi
1    1.000     -      -     -       -      -
2    4.143     -      -     -       -      -
3    7.799   0.310    -     -       -      -
4   39.406     -      -     -     0.984    -
5  375.614   0.421  0.995  0.995    -     0.814
```

From Table 3, we find that the 4th and 5th condition indices are greater than 30. We also find that for the 4th and 5th condition indices, the rate regressor has a variance decomposition proportion much greater than 0.5 and that the ct1, dpi, and d_dpi regressors all have a variance decomposition proportion greater than 0.5, respectively. Thus, these variables may be identified as a potential source for multicollinearity problem.

We can use the observational strategies offered by the *perturb* package and identify regressors that may be causing a multicollinearity problem. The two conclusions we can draw from the *perturb* function and the *colldiag* function are that multicollinearity might exist in the overall model and that the rate regressor may be associated with a multicollinearity problem.

2.3 The *mctest* package

The *mctest* package is created specifically for calculating multicollinearity detection measures and outputting descriptive summary tables. The package provides the *mctest*, *imcdiag*, *omcdiag*, *eigprop*, and *mc.plot* functions. The *mctest* function is the main function that output summary tables from the *imcdiag* and/or *omcdiag* functions. The *imcdiag* function calculates some of the diagnostics classified as individual diagnostic measures shown in Table 1. The *omcdiag* function calculates some of the overall model diagnostic measures listed in Table 1. Each function returns a summary table with the calculated metrics and outputs a suggestion regarding either the detection of multicollinearity in overall model or the regressors deemed insignificant to the model. The output tables are provided in Table 4 and 5. The *eigprop* function provides similar analysis and outputs as the *colldiag* function in the *perturb* package. The *mc.plot* function displays a VIF plots and an eigenvalue plot. For more information about the different functions offered in the *mctest* diagnostic package, please refer to the user guide (Imdadullah et al., 2015).

2.3.1 Example of the *mctest* package

The dataset and source code used for the following example can be found in the example section of the *mctest* function. The name of the dataset is "Hald" cement which consist of 13 observations and 5 predictor variables. The dependent variable is the Y (heat) variable. The predictor variables in the dataset are X1, X2, X3, and X4. Each predictor variable is a percentage integer of the four basic ingredients in cement. Table 4 shows the output for the

omcdiag function and Table 5 shows the output for the *imcdiag* function. The output for the *eigprop* function is shown in Table 6 and the output plots for the *mc.plot* function are displayed in Figure 2.

R code used to generate Table 4, 5, 6, and Figure 2

```
library("mctest")
data(Hald)
x  <- Hald[, -1]
 y <- Hald[, 1]
mctest(x,  y, type = "o",  Inter = FALSE)
mctest(x, y, type = "i")
eigprop(x)
mc.plot(x, y)
```

Table 4. Output of *omcdiag* function

```
Call:
omcdiag(x = x, y = y, Inter = FALSE, detr = detr, red = red,
    conf = conf, theil = theil, cn = cn)


Overall Multicollinearity Diagnostics

                       MC Results detection
Determinant |X'X|:         0.0011         1
Farrar Chi-Square:        59.8700         1
Red Indicator:             0.5414         1
Sum of Lambda Inverse:   622.3006         1
Theil's Method:            0.9981         1
Condition Number:          9.4325         0

1 --> COLLINEARITY is detected by the test
0 --> COLLINEARITY is not detected by the test
```

Table 5. Output of *imcdiag* function

```
Call:
imcdiag(x = x, y = y, method = method, corr = FALSE, vif = vif,
    tol = tol, conf = conf, cvif = cvif, leamer = leamer, all = all)


All Individual Multicollinearity Diagnostics Result

        VIF    TOL       Wi        Fi Leamer    CVIF Klein
X1  38.4962 0.0260 112.4886  187.4811 0.1612 -0.5846     0
X2 254.4232 0.0039 760.2695 1267.1158 0.0627 -3.8635     1
X3  46.8684 0.0213 137.6052  229.3419 0.1461 -0.7117     0
X4 282.5129 0.0035 844.5386 1407.5643 0.0595 -4.2900     1

1 --> COLLINEARITY is detected by the test
0 --> COLLINEARITY is not detected by the test

X1 , X2 , X3 , X4 , coefficient(s) are non-significant may be due to
multicollinearity

R-square of y on all x: 0.9824

* use method argument to check which regressors may be the reason of
collinearity
===================================
```

Table 6. Output of *eigprop* function

```
Call:
eigprop(x = x)

  Eigenvalues       CI Intercept     X1     X2     X3     X4
1      4.1197   1.0000    0.0000 0.0004 0.0000 0.0002 0.0000
2      0.5539   2.7272    0.0000 0.0100 0.0000 0.0027 0.0001
3      0.2887   3.7775    0.0000 0.0006 0.0003 0.0016 0.0017
4      0.0376  10.4621    0.0001 0.0574 0.0028 0.0457 0.0009
5      0.0001 249.5783    0.9999 0.9316 0.9969 0.9498 0.9973

==============================
Row 5==> X1, proportion 0.931570 >= 0.50
Row 5==> X2, proportion 0.996865 >= 0.50
Row 5==> X3, proportion 0.949846 >= 0.50
Row 5==> X4, proportion 0.997299 >= 0.50
```

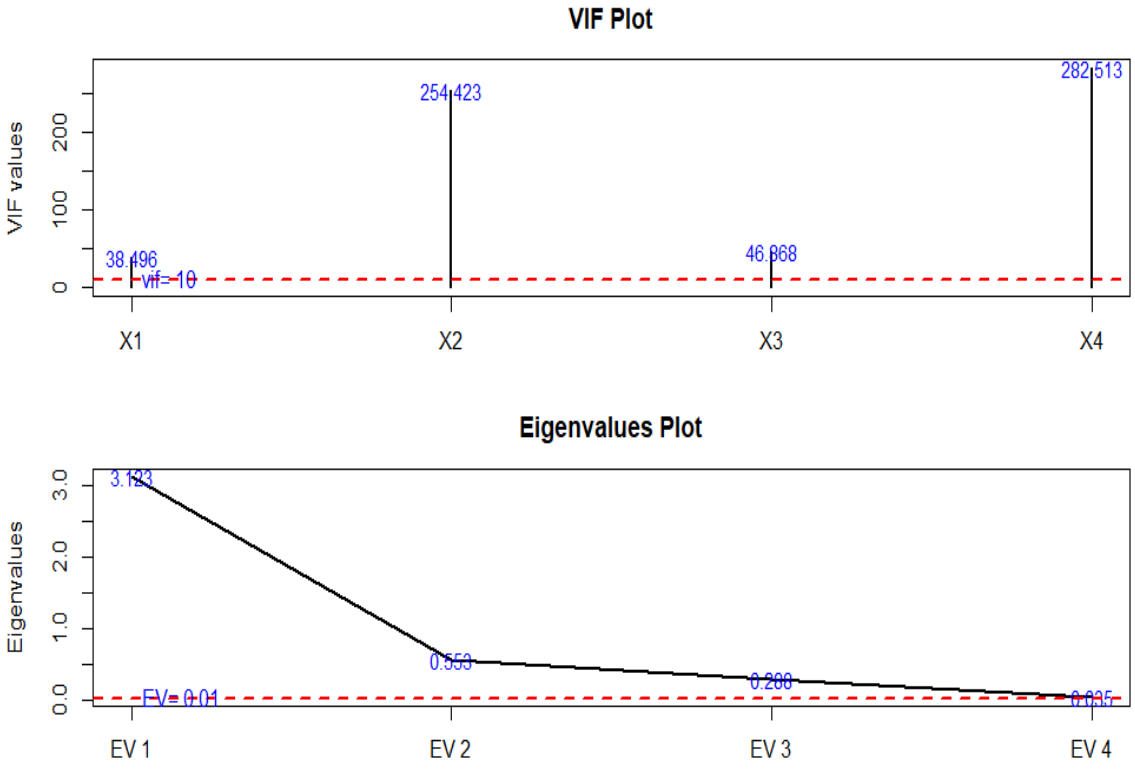

Figure 2. VIF and eigenvalues plots from the *mc.plot* function

5 of the 6 overall multicollinearity diagnostics in Table 4 indicate multicollinearity in the overall model. The condition number is the only overall diagnostic measure that does not indicate the presence of multicollinearity. However, the condition number calculated in the *eigprop* function in Table 6 does indicate a multicollinearity problem. Thus, there is a discrepancy between the two functions due to the source code. The *omcdiag* function scales the **X**-matrix before it calls upon the *eigprop* function to calculate the condition number. Therefore, the *omcdiag* function inadvertently scales the regressors twice before it calculates the condition number. Observing Table 6, we can identify condition index 5 is greater than 30. We can also identify each term that has a variance decomposition proportion greater than 0.5. Thus, all regressors, X1–X4, may be associated with a multicollinearity problem.

The individual diagnostic measures calculated by the *imcdiag* function are shown in Table 5. According to the default cutoffs used in the *mctest* package, the VIF, TOL, Wi, Fi, and Leamer's multicollinearity measure all indicate that each regressor contributes to the multicollinearity problem. The CVIF does not detect any regressor to be a source of multicollinearity and the Klein diagnostic only indicates X2 and X4 as the troubled regressors. Figure 2 displays the VIF plot and the eigenvalue plot. The VIFs calculated for the VIF plot are the same values calculated by the *imcdiag* function. However, the eigenvalues calculated for the eigenvalue plot are not the same for the *eigprop* function. This is because the inclusion of the intercept term is not included as a default for the *mc.plot* function. Each regressor variable in the "Hald" dataset may be related to a multicollinearity problem. Thus, each regressor should be evaluated further.

2.4 Strengths, weaknesses, and motivation for a new software package

The concept of perturbing the data to investigate the variability of the coefficients is good for detecting multicollinearity. Once noise is introduced, we can identify the regressors that have unstable coefficients. However, it is unclear how much noise is the right amount and an algorithm is missing for systematically perturbing the regressors. What the current perturbation strategy lacks is a way of relating the initial amount of noise to the output variation and how adding noise strategically can provide more information about the regressors. The *mctest* package does a good job providing multiple diagnostics and functionality for multicollinearity diagnostic analysis. What the *mctest* package lacks is a dynamic methodology, such as perturbation, to utilize the capability of the measures. Table

7 provides a check list of which diagnostics are covered for each package and the cutoff metric for each diagnostic.

Table 7. Diagnostics checklist by packages

	Name:	***mctest***	***perturb***	***mcperturb***	**cutoff**
1	Plotting the regressors density functions			✓	
2	Identifying implausible coefficients & standard errors			✓	
3	Perturbation analysis		✓	✓	
4	Identifying insignificant t-stats when overall F-test is significant			✓	
5	Change in the overall models R^2			✓	
6	Pair-wise correlation matrix				≥ 0.85
7	VIF	✓		✓	$\geq 5 \; or \; 10$
8	TOL limit	✓		✓	~ 0
9	F_j method	✓		✓	$F_j > F^*$
10	Leamer's method	✓		✓	$C_j \sim 0$
11	Corrected VIF	✓		✓	$\geq 5 \; or \; 10$
12	Kleins rule	✓		✓	$R_j^2 > R^2$
13	Variance decomposition proportions	✓	✓		≥ 0.5
14	Determinant of $\boldsymbol{X'X}$ matrix	✓		✓	~ 0
15	Farrar's $\boldsymbol{\chi^2}$	✓		✓	$\chi^2 > \chi^2_{\alpha,v}$
16	Red's indicator	✓		✓	RED ~ 1
17	Sum of the inverse λ	✓		✓	$\geq 5 * p$
18	Theil's indicator	✓		✓	M ~ 1
19	Condition number & condition indices	✓	✓	✓	≥ 30
20	Eigenvalues	✓		✓	~ 0

In Table 7, p is the number of regressors. n is the number of observations. α is the significance level. R_j^2 is the coefficient of determination from the auxiliary regression for the j[th] regressor. The definition or math formulas of key statistics listed in the last column of Table 7 are shown below.

For row 9 of Table 7,

$$F_j = \frac{R_j^2}{1 - R_j^2} \times \frac{n - p + 1}{p - 2} \sim F^* = F_{\alpha, p-2, n-p+1}.$$

For row 10 of Table 7,

$$C_j = \sqrt{\frac{\left(\sum_1^n (X_{\boldsymbol{ij}} - \bar{X}_{\boldsymbol{j}})^{\boldsymbol{2}}\right)^{\boldsymbol{-1}}}{(\boldsymbol{X'X})_{\boldsymbol{jj}}^{\boldsymbol{-1}}}}.$$

For row 15 of Table 7,

$$\boldsymbol{\chi^2} = -\left[n - 1 - \frac{1}{6(2p+5)}\right] \times \log_e |\boldsymbol{X'X}|,$$

and

$$v = \frac{1}{2} p(p-1)$$

degrees of freedom.

From row 16 of Table 7,

$$RED = \frac{\sqrt{p-1}}{p} \times \sqrt{\sum_{i=1}^{p} (\lambda_j - 1)^2}.$$

For row 18 of Table 7,

$$M = R^2 - \sum_{i=1}^{p} (R^2 - R_{-i}^2) \,,$$

where R_{-i}^2 is the regression model without the i[th] regressor.

The cutoff value for each diagnostic in the last column of Table 7 can be found throughout multiple literatures. The statistical and numerical diagnostic measure equations and cutoff values are suggested in Imdadullah et al. (2016). The pair-wise correlation matrix, variance decomposition proportions, and condition numbers/indices cutoff values can be found in Montgomery et al. (2012). The VIF and TOL limit values can be found in Ott and Longnecker (2015).

Regarding the information the multicollinearity diagnostic calculates, each package has its own strengths and limitations. Enhancing the abilities of both packages by combining their advantages is a part of the motivation behind the *mcperturb* package. Thus, we take the underlying concept of the perturbation strategy as a multicollinearity diagnostic and adapt the strategy to encompass the diagnostic measures in the *mctest* package. Therefore, the main functions for both packages will be used in the *mcperturb* package.

III. A NEW PACKAGE

3.1 Introduction

The main goal of the *mcperturb* package is to diagnose multicollinearity. In order to achieve this goal, we execute a perturbation strategy/analysis and include an observational strategy/analysis. Performing perturbation analysis before calculating diagnostic measures can provide a dynamic element to the otherwise static information obtained by calculating diagnostic measures. That is, performing multicollinearity diagnostic measures (static) after applying small perturbations to the regressors (dynamic) may lead to a more in-depth analysis of a multicollinearity problem. In addition to the perturbation analysis, including the observational analysis is one of the new contributions of this *mcperturb* package as it is not included in the currently available software packages.

The *mcperturb* package diagnoses multicollinearity by calculating both the overall and individual diagnostic measures after perturbation. This package consists of 5-steps as shown in Table 8 and explained below.

Step 1: Perform observational analysis

Step 2: Perform perturbation analysis

Step 3: Calculate the overall or individual diagnostic measures and plot their distributions

Step 4: Conduct summary analysis for each regressor and calculate the rate of change

Step 5: Rank the overall diagnostics and/or identify coupling regressors

The *mcperturb* package is a structured approach that combines both the observational strategy/analysis and perturbation strategy/analysis. It begins with the observational strategy/analysis, and then combines a perturbation technique with the overall or individual diagnostic measures. It systematically perturbs the regressors sequentially at different noise levels before calculating the diagnostic measures. The *mcperturb* package includes functions designed specifically for each step of the 5-step multicollinearity diagnostic procedure. It generates output summary tables, graphs, or boxplots for interpretation. Performing the 5-step multicollinearity diagnostic procedure can generate a plethora of information, especially when including a large number of regressors for analysis. To make it easier for the users, we will provide examples to show how to utilize the *mcperturb* package and interpret the results.

3.2 R function overview and the 5-step chart

Table 8 outlines the new 5-step multicollinearity diagnostic method. It summarizes each step, lists the functions associated with each step, and lists the arguments for each function. The first step of the 5-step procedure is to perform observational analysis. The functions for performing observational analysis are *densPlots*, *rsqdPlots,* and *implausStats*. Providing graphical output for interpreting multicollinearity is the main goal for the observational analysis. The second step is to perform the perturbation analysis. The function used for the perturbation analysis is the *noiseLevelDiagOutList* function. The third

step of the new strategy is to calculate and plot the distributions for the overall diagnostic measures or the individual diagnostic measures. The *overallDiagsPlots* function is used for displaying the distributions for the overall diagnostics calculated at each noise level. The *BoxplotsAllVars* function is used for displaying the distributions for the individual diagnostics calculated at each noise level. The *BoxplotsAllPercent* function is used for displaying the distributions of an individual regressor per diagnostic measure at each noise level. Using either the mean or median values, step four displays the minimum, maximum, and difference for each regressor per diagnostic for all noise levels. The *overallDiagOut* and *mcperSumTables* functions are used for displaying summary tables for step four. Finally, using the *overallDiagsRank*, isR*ateOfChange*, and *isBestFit* functions, step five calculates the rate of change, least squares best fit line, and rank sum values.

Table 8. 5-steps of the *mcperturb* package

Steps	Diagnostic/Purpose	R Functions	Arguments
1. Perform observational analysis	Identify regressors with similar density functions, implausible coefficients, inflated standard errors, and little impact on the R^2	*densPlots*, *implausStats*, *rsqdPlots*	x - matrix of regressors, y - response variable
2. Perform perturbation analysis	Add small amounts of noise to each regressor at multiple levels	*noiseLevelDiagOutList*	x - matrix of regressors, y - response variable, i - # of iterations, n - # of noise levels
3. Calculate the overall/individual diagnostic measures and plot their distributions	Observe how small perturbation affects the diagnostic measures	*overallDiagsPlots*, *BoxplotsAllVars*, *BoxplotsAllPercent*	x - matrix of regressors, y - response variable, i - # of iterations, n - # of noise levels, p - path
4. Conduct summary analysis for each regressor and calculate the rate of change	Summarize the max, min, and difference values for each diagnostic and rate of change	*overallDiagOut*, *mcperSumTables*	x - matrix of regressors, y - response variable, i - # of iterations, n - # of noise levels
5. Rank the overall diagnostics and/or identify coupling regressors	Rank the overall diagnostics by their impact on the model and identify coupling regressors.	*overallDiagsRank*, isR*ateOfChange*, *isBestFit*	x - matrix of regressors, y - response variable, i - # of iterations, n - # of noise levels

3.3 Body dimension dataset

Before performing step 1 of the 5-sep procedure, we introduce the body dimension dataset used for analysis. This dataset is published as an observational study (Grete et al., 2003). The dataset is collected by the original authors, Grete Heinz and Louis J. Peterson, at San Jose State University and at the U.S. Naval Postgraduate School in Monterey California. The authors investigated relationships between individual's body frame size, frame girths, and weight of active adults in the military. Because multiple body measurements are performed on the same individual who participated in the study, the high correlation between variables is inevitable.

The original body dimension dataset consists of 25 variables (body measurements) and 507 observations (profiles). From the 25 body measurements, the weight measurement is selected as the response variable and the shoulder diameter, chest girth, bicep girth, forearm girth, wrist minimum girth, height, and age variables are selected as the regressors for this thesis project. A summary of the subset of variables used for analysis is listed in Table 9.

R code for generating Table 9.

```
x = body.dat[, c(11, 12, 16, 17, 21, 22, 24]
colnames(x) = c("shoulder", "chest", "bicep", "forearm", "wrist", "age", "height")
summary(x)
```

Table 9. Summary statistics for a subset of the body dimension dataset

Variable Names	**Units**	**Type**	**Min**	**1st Qu**	**Median**	**Mean**	**3rd Qu**	**Max**
Shoulder diameter	cm	Continuous	85.9	99.5	108.3	108.2	116.6	134.8
Chest girth	cm	Continuous	72.6	85.3	91.8	93.42	101.2	118.7
Bicep girth	cm	Continuous	22.4	27.6	31	31.2	34.5	42.4
Forearm girth	cm	Continuous	19.6	23.6	25.8	25.97	28.4	32.5
Wrist minimum girth	cm	Continuous	13	15	16.1	16.11	17.1	19.6
Age	yrs.	Quantitative	18	23	27	30.36	36	67
Weight	kg	Continuous	42	58.2	68.2	69.28	79.1	116.4
Height	cm	Continuous	147.2	164	170.2	171.1	177.8	198.1

The correlations between the regressor variables are displayed in a correlation plot in Figure 3. The shoulder diameter, chest girth, bicep girth, forearm girth, and wrist minimum girth variables are chosen for multicollinearity analysis because of the high correlations between each other. The height variable is included for analysis because of its moderate correlations with the other regressors. The age variable is selected because of its low correlations with the other regressors. Using the correlation matrix to identify interdependencies, we can hypothesize that multicollinearity may exist among the regressors because of their high intercorrelations. The correlation matrix in Figure 3 shows the intercorrelations are high between the shoulder, chest, bicep, forearm, and wrist regressors.

```
R code for generating Figure 3.
library("corrgram")
corrgram(x, lower.panel = panel.shade, upper.panel = panel.cor, col.regions =
colorRampPalette(c("orange",  "yellow",  "green",  "blue",  "black")))

title(main = "Pairwise Correlations")
```

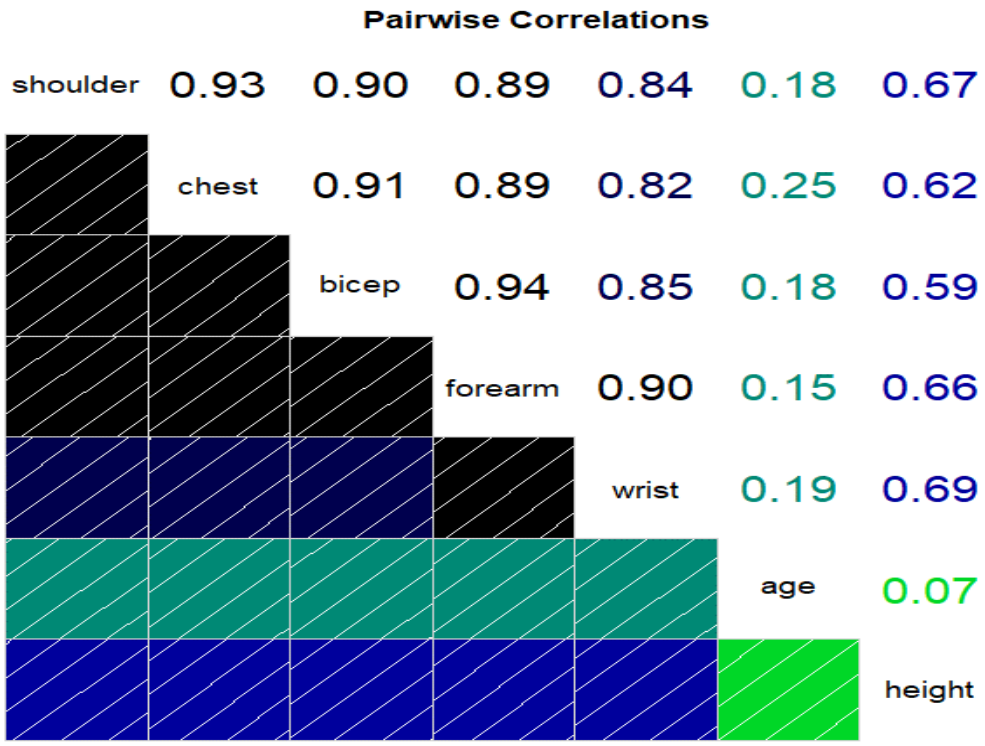

Figure 3. Correlation matrix of the regressors

3.4 Step 1. Perform observational analysis

When diagnosing multicollinearity, the *mcperturb* package provides the functions *densPlots*, *implausStats*, and *rsqdPlots* for the observational analysis. These functions plot the regressor's density functions, display implausible coefficients & standard errors, and plot the change in the overall model R^2 value respectively. By providing graphical outputs,

the observational analysis of the *mcperturb* package may indicate a multicollinearity problem. However, performing observational analysis does not offer a measure for the degree of the multicollinearity problem.

The first observational analysis is not used as a multicollinearity diagnostic strategy in currently available packages. However, plotting each regressor's density function is useful for exploring how regressors relate to each other. The *densPlots* function performs the observational analysis and can be used for finding patterns in the regressors' density plots. These plots can help us observe patterns for diagnosing multicollinearity by identifying regressors who share similar variances and have similar inflection points in their distributions. The steps for performing the *densPlots* function are:

a. Mean center the regressors
b. Plot the estimated density functions on the same graph
c. Identify the variables that have similar spreads
d. Identify the variables that have similar inflection points

Note: The regressor variables do not have to be mean centered in order to perform this type of observational analysis. However, it is much easier to compare multiple regressors' density plots on the same graph when each regressor distribution is centered around 0.

Using the body dimension dataset, we show a graph consisting of the mean centered density plots for the regressors listed in Table 9, see Figure 4.

R code for generating Figure 4.
densPlots(x, TRUE)

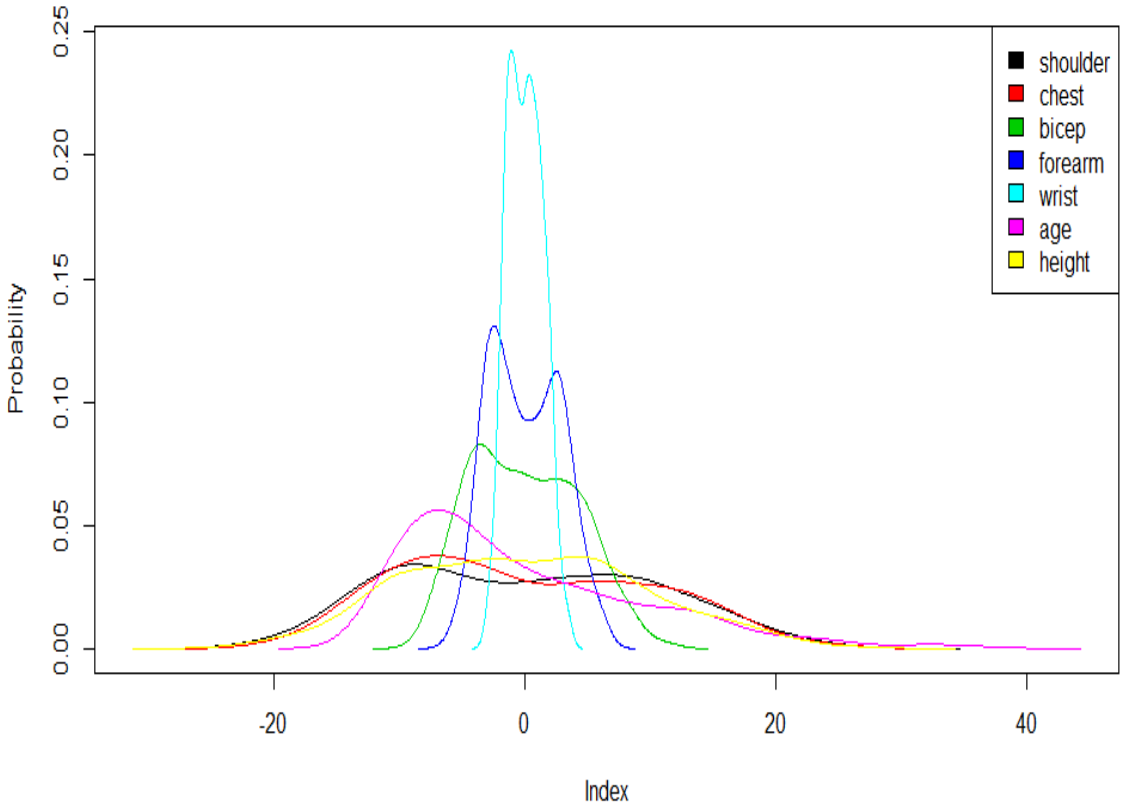

Figure 4. The mean-centered density plots for the body dimension dataset

Based on Figure 4, we can identify the regressors that have relatively narrow density functions and that show similar distribution patterns. The regressors with relatively narrow density functions should be identified because their small variances might be able to help us identify a source of a multicollinearity problem. Regressors with small variances may only be able to provide a small amount of information to the MLR model. If this is the case, the limited information these regressors may provide should be as unique as possible

to avoid being redundant with the information provided from the other regressors. When regressors are selected from the same population, it is possible for their density plots to display similar distributions. If this is the case, the information these regressors can provide in the MLR model might be redundant. The regressors that have narrow density functions and that display similarities in their density plots might be more likely to provide the MLR model with redundant information and be a source of multicollinearity.

In Figure 4, the regressors with relatively narrow density plots are wrist minimum girth, forearm girth, and bicep girth. In addition, in Figure 4, there is a distinct bimodal pattern occurring among the regressors. Multiple regressors appear to have a natural separation, i.e., bimodal pattern, in their density plots. These regressors are shoulder diameter, chest girth, forearm girth, and wrist minimum girth. Thus, the two regressors with narrow ranges of their density plots and similar patterns in their density functions are wrist min girth and forearm girth.

We investigate further and find that the bimodal pattern may be due to a latent variable missing from the analysis. We identify the latent variable as the gender variable. The natural separation in the regressors may be influenced by the differences between two subpopulations, male and female. Once the regressors are separated by gender, the bimodal pattern does not exist anymore.

Next, we continue the diagnosis of multicollinearity by performing the second observational analysis using the *densPlot* function, see Figure 5. This figure displays the

density plots for two regressors of interest, shoulder diameter and chest girth. In Figure 4, shoulder diameter and chest girth are identified as two regressors that have very similar density functions because of their bimodal distributions and spread. Due to their similar density functions, it is likely that these regressors are associated with a multicollinearity problem. Although this may be generally true, observational analysis for detecting multicollinearity should be investigated on a case-by-case basis.

R code for generating Figure 5.
densPlots(x[, 1:2], TRUE)

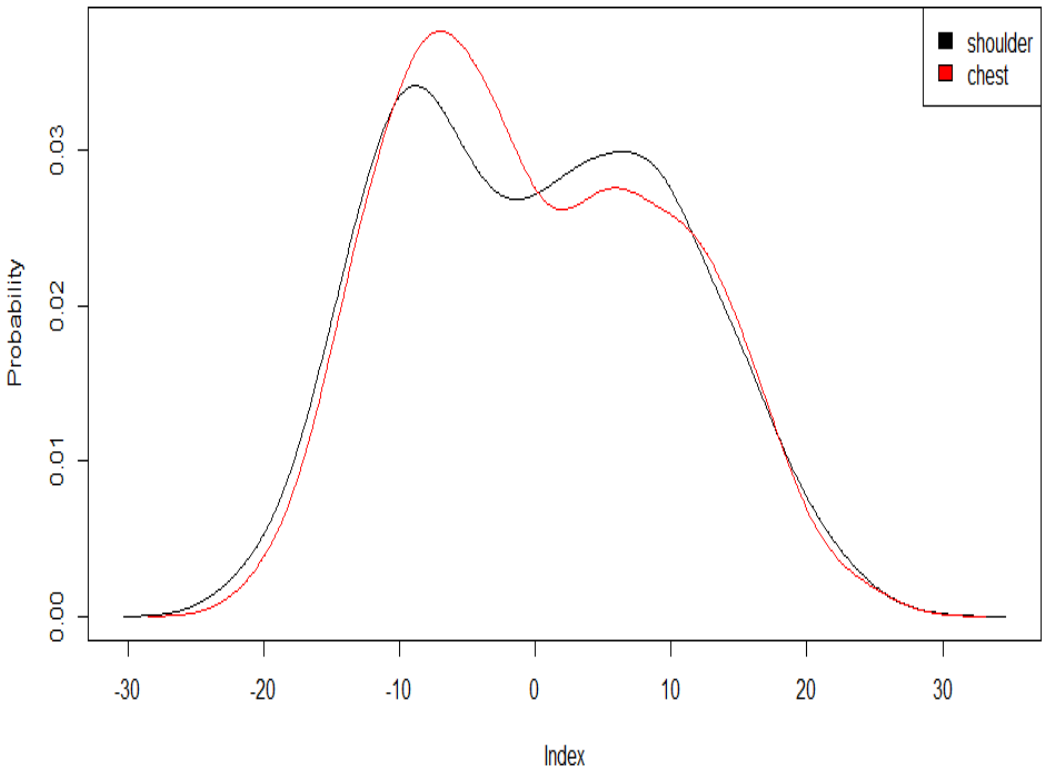

Figure 5. The mean-centered density function for shoulder diameter and chest girth

Identifying implausible coefficients and standard errors is the next observational analysis we will explore using the *mcperturb* package. The procedure behind this observational analysis is to perform a SLR model for each regressor and an MLR model using all of the regressors, then identify the regressors whose coefficients and standard errors are inconsistent in the SLR and MLR model. The *implausStats* function can output a summary table that displays the necessary statistics for performing the identification of implausible coefficients and standard errors. This function is described in detail below:

a. Calculate the correlations between each regressor and the response variable
b. Calculate the SLR model coefficients and standard errors
c. Calculate the MLR model coefficients and standard errors

The rationale behind the observational strategy/analysis is that the correlations between each regressor and the response variable should correspond with the coefficients estimated by the SLR models, and the parameter estimation from the SLR model should be consistent with the MLR model parameter estimation. A common example of an inconsistency that is often attributed to multicollinearity is when a regressor's coefficient has a wrong sign. For example, the coefficient in the SLR model is positive, but it changes to negative in the MLR model. Identifying any implausible coefficients, inflated standard errors, and inconsistencies may indicate a possible multicollinearity problem. Next, we use the body dimension variables to show a potential multicollinearity problem, see Table 10 for the output from the *implausStats* function.

R-code for generating Table 10.
implausStats(x, y)

Table 10. Summary table of implausible coefficients & standard errors

	Resp.corr	SLR.coeff	MLR.coeff	SLR.std.err	MLR.std.err
shoulder	0.8788	1.130	0.0908	0.0273	0.0648
chest	0.8989	1.196	0.6023	0.0259	0.0685
bicep	0.8666	2.723	0.4674	0.0697	0.1801
forearm	0.8695	4.099	0.6479	0.1036	0.2997
wrist	0.8164	7.890	-0.3194	0.2482	0.4011
age	0.2072	0.2878	0.0357	0.0604	0.0244
height	0.7173	1.017	0.3316	0.0439	0.03407

Observing the statistics for the regressors in Table 10, we can identify an implausible coefficient change for the wrist regressor. The wrist regressor's SLR coefficient, 7.890, changes dubiously to -0.3194 in the MLR model. This implausible coefficient change may be attributed to the MLR standard error for wrist 0.4011 being larger than the magnitude of its MLR coefficient. Because of its implausible MLR coefficient, the wrist regressor may be identified as a potential regressor that may be related to a multicollinearity problem. In Table 10, we can also identify the regressors with inflated standard errors. Because the magnitudes of their MLR standard errors are significantly larger than the magnitudes of their SLR standard errors, the shoulder, forearm, and wrist regressors can be identified as the regressors with inflated standard errors. These regressors with inflated standard errors may influence their associated regressor coefficients to be insignificant. These inflated standard errors may be an indirect indicator that a multicollinearity problem exists in the MLR model.

The third observational analysis offered by the *mcperturb* package is a summary of the overall model's R^2 or adjusted R^2 values as more variables are included in the MLR. The function used to perform this analysis is *rsqdPlots* function. This function calculates the overall model's R^2 or adjusted R^2 values as each regressor is sequentially added into the MLR model by performing the following steps:

a. Rank the regressors by their correlations with the response variable.
b. Add in one regressor at a time into the MLR model and calculate the R^2 or adjusted R^2.
c. Plot the R^2 values and identify patterns with a small change of the slope.

Using the body dimension dataset, we show the output from the *rsqdPlots* function, see Figure 6.

```
R code for generating Figure 6
rsqdPlots(x, y, T)
```

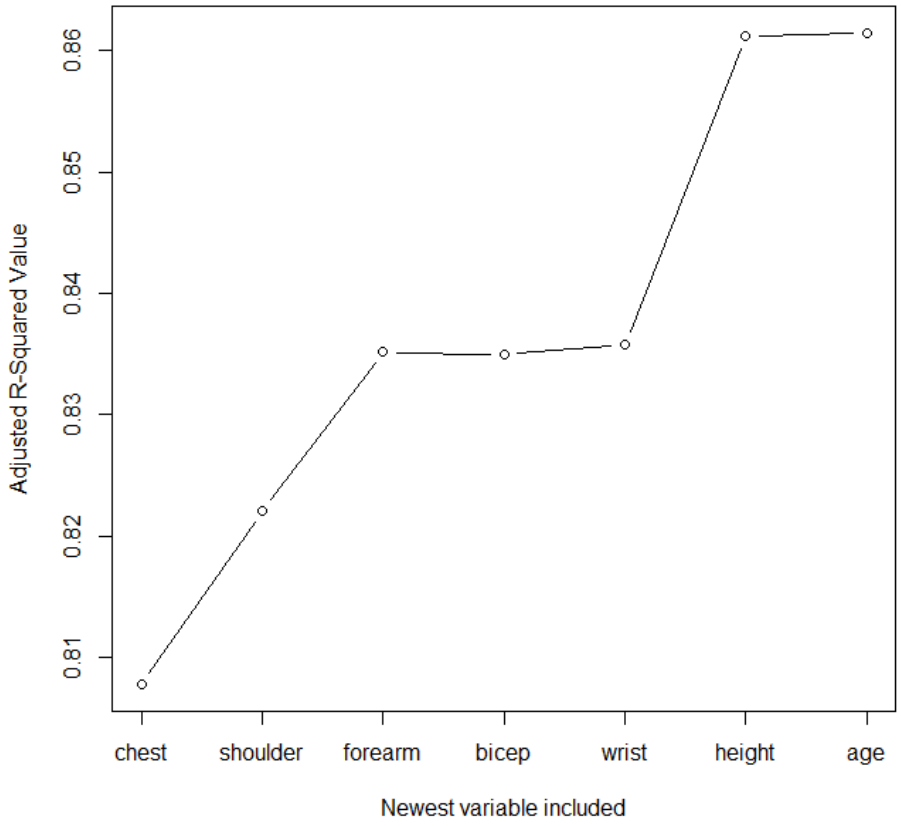

Figure 6. MLR adjusted R^2 values

If there is no significant change in the R^2 and/or adjusted R^2 values as regressors are sequentially added into the MLR model, this may imply that no more variation accounted by the newly added variable is being introduced into the model. Therefore, these regressors may be identified as regressors providing redundant information. When redundancy exists in the MLR model, the model may have a multicollinearity problem. In Figure 6, when the bicep, wrist, or age regressors is introduced into the MLR model, there seems to be no significant change in the R^2 and adjusted R^2 values. The inclusion of these regressors in the overall MLR model may be related to a multicollinearity problem because the model is possibly overfitted.

In summary, the three observational analyses offered by the *mcperturb* package are plotting the density functions, identifying the implausible coefficients and standard errors, and plotting the R-squared values as more variables are included in the model. Each analysis has its advantages and can be used to summarize different aspects of regression analysis. After the observational analysis is performed on the body dimension dataset, we now know the variables of interest we would like to investigate further. The regressors that may contribute to a multicollinearity problem according to the observational analysis are shoulder diameter, chest girth, bicep girth, forearm girth, and the wrist minimum girth.

3.5 Step 2. Perform perturbation analysis

From the observational analysis in step 1, we are able to identify the regressors that may associate with a multicollinearity problem. In step 2 of the 5-step strategy, small perturbations of random noise are systematically applied to each regressor. The *noiseLevelDiagOutList* function performs the perturbation analysis by perturbing each regressor sequentially for multiple iterations at different noise levels. Because the body dimension dataset consists of continuous regressors, random normally distributed noise is applied to each regressor. Using an iterative process, the noise levels are calculated with respect to the noise regressor. The iterative process is as follows:

$$\text{noise level } 1 = \text{ noise start} \times \text{sd(regressor)}$$

$$\text{noise level } 2 = \text{ noise level } 1 + \text{noise step}$$

...

$$\text{noise level n} = \text{ noise level } (\text{n} - 1) + \text{noise step}$$

where noise start is the initial percentage of noise (i.e., 5%), noise step is the increase of percentage (i.e., 5%), and

$$n = \frac{noise\ end - noise\ start}{noise\ step} + 1$$

The output for the *noiseLevelDiagOutList* function is a list of i noise matrices, each noise level is a list of *j* noise levels, and for each noise level, it includes a list of k noise regressors. Therefore, this function's output is a list of lists. That is, it generates a total of $i \times j \times$ k matrices structured in a list of lists. Using each noise matrix, we calculate the overall and individual diagnostic measures. Calculating the multicollinearity diagnostics measures after applying this perturbation analysis can provide even more information for multicollinearity diagnostics.

3.6 Step 3. Calculate the overall or individual diagnostic measures and plot their distributions

After performing step 1 and step 2, more information about the multicollinearity problem can be generated by calculating the overall and/or individual diagnostic measures for each resulting matrix and plotting their distributions. The third step of the 5-step strategy is to plot the resulting distributions after calculating the overall and individual diagnostic measures. We sequentially perturb each regressor for i iterations, e.g., $i = 50$, at each noise level before plotting the calculated diagnostic measures. The *overallDiagsPots* function provides a graphical output of the distribution for each noise level and for each overall multicollinearity diagnostic per individual noise regressor. The distributions for the

determinant diagnostic at each noise level are displayed in the boxplots in Figure 7.

```
R code for generating Figure 7
for (i in 1:dim(x)[2]){
  special.Vars = colnames(x)[i]
  overallDiagsPlots(xmat = x, yvar = y, noiseLevels = noiseLevs, spec.Vars =
special.Vars, iter =   iteration, choiceDig = c("determinant"))
}
```

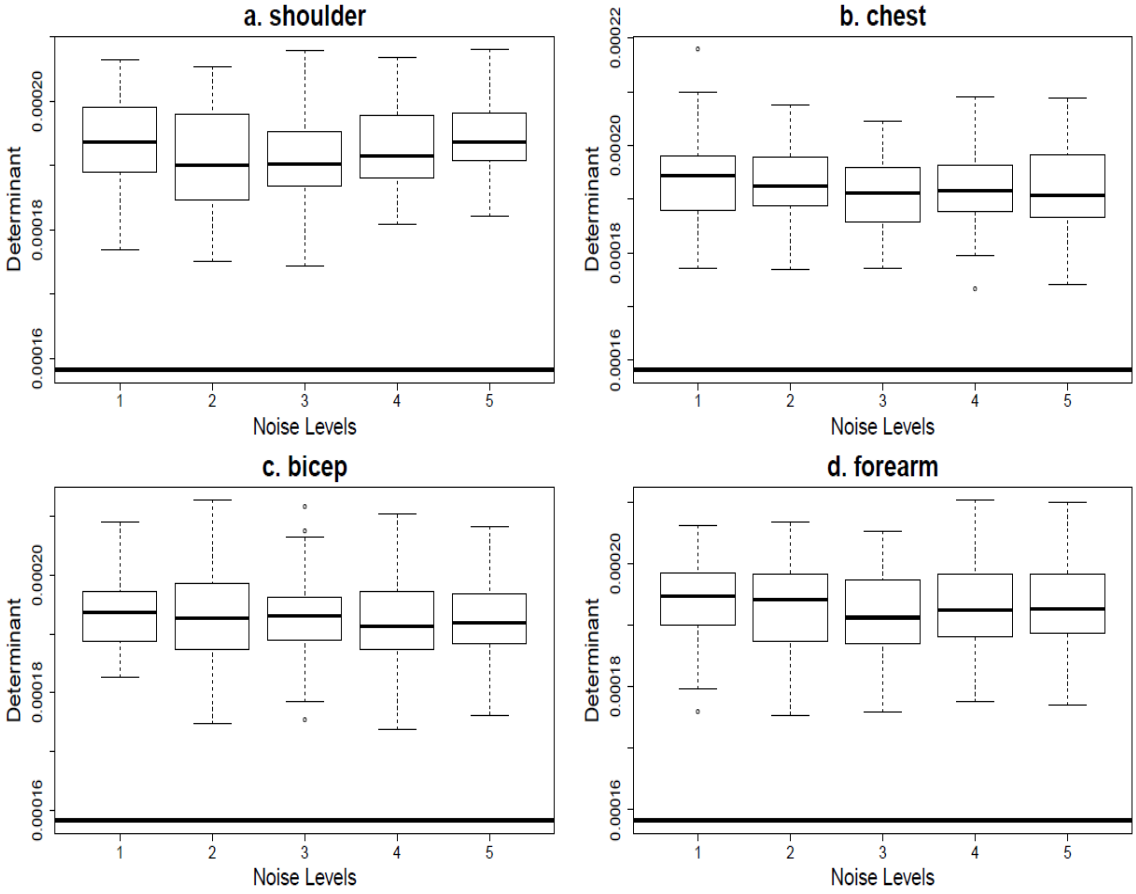

Figure 7. Boxplots of the determinant of $\boldsymbol{X'X}$ with respect to the noise variable

For each boxplot in Figure 7, the change in the determinant does not increase as the noise levels increase. This may be because the determinant measure is dependent on multiple

regressors. Therefore, adding noise to one regressor does not significantly affect the determinant. The distribution at each noise level follows a normal distribution, and the mean is used when the difference is calculated between the original determinant and the determinant at an individual noise level.

Plotting the distributions for the individual diagnostic measures can be performed in two different ways. The first way is to plot all the regressors with respect to each diagnostic at multiple noise levels. The second way is to plot the change in one regressor over the different noise levels. *boxplotAllPerc* is the function used to display the individual diagnostics with noise added to a single regressor. The *boxplotAllPerc* function outputs a boxplot of the distributions of each regressor at different noise levels per diagnostic measure. Figure 8 shows the different distributions for each regressor as the shoulder regressor is being perturbed and the VIF is the diagnostic of interest.

```
R code for generating Figure 8
special.Vars = c("shoulder")
boxplotoutperc = BoxplotAllPerc(xmatrix = x, y = y, noiseLevs = noiseLevs, special.Vars = special.Vars, iteration = iteration)
```

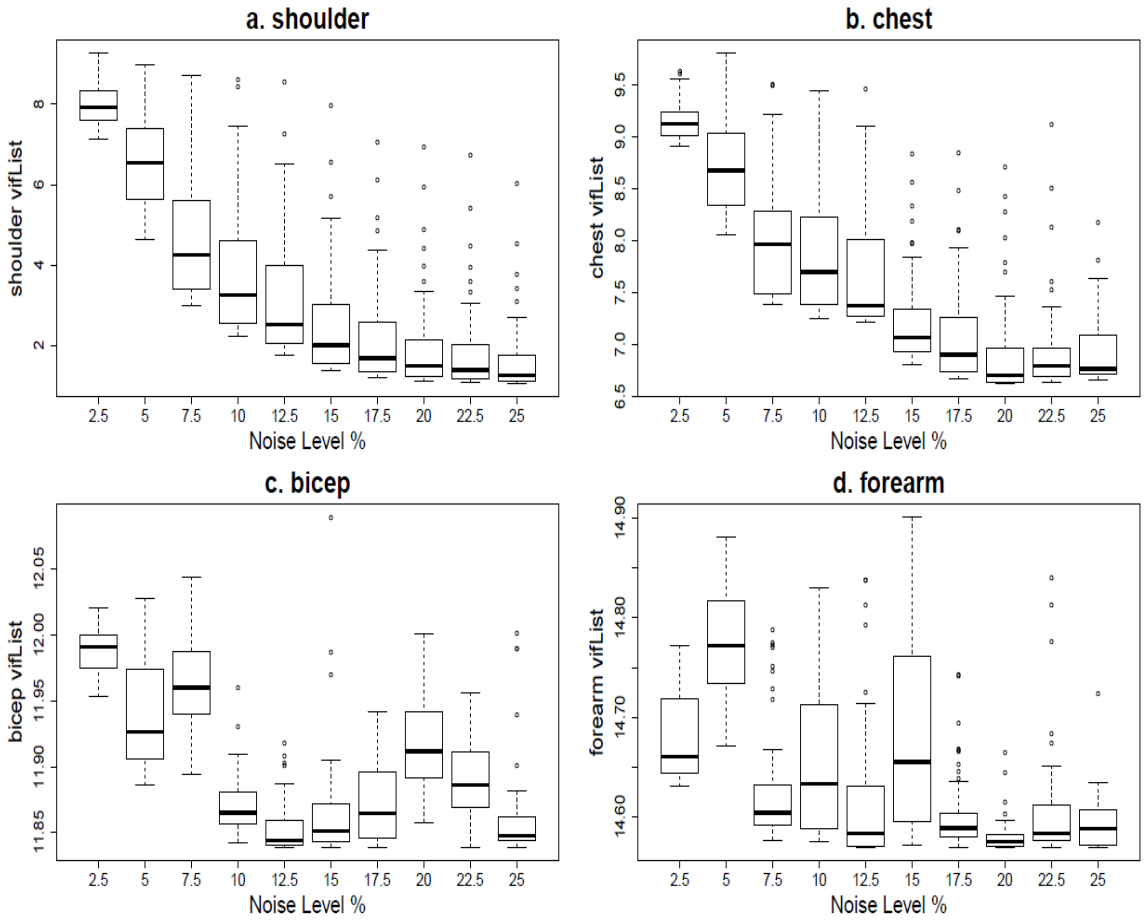

Figure 8. Boxplots of variables' vifList with noise added to the shoulder regressor

From Figure 8, observing the calculated VIF distributions of the shoulder diameter and chest girth, we can identify a significant decrease in their VIFs with respect to the increase of noise levels and the noise variable shoulder. Recalling the density plots in Figure 5, these two regressors have very similar distributions. Thus, now there is more evidence to suggest that as the shoulder diameter gets perturbed, the regressor that acts accordingly is the regressor, chest.

For each diagnostic and regressor, we can observe the change in the distribution at every noise level. Figure 9 displays the output boxplots from the function *boxplotAllVars*. In

Figure 9, the perturbed regressor is shoulder diameter and the diagnostic of interest is VIF. The distributions of VIFs for each regressor are plotted for each noise level. Figure 9 shows that the regressor being perturbed, shoulder diameter, has the most variation of all the calculated VIF distributions. In Figure 9, we can identify the regressors that have noticeable changes to their VIF, i.e., shoulder diameter and chest girth.

```
R code for generating Figure 9
special.Vars = c("shoulder")
boxplotout = boxplotsAllVars(xmatrix = x, y = y, noiseLevs = noiseLevs, special.Vars =
special.Vars, iteration = iteration)
```

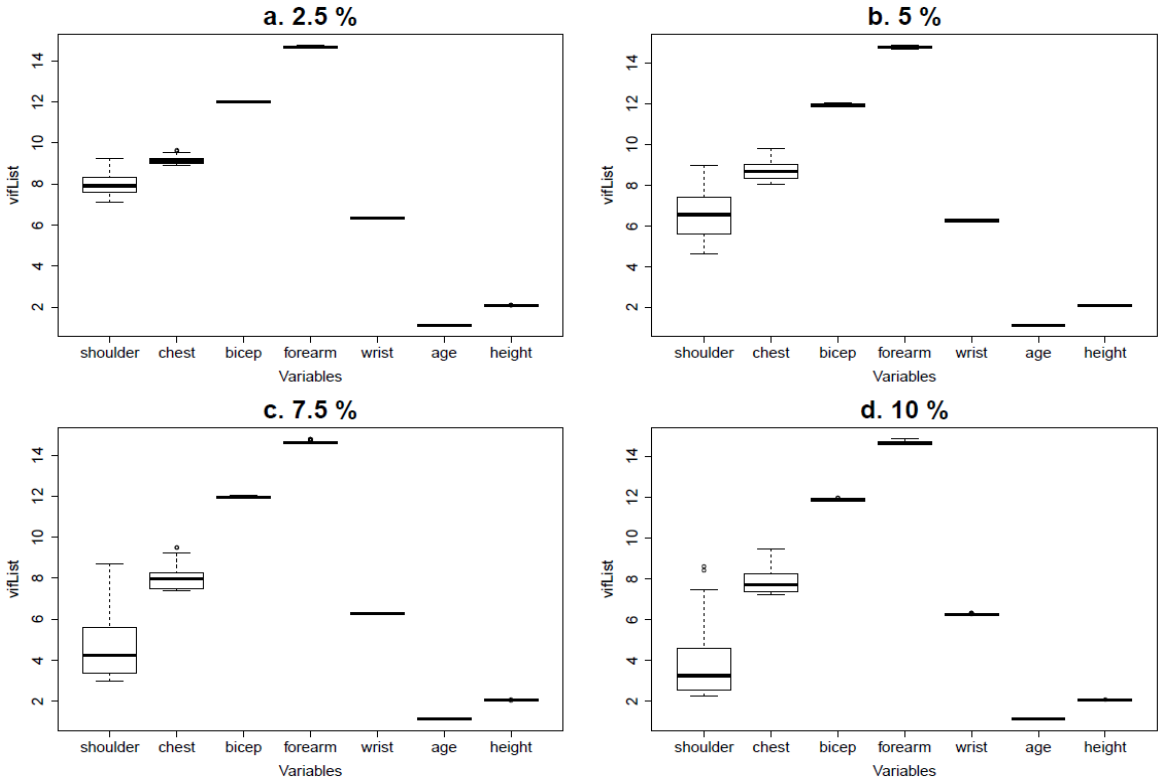

Figure 9. Boxplots of the VIF's for all the regressors at multiple noise levels

3.7 Step 4. Conduct summary analysis for each regressor and calculate the rate of change

The *overallDiagOut* function provides a table displaying the minimum of calculated means, maximum means, and the difference between the maximum and minimum when each regressor is the noise variable. The original diagnostic measure is used in the search for the maximum or minimum values. Table 11 shows the output for the *overallDiagOut* function when the determinant of the $\boldsymbol{X'X}$ is the diagnostic of interest.

```
R code for generating Table 11
overallDiagsSummaryTable(x = x, y = y, noiseLevs = noiseLevs, iteration = iteration)
```

Table 11. Output table of minimum, maximum, and max/min difference for the determinant

Noise.variable	Min.mean	Max.mean	Difference
shoulder	0.000158	0.000193	3.531e-05
chest	0.000158	0.000197	3.905e-05
bicep	0.000158	0.000204	4.675e-05
forearm	0.000158	0.000217	5.913e-05
wrist	0.000158	0.000181	2.36e-05
age	0.000158	0.000158	5.881e-07
height	0.000158	0.000163	4.909e-06

In table 11, the minimum mean for each regressor is the original determinant before perturbation analysis. The maximum mean for each regressor does not vary much and is relatively close to the original mean. Thus, the differences between the maximum and minimum means are relatively small and may be an insignificant change.

The *mcperSumTables* function provides an output summary table for each individual diagnostic with respect to a single noise regressor. Table 12 displays the VIFs for each regressor as the shoulder diameter regressor is perturbed at multiple noise levels. The medians are used for the analysis because the distributions shown in Figure 8 do not seem to follow a normal distribution. The original VIF for each regressor is used to calculate the difference between the maximum and minimum median, least-squares fit, and rate of change values.

Because the VIF distributions seem to be changing in a linear fashion for the body dimension dataset, the rate of change line and a least square best fit line are calculated in step 4. The rate of change is calculated by taking the ratio of the difference between the maximum and minimum medians and noise levels. The least squares best fit line is calculated by using the noise levels as the regressors and the VIF values as the response variable. The rate of change and the best fit values measure the rate a regressor changes as the noise change. Both values will agree with each other when the VIF values change linearly with respect to the noise levels.

In order to interpret the rate of change and best fit values, we must first identify which regressors have a significant original VIF value. A VIF greater than 10 may be used as a threshold for identifying regressors causing a multicollinearity problem (Kleinbaum et al., 2007). Using 10 as a cutoff value, we can identify bicep and forearm as the regressors that are significantly causing a multicollinearity problem. After noise is added to the shoulder

regressor, we can observe in Table 12 that even though bicep and forearm have the largest original VIF values, this does not imply they are going to have the greatest change of VIF values.

```
R code for generating Table 12
summaryTableList = rateOfChange(x = x, y = y, noiseLevs = noiseLevs, special.Vars = special.Vars, iteration = iteration)
```

Table 12. Summary table of VIF diagnostics with the variable shoulder perturbed

```
$`VIF Table`
         Original-Values Diff-Median Least-squares-fit Rate-of-Change
shoulder           9.293       8.024            -3.057         -3.094
chest              9.691       2.983            -1.166         -1.150
bicep             12.002       0.161            -0.069         -0.062
forearm           14.764       0.181            -0.041         -0.070
wrist              6.293       0.056            -0.014         -0.022
age                1.133       0.010            -0.003         -0.004
height             2.108       0.085            -0.031         -0.033
```

3.8 Step 5. Rank the overall diagnostics and/or identify coupling regressors

The *overallDiagRank* function outputs a table of rankings for each of the differences between the mean of the regressor and the original calculated diagnostic measures. Each regressor is ranked by how much of an influence it has on an overall diagnostic measure when it is the variable being perturbed. Table 11 shows the magnitude of the difference between maximum and minimum mean determinants for each noise regressor. We take the magnitudes found in Table 11 and rank them. Table 13 summarizes the ranking of the differences for each diagnostic. The lower the ranking is for a regressor, the greater the difference is. In table 13, we calculate the rank sum for each regressor with respect to each

diagnostic and conduct a final ranking of these sums.

R code for generating Table 13
overallDiagsPlots(xmat = x, yvar = y, noiseLevels = noiseLevs, spec.Vars = special.Vars, iter = iteration, choiceDig = c("d"))

Table 13. Overall diagnostic ranks by differences

Noise.variable	Det	ChiSqr	RedInd	SumOfLam	TheilInd	Condition	RankSum	Overall
Shoulder	4	4	4	4	2	5	23	4
chest	3	3	2	3	5	4	20	3
bicep	2	2	3	2	3	2	14	2
forearm	1	1	1	1	1	1	6	1
wrist	5	5	5	5	4	3	27	5
age	7	7	7	7	7	6	41	7
height	6	6	6	6	6	7	37	6

In table 13 we can identify forearm as the regressor that has the greatest overall impact on the overall diagnostic measures, which suggests forearm may have the greatest impact on the overall model's multicollinearity issue.

Using the functions isR*ateOfChange* and *isBestFit*, we can calculate the least squares best fit and rate of change values for each diagnostic. Each function will display a list of summary tables for the individual diagnostic. Table 14 displays the least squares best fit values for the VIF diagnostic after each regressor is perturbed. The least squares best fit values can be calculated for the VIF diagnostic and the body dimension dataset, but it may not be a useful if linearity is not satisfied for other datasets and diagnostics. Therefore, we also calculate and report the average rate of change values as a consistency check for the least squares best fit calculation and as a non-parametric analysis. Table 15 displays the rate of change for the VIF diagnostic after each regressor is perturbed.

```
R code for generating Table 14 and 15
VifLeastSqrsMat = matrix(NA, nrow = 7, ncol = 7)
VifRateOfChangeMat = matrix(NA, nrow = 7, ncol = 7)

for(i in 1:dim(x)[2]){
  special.Vars = colnames(x)[i]
  summaryTableList = rateOfChange(x = x, y = y, noiseLevs = noiseLevs, special.Vars
= special.Vars, iteration = iteration)
  VifLeastSqrsMat[, i] = summaryTableList[[4]][[3]]
  VifRateOfChangeMat[, i] = summaryTableList[[4]][[4]]
}

VifLeastSqrsMat
VifRateOfChangeMat
```

Table 14. Least squares best fit values for VIF

Noise variable	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]
shoulder	-3.141	-1.210	-0.099	-0.128	-0.120	-0.024	-0.169
chest	-1.155	-3.365	-0.798	0.016	0.085	-0.169	0.012
bicep	-0.067	-0.472	-9.583	-5.871	-0.011	0.001	-0.172
forearm	-0.040	0.017	-4.831	-18.026	-10.555	-0.104	-0.043
wrist	-0.011	0.001	-0.006	-2.120	-15.685	-0.052	-0.178
age	-0.002	-0.022	0.000	-0.021	-0.056	-0.047	-0.003
height	-0.033	0.000	-0.070	-0.020	-0.504	-0.002	-0.444

Table 15. Rate of change values for VIF

Noise variable	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]
shoulder	-3.087	-1.163	-0.127	-0.168	-0.154	-0.032	-0.164
chest	-1.133	-3.376	-0.926	-0.079	-0.168	-0.194	-0.035
bicep	-0.074	-0.496	-10.044	-6.160	-0.119	-0.007	-0.189
forearm	-0.130	-0.042	-5.030	-19.080	-10.045	-0.123	-0.051
wrist	-0.014	-0.002	-0.024	-2.214	-14.610	-0.062	-0.181
age	-0.003	-0.025	-0.002	-0.035	-0.064	-0.042	-0.004
height	-0.032	-0.003	-0.080	-0.028	-0.472	-0.010	-0.409

In Table 14 and 15, we can identify the regressors whose best fit and rate of change absolute values are greater than one. These values are highlighted in the tables above. A line with a

slope greater than a magnitude of one will signify that there is a significant difference in the calculated VIFs. We can observe that when the shoulder regressor is the noise variable, the chest regressor is the only regressor with a rate of change and best fit value greater than 1. The same is true when chest is the noise regressor. When bicep is the noise regressor, the forearm regressor changes significantly. However, when the forearm is perturbed, the bicep and the wrist regressors changes significantly. When noise is added to the wrist regressor, only the forearm regressor changes dramatically. We call these types of relationships, coupling relationships. We define that a coupling relationship, or coupling dependency, exists between two or more regressors if significant changes of their best fit slope and rate of change values exists as either one is being perturbed. Thus, the coupling variables for the body dimensions dataset are shoulder and chest, bicep and forearm, and forearm and wrist.

Choosing a magnitude of 1 as a threshold for the rate of change may be unique to the VIF diagnostic and this subset of variables. We chose 1 to perform our analysis because a magnitude greater than 1 for the rate of change or least squares best fit values indicates that the VIF diagnostic changed proportionally more than the noise added to each variable. Other diagnostics may indicate a different threshold value and other datasets might show a significant threshold other than 1 for the VIF diagnostic.

The regressors that have the largest in magnitude rate of change and least squares fit values are the regressors that have the smallest variance. These variables are bicep girth, forearm

girth, and wrist girth. Because forearm's variance is in between biceps and wrist variance, it is coupled with both of them. Age and height are both insignificant variable with respect to the multicollinearity problem. Thus, their insignificant changes in VIFs are not surprising.

IV. DISCUSSION

In this chapter, we'll discuss the application of the *mcperturb* package to a larger dataset and explore a variable selection method. Then, we'll discuss how to fix the multicollinearity problem. Finally, we'll discuss some limitations of the *mcperturb* package.

4.1 Application to a larger dataset

The application of the *mcperturb* package on a bigger dataset is explored by using all 24 continuous variables in the body dimension dataset. Tables 16 and 17 show the least squares best fit and rate of change VIF values for each regressor respectively.

Even with a much larger dataset used for analysis, the *mcpertub* package can identify coupling relationships. Parsing Table 16 and 17, the new coupling relationships among the regressors can be identified. The new coupling regressors are hip girth with thigh girth, knee diameter with ankle diameter, and wrist diameter with wrist girth. Wrist girth with forearm girth and forearm girth with chest girth are still coupling regressors. Shoulder diameter with chest girth are no longer identified as coupling regressors. This may be due to shoulder diameter being related to many of the new regressors included in the larger dataset. Thus, even as chest girth is being perturbed, the other variables relate with shoulder diameter enough to keep its VIF from changing.

R-code for generating Table 16 and 17

```
VifLeastSqrsMat = matrix(NA, nrow = 23, ncol = 23)
VifRateOfChangeMat = matrix(NA, nrow = 23, ncol = 23)
for(i in 1:dim(x)[2]){
  special.Vars = colnames(x)[i]
  summaryTableList = rateOfChange(x = x, y = y, noiseLevs = noiseLevs, special.Vars
= special.Vars, iteration = iteration)
  VifLeastSqrsMat[, i] = summaryTableList[[4]][[3]]
  VifRateOfChangeMat[, i] = summaryTableList[[4]][[4]]
}
VifLeastSqrsMat
VifRateOfChangeMat
```

Table 16. Least squares best fit values for VIF with larger dataset

Noise.Variable	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]
bicromial	-4.448	-0.094	-0.248	-0.016	-0.311	-0.089	-0.136	-0.092	-0.109	-0.108	-0.014
biiliac	-0.025	-2.128	-0.589	-0.013	-0.009	-0.092	-0.051	-0.053	-0.074	-0.013	-0.004
bitrochanteric	-0.157	-0.926	-5.343	-0.048	-0.102	-0.299	-0.030	-0.059	-0.067	-0.013	-0.020
chestDepth	-0.022	-0.016	-0.053	-3.625	-0.140	-0.086	-0.136	-0.059	-0.051	-0.005	-0.103
chestDim	-0.345	-0.022	-0.234	-0.264	-6.434	-0.024	-0.174	-0.083	-0.080	-0.083	-0.287
elbowDim	-0.043	-0.103	-0.370	-0.046	-0.032	-17.669	-1.372	-0.154	-2.030	-0.018	-0.025
wristDim	-0.042	-0.025	-0.024	-0.070	-0.026	-0.686	-16.375	-0.098	-0.366	-0.003	-0.006
kneeDim	-0.039	-0.102	-0.065	-0.043	-0.026	-0.177	-0.178	-8.008	-1.119	-0.008	-0.018
ankleDim	-0.035	-0.085	-0.041	-0.024	-0.022	-1.138	-0.415	-1.128	-9.896	-0.017	-0.016
shoulderDim	-0.911	-0.288	-0.175	-0.188	-0.614	-0.216	-0.102	-0.202	-0.507	-4.100	-0.587
chestGir	-0.195	-0.031	-0.565	-1.897	-2.952	-0.343	-0.195	-0.410	-0.529	-0.790	-6.171
waistGir	-0.122	-0.056	-0.057	-0.763	-0.191	-0.612	-0.114	-0.056	-0.058	-0.010	-0.188
navelGir	-0.078	-0.761	-0.108	-0.132	-0.069	-0.109	-0.157	-0.181	-0.080	-0.020	-0.057
hipGir	-0.129	-0.123	-4.022	-0.126	-0.108	-0.166	-0.352	-0.148	-0.465	-0.100	-0.049
thighGir	-0.042	-0.025	-0.152	-0.145	-0.026	-0.225	-0.280	-0.137	-0.080	-0.023	-0.022
bicepGir	-0.081	-0.034	-0.197	-0.114	-0.395	-0.210	-0.216	-0.154	-0.276	-0.213	-0.300
forearmGir	-0.073	-0.123	-0.221	-0.072	-0.133	-2.513	-0.661	-0.089	-0.212	-0.020	-0.016
kneeGir	-0.044	-0.022	-0.026	-0.011	-0.004	-0.083	-0.195	-1.297	-0.423	-0.009	-0.017
calfGir	-0.043	-0.085	-0.089	-0.025	-0.031	-0.009	-0.080	-0.065	-0.055	-0.008	-0.012
ankleGir	-0.020	-0.100	-0.061	-0.025	-0.012	-0.086	-0.110	-0.223	-0.933	-0.007	-0.017
wistGir	-0.243	-0.178	-0.043	-0.110	-0.166	-0.098	-4.668	-0.386	-0.055	-0.006	-0.017
age	-0.042	-0.005	-0.063	-0.035	-0.009	-0.018	-0.055	-0.024	-0.029	-0.002	-0.004
height	-0.292	-0.183	-0.045	-0.097	-0.028	-0.254	-0.148	-0.095	-0.301	-0.010	-0.004

Table 16. Continued. Least squares best fit values for VIF with larger dataset

[,12]	[,13]	[,14]	[,15]	[,16]	[,17]	[,18]	[,19]	[,20]	[,21]	[,22]	[,23]
-0.019	-0.015	-0.028	-0.017	-0.016	-0.011	-0.041	-0.035	-0.043	-0.342	-0.030	-0.133
-0.002	-0.072	-0.013	-0.004	-0.008	-0.016	-0.017	-0.010	-0.082	-0.087	-0.003	-0.038
-0.016	-0.016	-0.463	-0.043	-0.020	-0.047	-0.020	-0.038	-0.034	-0.035	-0.037	-0.009
-0.072	-0.006	-0.013	-0.037	-0.014	-0.031	-0.014	-0.032	-0.064	-0.098	-0.025	-0.036
-0.024	-0.017	-0.023	-0.026	-0.117	-0.044	-0.037	-0.034	-0.013	-0.113	-0.006	-0.008
-0.061	-0.018	-0.012	-0.101	-0.048	-0.523	-0.037	-0.051	-0.095	-0.081	-0.008	-0.072
-0.009	-0.038	-0.017	-0.044	-0.008	-0.033	-0.031	-0.041	-0.071	-1.686	-0.014	-0.013
-0.004	-0.016	-0.014	-0.021	-0.017	-0.018	-0.550	-0.031	-0.210	-0.113	-0.012	-0.024
-0.004	-0.008	-0.033	-0.021	-0.018	-0.044	-0.179	-0.011	-0.616	-0.026	-0.007	-0.057
-0.046	-0.049	-0.115	-0.083	-0.473	-0.066	-0.095	-0.067	-0.146	-0.171	-0.006	-0.081
-0.362	-0.120	-0.033	-0.038	-0.843	-0.092	-0.102	-0.233	-0.079	-0.620	-0.047	-0.057
-2.610	-0.246	-0.223	-0.439	-0.042	-0.031	-0.081	-0.132	-0.056	-0.246	-0.065	-0.009
-0.169	-1.977	-0.627	-0.037	-0.081	-0.175	-0.131	-0.128	-0.129	-0.072	-0.121	-0.017
-0.181	-0.849	-6.098	-4.005	-0.258	-0.055	-0.141	-0.077	-0.157	-0.122	-0.007	-0.020
-0.125	-0.010	-1.510	-4.441	-0.421	-0.033	-0.336	-0.728	-0.161	-0.417	-0.137	-0.026
-0.016	-0.090	-0.222	-0.872	-12.040	-6.820	-0.521	-0.097	-0.110	-0.258	-0.022	-0.038
-0.052	-0.115	-0.021	-0.085	-5.522	-22.093	-0.384	-0.371	-0.133	-5.115	-0.060	-0.037
-0.022	-0.018	-0.014	-0.152	-0.115	-0.106	-5.410	-0.327	-0.526	-0.029	-0.022	-0.068
-0.032	-0.039	-0.016	-0.329	-0.020	-0.120	-0.322	-4.274	-1.055	-0.058	-0.005	-0.054
-0.002	-0.017	-0.006	-0.038	-0.023	-0.041	-0.310	-0.659	-5.960	-0.776	-0.007	-0.005
-0.019	-0.019	-0.032	-0.187	-0.040	-1.391	-0.026	-0.094	-1.330	-23.78	-0.032	-0.024
-0.011	-0.036	-0.002	-0.079	-0.006	-0.023	-0.002	-0.010	-0.026	-0.038	-0.218	-0.020
-0.004	-0.007	-0.004	-0.029	-0.020	-0.059	-0.170	-0.052	-0.015	-0.127	-0.036	-0.911

Table 17. Rate of change values for VIF with larger dataset

Noise.variable	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]	[,11]
bicromial	-4.675	-0.066	-0.227	0.003	-0.268	0.004	0.068	-0.075	-0.073	-0.094	-0.006
biiliac	-0.011	-2.449	-0.584	0.007	0.002	-0.074	0.029	-0.048	-0.013	-0.011	0.002
bitrochanteric	-0.162	-0.992	-5.789	0.029	-0.106	-0.327	-0.003	0.016	0.014	-0.009	-0.011
chestDepth	0.006	0.008	0.007	-4.011	-0.128	-0.043	-0.113	0.008	0.034	-0.001	-0.093
chestDim	-0.302	0.001	-0.154	-0.248	-6.995	0.000	0.026	-0.020	0.043	-0.054	-0.239
elbowDim	0.005	-0.097	-0.334	-0.037	0.023	-18.545	-1.087	-0.108	-2.016	-0.001	0.003
wristDim	-0.002	0.017	-0.016	-0.053	-0.002	-0.683	-17.565	-0.071	-0.324	0.000	-0.002
kneeDim	-0.021	-0.046	-0.030	-0.007	0.001	-0.077	-0.099	-8.770	-1.125	-0.003	-0.012
ankleDim	-0.024	-0.026	0.023	0.011	0.015	-1.141	-0.415	-1.067	-10.728	-0.011	-0.010
shoulderDim	-0.860	-0.322	-0.043	0.042	-0.674	-0.168	0.020	0.095	-0.083	-3.915	-0.480
chestGir	-0.106	0.025	-0.395	-1.897	-2.907	-0.071	-0.080	-0.399	-0.137	-0.778	-5.602
waistGir	-0.090	0.041	-0.010	-0.825	-0.155	-0.491	-0.045	-0.026	0.008	-0.005	-0.134
navelGir	-0.043	-0.823	-0.070	0.029	-0.030	-0.095	-0.128	-0.128	-0.042	-0.018	-0.042
hipGir	-0.105	-0.081	-4.204	-0.024	-0.027	0.073	-0.282	-0.149	-0.197	-0.071	0.016
thighGir	-0.001	0.005	-0.146	-0.067	-0.016	-0.086	-0.134	0.005	0.003	-0.018	0.004
bicepGir	-0.005	0.012	-0.039	0.018	-0.297	0.132	0.048	-0.079	-0.020	-0.217	-0.298
forearmGir	0.046	0.006	-0.118	0.033	-0.023	-2.611	0.016	-0.004	0.117	0.007	-0.005
kneeGir	-0.034	-0.014	-0.005	0.007	0.003	0.009	0.076	-1.317	-0.408	0.004	0.000
calfGir	0.024	-0.004	-0.024	-0.010	0.000	0.006	-0.028	-0.046	0.001	-0.005	-0.006
ankleGir	-0.001	-0.057	0.028	0.001	0.003	-0.040	-0.074	-0.200	-0.997	0.001	0.003
wistGir	-0.210	-0.135	0.022	-0.024	-0.098	-0.013	-4.649	-0.376	0.008	0.002	-0.001
age	-0.020	0.002	-0.049	-0.022	0.002	0.010	0.011	-0.008	-0.019	0.001	0.001
height	-0.340	-0.207	-0.017	-0.104	0.007	-0.265	0.040	-0.075	-0.265	-0.008	-0.002

Table 17. Continued. Rate of change values for VIF with larger dataset

[,12]	[,13]	[,14]	[,15]	[,16]	[,17]	[,18]	[,19]	[,20]	[,21]	[,22]	[,23]
-0.010	-0.011	-0.023	-0.008	-0.005	0.001	-0.027	0.008	-0.012	-0.262	-0.022	-0.136
0.000	-0.068	-0.010	0.000	0.004	-0.009	-0.003	-0.005	-0.060	-0.080	0.002	-0.032
-0.001	-0.012	-0.491	-0.042	-0.007	-0.039	-0.009	-0.030	0.020	0.005	-0.031	-0.003
-0.068	-0.003	-0.011	-0.018	-0.001	0.006	0.007	-0.013	0.034	0.026	-0.022	-0.029
-0.018	0.001	0.007	-0.008	-0.103	-0.009	0.021	-0.008	0.003	-0.049	-0.002	0.002
-0.060	-0.016	0.004	-0.055	0.022	-0.536	0.019	0.019	-0.043	-0.025	0.002	-0.076
0.001	-0.017	-0.014	-0.030	0.003	-0.001	0.004	0.009	-0.057	-1.596	0.004	-0.002
0.001	-0.015	-0.007	0.002	-0.012	0.002	-0.517	-0.034	-0.139	-0.067	0.005	-0.017
0.000	-0.005	-0.023	0.002	0.004	0.004	-0.178	0.002	-0.654	0.003	-0.003	-0.058
0.005	-0.040	-0.106	-0.051	-0.441	-0.005	-0.010	-0.001	-0.015	0.011	0.003	-0.048
-0.335	-0.068	0.001	0.018	-0.789	-0.005	-0.065	-0.120	0.016	-0.199	-0.029	-0.010
-2.671	-0.241	-0.204	-0.466	0.005	-0.016	-0.080	-0.121	-0.012	-0.127	-0.053	0.001
-0.160	-2.062	-0.588	-0.011	-0.050	-0.123	-0.031	-0.120	-0.127	0.029	-0.139	0.003
-0.137	-0.871	-6.002	-4.159	-0.231	-0.010	0.038	-0.067	0.044	-0.028	0.003	0.002
-0.119	-0.004	-1.448	-4.627	-0.398	-0.004	-0.218	-0.745	-0.059	-0.388	-0.121	-0.022
0.000	-0.058	-0.233	-0.780	-11.417	-6.427	-0.523	-0.048	0.060	-0.164	0.007	0.007
0.012	-0.084	0.002	0.020	-4.962	-20.885	-0.398	-0.210	0.011	-5.293	-0.039	-0.011
-0.006	-0.011	-0.005	-0.145	-0.106	-0.096	-5.730	-0.293	-0.518	-0.023	0.008	-0.068
-0.024	-0.014	-0.012	-0.326	-0.010	-0.079	-0.341	-4.802	-1.060	-0.013	0.000	-0.015
0.000	-0.010	-0.002	-0.006	0.004	-0.011	-0.334	-0.724	-6.471	-0.706	-0.008	0.002
-0.005	0.011	0.002	-0.151	-0.020	-1.291	-0.005	0.012	-1.438	-24.45	-0.024	-0.002
-0.012	-0.037	0.001	-0.073	0.001	-0.013	-0.001	0.000	-0.013	-0.031	-0.236	-0.023
0.001	-0.005	-0.001	-0.016	-0.004	-0.017	-0.125	-0.041	0.005	0.036	-0.034	-0.974

4.2 Comparison to a variable selection method

Variable selection for MLR can be used to select the regressors that provide significant information to the MLR model (Mertler & Reinhart, 2016). A stepwise variable selection method uses a model's Akaike Information Criteria (AIC) measure to estimate the quality of each model when different regressors are included in different models. The stepwise procedure terminates once the calculated AIC value for a new model is no longer smaller than the existing AIC value. A possible sign that a multicollinearity problem exists in an MLR model is when practically significant regressors are deemed statistically insignificant. This was discussed in chapter 3 section 4 with the wrist minimum girth regressor. This statistical insignificance may occur, which depends on how inflated the regressor standard error is. Inflated standard errors can indicate redundancy between regressors. Comparing the regressors deemed insignificant by a stepwise variable selection method with the regressors identified as causing a multicollinearity problem by the *mcperturb* package may provide a more thorough diagnosis for fixing a multicollinearity problem. The summary results of the forward stepwise variable selection method are shown in Table 18. The forward and backwards selection methods conclude the same results.

```
Performing stepwise variable selection procedure
fullMat = cbind(x, y)
null = lm(y ~ 1, data = fullMat)
fullMod = lm(y ~., data = fullMat)
step(null, scope = list(lower = null, upper = fullMod), direction = "forward")
```

Table 18. Summary table of the forward selection method

```
Call:
lm(formula = y ~ chest + height + bicep + forearm, data = x)

Residuals:
    Min      1Q  Median      3Q     Max
-15.842  -3.049  -0.255   2.644  22.875

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -80.38607    4.15716 -19.337  < 2e-16 ***
chest         0.67128    0.05499  12.207  < 2e-16 ***
height        0.33166    0.03183  10.418  < 2e-16 ***
bicep         0.49943    0.17891   2.791  0.00545 **
forearm       0.56092    0.25425   2.206  0.02782 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 4.972 on 502 degrees of freedom
Multiple R-squared:  0.8623,   Adjusted R-squared:  0.8612
F-statistic: 785.9 on 4 and 502 DF,  p-value: < 2.2e-16
```

The regressors not included in the variable selection result are shoulder diameter, wrist minimum girth, and age. The shoulder diameter regressor may be left out of the variable selection method because of its coupling relationship with chest girth. Because of this coupling relationship, shoulder diameter provides the MLR model with a lot of the same or similar information as chest girth. Because chest girth has a higher correlation with the response variable weight, shoulder diameter may be deemed insignificant by the variable selection method. Although age is not a variable causing a multicollinearity problem, the regressor is left out of the resulting variable selection due to lack of significance. The wrist minimum girth variable may have been left out of the selected model because of the coupling effect it has with forearm. Thus, wrist may be insignificant because it provides the MLR model with the same type of information as forearm and forearm has a higher correlation with the response variable.

It may be possible to use multicollinearity diagnostic measures as threshold criteria for a variable selection method. Systematically removing the less significant coupling regressors could be used as a strategy for the dimension reduction in regression analysis and will be explored in future research.

4.3 Dealing with multicollinearity

Although some researchers might argue that nothing should be done to fix the multicollinearity problem, there exist many solutions. These solutions consist of but are not limited to (Montgomery et al., 2012; Imdadullah et al, 2016),

a. Collecting additional observations
b. Deleting variable(s) that may cause the problem
c. Combining variable(s) that may cause the problem
d. Transforming variables
e. Performing principal component regression (PCR) or ridge regression

It has been suggested that the best method for combating multicollinearity is collecting additional observations in a way that addresses the multicollinearity problem (Montgomery et al., 2012). Although this method may be considered the best method, it is may be difficult to achieve. Deleting the variables that cause a multicollinearity problem can be executed by performing a variable selection method (Mertler & Reinhart, 2016). From the analysis covered in this thesis, we may identify which regressors are left out of the selected model because of insignificance or because of a coupling effect with another regressor.

Combining variables has been suggested to combat the multicollinearity problem (Hocking, 2013; Mertler & Reinhart, 2016) . If regressors are highly correlated with each other, they can be combined into one regressor. If a regressor is highly correlated with multiple regressors, then finding its coupling regressors can be used as reference to select which variables to combine. Variable transformation can be performed for multiple reasons and will inherently change the distribution of the regressor. With respect to multicollinearity, it has been suggested to perform a mean-center transformation when a higher ordered term is included in the model (Hocking, 2013; Iacobucci, et al., 2016). Finally, performing PCR combats multicollinearity by using less than the full set of principal components in the model. Because principal components are uncorrelated, there will be no issue with multicollinearity if PCR regression is performed (Jolliffe, 2002; Hocking, 2013). Ridge regression is designed to provide more stable parameter estimates by shrinking the least squares estimators. This will lead to the ridge estimators having less variance than the least squares estimators (Hocking, 2013; Firinguetti et al, 2017). Here we only list a few ways of dealing with multicollinearity, identifying which strategy is the best for alleviating a multicollinearity problem can be explored in future work.

4.4 Limitations

Although the *mcperturb* package accomplishes dynamic multicollinearity diagnostic analysis, some limitations about the perturbation analysis should be discussed. The existing *perturb* package can be performed with the inclusion of categorical variables and with the application of randomly generation noise from a uniform distribution. However, the current

mcperturb package can only conduct the perturbation analysis with randomly generated noise from a normal distribution. Implementing categorical variables into the *mcpertub* package will be explored in future research. A weakness of performing perturbation analysis before calculating the overall diagnostics is that small perturbations applied to each regressor may not significantly change the diagnostic measure. That is, there may not be a noticeable difference between the diagnostic measure before and perturbation analysis. From Table 11 we can rank the regressors by their influence on the determinant. However, because the determinant is initially small and close to zero, the differences will be relatively small and close to zero. Thus, the ranking may not be very helpful. Another limitation of *mcperturb* package is that the cutoff values for identifying coupling relationships are left open for interpretation. Therefore, developing robust rate of change cut off values for all the individual diagnostics can be explored in future work.

V. CONCLUSION

Multicollinearity is a complex mathematical problem that should be addressed in order to have accurate statistical inference when using MLR models. The main contribution of this thesis is to improve multicollinearity detection methods by developing the new package, *mcperturb*. Advancing the diagnosis of multicollinearity by combining perturbation analysis with the calculation of diagnostic measures is a new contribution of the *mcperturb* package. Using observational analysis, the *mcperturb* package helps identify the regressors that may be causing a multicollinearity problem. Applying perturbation analysis to the overall multicollinearity diagnostic measures will provide evidence for ranking the influence that each regressor has on the overall model. Applying perturbation analysis to the individual multicollinearity diagnostic measures may help identify coupling relationships between regressors. These analyses can provide the analyst with the opportunity to further diagnose a multicollinearity problem, which allows for better ways to alleviate the multicollinearity problem and have more accurate statistical inference.

REFERENCES

Baird, G. L., & Bieber, S. L. (2016). The Goldilocks dilemma: Impacts of multicollinearity--a comparison of simple linear regression, multiple regression, and ordered variable regression models. *Journal of Modern Applied Statistical Methods*, *15*(1), 18.

Belsley, D. A. (1991). A guide to using the collinearity diagnostics. *Computer Science in Economics and Management*, *4*(1), 33-50.

Belsley, D. A., Kuh, E. & Welsch, R. E. (1980). *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. New York: John Wiley & Sons.

Curto, J. D., & Pinto, J. C. (2007). New multicollinearity indicators in linear regression models. *International Statistical Review*, *75*(1), 114-121.

Disatnik, D., & Sivan, L. (2016). The multicollinearity illusion in moderated regression analysis. *Marketing Letters*, *27*(2), 403-408.

Firinguetti, L., Kibria, G., & Araya, R. (2017). Study of partial least squares and ridge regression methods. *Communications in Statistics-Simulation and Computation*, *46*(8), 6631-6644.

Grete, H., Louis, J. P., Roger, W. J., & Carter, J. K. (2003). Exploring relationships in body dimensions. *J. Statist. Educ*, *11*.

Hendrickx, J. (2012). perturb: Tools for evaluating collinearity. *R package version*, *2*.

Hocking, R. R. (2013). *Methods and applications of linear models: regression and the analysis of variance*. John Wiley & Sons.

Iacobucci, D., Schneider, M. J., Popovich, D. L., & Bakamitsos, G. A. (2016). Mean centering helps alleviate "micro" but not "macro" multicollinearity. *Behavior research methods*, *48*(4), 1308-1317.

Imdadullah, M., Aslam, M., & Altaf, S. (2016). Mctest: An R package for detection of collinearity among regressors. *The R Journal*, *8*(2), 495-505.

Jolliffe, I. T. (2002). Springer series in statistics. *Principal component analysis*, *29*.

Kleinbaum, D., et al. (2007) *Applied regression analysis and other multivariable methods*. Vol. 4. Cengage Learning.

Kovács, P., Petres, T., & Tóth, L. (2005). A new measure of multicollinearity in linear regression models. *International Statistical Review, 73*(3), 405-412.

Mertler, C. A., & Reinhart, R. V. (2016). *Advanced and multivariate statistical methods: Practical application and interpretation*. Routledge.

Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). *Introduction to linear regression analysis* (Vol. 821). John Wiley & Sons.

O'Hagan, J., & McCabe, B. (1975). Tests for the severity of multicollinearity in regression analysis: A comment. *The Review of Economics and Statistics*, 368-370.

Ott, R. L., & Longnecker, M. T. (2015). *An introduction to statistical methods and data analysis*. Nelson Education.