

**DEVELOPING A COST-EFFECTIVE COMPUTING MODEL FOR
OPTIMAL DIETS FOR PEOPLE LIVING WITH HIV**

Yasin Kaunda Kowa

**A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree
of Master's in Mathematical and Computer Sciences and Engineering of the Nelson
Mandela African Institution of Science and Technology**

Arusha, Tanzania

April, 2016

ABSTRACT

People living with HIV (PLWHIV) without proper treatment are vulnerable to many kinds of opportunistic infections due to their weaker immune systems than healthy people. Poor nutrition intensifies the progression of HIV into AIDS by further compromising the immune system. Therefore, achieving basic nutritional recommendations is important at all stages of the disease. However, economic limitation (poverty) and lack of knowledge to find adequate amounts with right combinations of different locally available foods hinders them to meet the recommended daily nutrients requirements, leading them to become weak in a very short time and even experience early mortality. In this research, I developed a mathematical model and extended it to a MATLAB based graphical user interface (GUI) that could be used as a computation tool to compute adequate amounts of available foods to achieve the recommended nutrients at a minimum cost compared to an alternative. The mathematical model is the combination of multiple linear regression models and a linear programming model. Multiple linear regression models use the factors of age, weight, height, and gender to predict the nutritional requirements in the body. The results from the multiple linear regression model were used to define the constraints in the linear programming model. The linear programming model was used to compute the adequate amounts of foods that would lead to the achievement of the recommended nutrients taking into consideration practical biological/physical and economic constraints.

The Graphical User Interface (GUI) was developed by the Graphical User Interface Development Environment (GUIDE) method in MATLAB. With the incorporated mathematical models it could be used to compute the adequate amount of foods. The GUI has five parts: the first part contains list of foods the user needs to select, the second part is to enter user's particulars of age, weight, height and gender. The third part is to enter the cost of each selected foods. The fourth part is the computation part, which will initiate the computation. There is a status box, which shows whether the food combinations and financial constraints produce an optimal or non-optimal output and a reset button to enable clearance of previous computations and allowance of new data entrances. The last part is the output section which displays the amounts of foods to be bought and the total cost to be incurred when the computation is optimal. Results show that the multiple linear regression model has high predictive power by suggesting values that are close to the recommended daily/dietary intake (RDI). This was validated by testing the mean difference between paired samples using a t-test. By this analysis we found that there was no statistical

difference between the means as the p – values were greater than the significance level of 0.05. The cost for optimal diets was less when model predicted values are used to limits the constraints in linear programming compared to when RDI values are used. The GUI developed could serve as the computation tool to compute adequate amount of foods to meet the recommended nutrients at minimum costs.

DECLARATION

I, Yasin Kaunda Kowa do hereby declare to the Senate of Nelson Mandela African Institution of Science and Technology that this dissertation is my own original work and that it has neither been submitted nor being concurrently submitted for degree award in any other institution.

Yasin Kaunda Kowa



16/04/2016

Name and signature of candidate

Date

The above declaration is confirmed

Dr. Yaw Nkansah-Gyekye

Name and signature of supervisor 1

Date

Dr. Emmanuel Mpolya



Name and signature of supervisor 2

20/04/2016

Date

COPYRIGHT

This dissertation is copyright material protected under the Berne Convention, the Copyright Act of 1999 and other international and national enactments, in that behalf, on intellectual property. It must not be reproduced by any means, in full or in part, except for short extracts in fair dealing; for researcher, private study, critical scholarly review or discourse with an acknowledgement, without a written permission of the Deputy Vice Chancellor for Academic, Research and Innovation, on behalf of both the author and the Nelson Mandela African Institution of Science and Technology.

© Yasin Kaunda Kowa

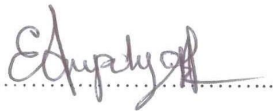
CERTIFICATION

The undersigned certify that have read and found the dissertation acceptable by the Nelson Mandela African Institution of Science and Technology.

.....
Dr. Yaw Nkansah-Gyekye

.....
Date

(1st Supervisor)

.....


Dr. Emmanuel Mpolya

(2nd Supervisor)

.....
20/04/2016

.....
Date

ACKNOWLEDGEMENT

First and foremost, I would like to thank the Almighty God for giving me life and everything else that he has granted to me. God is the source of my strength, faith and health that enabled me to complete this study successfully. Secondly, I am indebted to many individuals and groups of people and the Institution for their moral and material support towards the completion of this study. My sincere thanks go to my supervisors, Dr. Yaw Nkansah-Gyekye (NM-AIST) and Dr. Emmanuel Mpolya (NM-AIST), whose constructive ideas and mentorship has turned my mind into this work, may our almighty God bless you plentifully exceptionally. I would also, uniquely, like to express my deep thankfulness to NM-AIST for offering me a scholarship for my studies and a place to stay at NM-AIST. In fact, without such a support this research would not exist. I would also like to register my special appreciation to my family for their prayers, love and encouragement that they expressed to me while I was away from them for the whole period of the studies.

Since it is not possible to mention everybody who has contributed to this work, allow me to express my heartfelt appreciation to everybody who has contributed in one way or the other towards the success of this work.

TABLE OF CONTENTS

ABSTRACT.....	i
DECLARATION	iii
COPYRIGHT.....	iv
CERTIFICATION	v
ACKNOWLEDGEMENT	vi
TABLE OF CONTENTS.....	vii
LIST OF TABLES	ix
LIST OF FIGURES	x
LIST OF APPENDICES.....	xi
LIST OF ABBREVIATIONS.....	xiii
CHAPTER ONE	1
Introduction.....	1
1.1. Background	1
1.2. Problem Statement	3
1.3. Research Justification.....	4
1.4. General Objective.....	4
1.4.1. Specific Objectives	4
1.5. Research Questions	4
1.6. Dissertation Outline.....	5
CHAPTER TWO	6
Modeling the Diet Decisions for People Living with HIV in Consideration of Age, Weight, Height, and Gender Constraints.....	6
2.1. Introduction	7
2.2. Research Methodology.....	9
2.3. Mathematical Model Formulation.....	9
2.3.1. Multiple Linear Regression Model.....	10
2.3.2. Linear Programming Model	13
2.4. Results	13
2.4.1. Multiple Linear Regression Model Predictive Power	13
2.4.2. Linear Programming Model Results.....	18
2.5. Discussion and Conclusion	22

CHAPTER THREE	24
A MATLAB Based Graphical User Interface for Computing Optimal Diet of People Living with HIV at Asymptomatic Stage	24
3.1. Introduction	24
3.2. Research Methodology	25
3.3. Mathematical Model Formulation.....	26
3.3.1. Regression Analysis	26
3.3.2. Linear Programming Model	29
3.4. Graphical User Interface Development.....	29
3.4.1. Illustration.....	30
3.5. Discussion and Conclusion	31
CHAPTER FOUR.....	35
General Discussion, Conclusion and Recommendations.....	35
4.1. General Discussion.....	35
4.2. Conclusion.....	36
4.3. Recommendations	36
References.....	37
Appendices.....	40

LIST OF TABLES

Table 2.1: Comparison of model predicted values to the actual RDI values.....	14
Table2.2: Paired Samples Statistics	16
Table2.3:Paired Samples Correlations.....	16
Table2.4:Paired Samples Test.....	17
Table 2.5: Cost comparison incurred for computing adequate amount of foods to reach the optimal nutrients using our model predicted values and using RDI values	18

LIST OF FIGURES

Figure 2.1: Cost comparison when RDI values and model predicted values were used to compute adequate amount of foods to achieve recommended nutrients	21
Figure 3.1: Graphical user interface for computing optimal diet.....	32
Figure 3.2: Illustration of GUI computation for optimal diet	33
Figure 3.3: Illustration of GUI computation for not optimal diet	34

LIST OF APPENDICES

Appendix 1: summary of multiple linear regression analysis results in graphs and tables for energy	40
Appendix 2: summary of multiple linear regression analysis results in graphs and tables for protein	42
Appendix 3: summary of multiple linear regression analysis results in graphs and tables for fiber	44
Appendix 4: summary of multiple linear regression analysis results in graphs and tables for vitamin A.....	46
Appendix 5: summary of multiple linear regression analysis results in graphs and tables for vitamin E	48
Appendix 6: summary of multiple linear regression analysis results in graphs and tables for vitamin C.....	50
Appendix 7: summary of multiple linear regression analysis results in graphs and tables for Thiamin	52
Appendix 8: summary of multiple linear regression analysis results in graphs and tables for Riboflavin	54
Appendix 9: summary of multiple linear regression analysis results in graphs and tables for Niacin	56
Appendix 10: summary of multiple linear regression analysis results in graphs and tables for Vitamin B6.....	58
Appendix 11: summary of multiple linear regression analysis results in graphs and tables for Folate	60
Appendix 12: summary of multiple linear regression analysis results in graphs and tables for Vitamin B12.....	62
Appendix 13: summary of multiple linear regression analysis results in graphs and tables for Pantothenic.....	64
Appendix 14: summary of multiple linear regression analysis results in graphs and tables for Phosphorus.....	66
Appendix 15: summary of multiple linear regression analysis results in graphs and tables for potassium	68
Appendix 16: summary of multiple linear regression analysis results in graphs and tables for Sodium	70

Appendix 17: summary of multiple linear regression analysis results in graphs and tables for Zinc	72
Appendix 18: summary of multiple linear regression analysis results in graphs and tables for Copper.....	74
Appendix 19: summary of multiple linear regression analysis results in graphs and tables for Manganese	76
Appendix 20: summary of multiple linear regression analysis results in graphs and tables for Upper vitamin E.....	78
Appendix 21: summary of multiple linear regression analysis results in graphs and tables for Upper niacin.....	80
Appendix 22: summary of multiple linear regression analysis results in graphs and tables for Upper vitamin B6.....	82
Appendix 23: summary of multiple linear regression analysis results in graphs and tables for Upper folate	84
Appendix 24: summary of multiple linear regression analysis results in graphs and tables for Upper sodium.....	86
Appendix 25: summary of multiple linear regression analysis results in graphs and tables for upper zinc.....	88
Appendix 26: summary of multiple linear regression analysis results in graphs and tables for upper copper.....	90
Appendix 27: MATLAB code for the function TwoPhaseSimplex	92
Appendix 28: MATLAB code for the Graphical User Interface	94

LIST OF ABBREVIATIONS

AIDS - Acquired Immune Deficiency Syndrome

ARV - Anti-Retroviral

Dr - Doctor

FAO - Food and Agriculture Organization

GUI - Graphical User Interface

GUIDE - Graphical User Interface Development Environment

HAART - Highly Active Anti-Retroviral Therapy

HIV - Human Immunodeficiency Virus

NM - AIST- Nelson Mandela African Institution of Science and Technology

MODEL24.51- Model predicted values for a man aged 24.5 years

MODEL60.50- Model predicted values for a woman aged 60.5 years

PLWHIV- People Living With HIV

RDI - Recommended Dietary Intake

RDI24.51- Recommended Dietary Intake for a man aged 24.5 years

RDI60.50- Recommended Dietary Intake for a woman aged 60.5 years

TFNC - Tanzania Food and Nutrition Center

WHO - World Health Organization

CHAPTER ONE

Introduction

This chapter describes the general introduction of the study, which includes: introduction, background, problem statement, research justification, objectives and research questions.

1.1. Background

Human immunodeficiency virus (HIV) is the virus that affects the immune system, particularly cells responsible for body's immune system called CD4+T. The virus impairs the immune function of these cells making the infected person weak and easily infected by opportunistic pathogens (Mbabazi, 2008). According to the World Health Organization (WHO) (2014) there were approximately 34 million people worldwide living with HIV/AIDS in the 2014, the majority of people living with HIV/AIDS are in the low and middle income countries. Sub-Sahara African region is the most affected one with about 25.8 [24.0–28.7] million people in 2014 living with HIV. HIV is still the world's leading infectious killer since the first case, about 39 million people have died of HIV related causes until 2013, and about (1.0-1.5) million people died in 2014 worldwide.

According to the Tanzania Commission for AIDS (TACAIDS), the Zanzibar AIDS Commission (ZAC), the National Bureau of Statistics (NBS), the Office of the Chief Government Statistician (OCGS), and the ICF International (2013), about 5.1% of Tanzanians aged 15-49 years are HIV positive, with ten regions having HIV/AIDS infections above 5% as follows: Pwani 5.9%, Dar Es Salaam 6.9%, Ruvuma 7.0%, Iringa 9.1%, Mbeya 9.0%, Tabora 5.1%, Rukwa 6.2%, Shinyanga 7.4%, Njombe 14.8% and Katavi 5.9%.

HIV affects the immune system making people infected with the virus to be very susceptible to many kinds of other infections. It is important that they keep their bodies as healthy as possible by making sure that they get all the nutrients a person needs (Clark *et al.*, 2012).

For people living with HIV (PLWHIV), poor nutrition worsens the effects of HIV by further weakening the immune system, and hence damaging people's ability to resist and fight infections. This may lead to a quicker progression of HIV into AIDS disease. Moreover, HIV interferes with the ability to access, control, prepare, eat and use food, and thus increase the risk of malnutrition

among people living with HIV/AIDS. Food and nutritional intake can affect adherence to antiretroviral drugs (ARVs) as well as their effectiveness (WHO, 2007; Fawzi and Hunter, 1998). According to Duran *et al.* (2008), achieving basic nutritional recommendations is an important issue when treating people living with HIV/AIDS at all stages of the disease. The effects of malnutrition (especially undernutrition) and HIV are synergistic as they amplify their individual deleterious effects. Malnutrition exaggerates the effects of HIV on the individual by promoting fatigue and disease propagation, resulting in increased morbidity and earlier death (Colecraft, 2008).

Lack of sufficient food intake and or malabsorption leads to weight loss, which further exacerbates the catabolic nature of HIV infection (Ivers *et al.*, 2009). A healthy and balanced diet helps in maintaining balanced body weight and fitness. Eating well helps to maintain and improve the performance of the immune system for the body's protection against infection and therefore helps a person to stay healthy (Akram *et al.*, 2003).

Dietary components have the potential to reduce ARVs side effects and specific dietary intake assists the absorption and metabolism of ARVs (Dora *et al.*, 2007). A synergistic relationship between immune dysfunction and malnutrition has long been recognized, suggesting that inadequate nutritional status may be an important cofactor of HIV disease progression (Baum *et al.*, 1997).

Dieting is the practice or habit of eating and drinking in a controlled manner with the aim of losing or sometimes gaining weight, or in some cases to regulate the amounts of certain nutrients. Good nutrition (dieting) plays a central role in maintaining and improving immune capability and performance in the human being which protects the body against bacteria, viruses and other organisms which cause diseases (Toumasis, 2004; Langseth, 1999). Recommended Dietary Intake (RDI) is the average daily dietary intake level that is sufficient to meet the nutrient requirements of nearly all (97–98 %) healthy individuals in a particular life stage and gender group (Capra, 2006).

Linear programming is a mathematical model that minimizes a linear function of a set of variables with regard to multiple linear constraints on these variables. It is used to minimize the cost of a diet while satisfying constraints introduced to ensure a pleasant, nutritionally adequate diet (Briend *et al.*, 2001). The study by Darmon *et al.* (2002) describes the use of linear programming as a

technique to design nutrient adequate diets to reach the nutritional recommendations constraints of optimal nutrient density. A linear programming optimization model has been employed to model the diet for Malaysian boarding school caterers with the aim of minimizing the budget provided by the Malaysian government and to maximize the variety of food intake (Sufahani and Ismail, 2014). Also, a linear programming optimization model has been used to model food consumption to meet daily standard World Health Organization (WHO) nutrients needed by the standard woman and the standard man at the minimal cost, the cost of foods to be minimized is defined as the objective function in the optimization model (Pasec *et al.*, 2011).

1.2. Problem Statement

The extent of the existence of HIV infection and its effects as well as the importance of good nutrition to people living with HIV have been elaborated in the preceding literature. However, economic limitation (poverty) and lack of knowledge to find adequate amounts and the right combination of different locally available foods makes it difficult for people living with HIV to have a balanced diet that can help to keep their bodies healthier, stronger and respond positively to ARVs. As a result, they become weakened in a short time and even experience early deaths. This has negative social and economic impacts to the individual, family, community and the nation at large.

Computationally, this problem can be tackled by a linear programming model. This model is used to find the optimal nutrients from a combination of different available foods at minimum cost. In this research, multiple linear regression models were developed to predict the nutrient requirements in the body by considering factors such as individual's weight, age, gender and height. Consideration of these factors was done so as to avoid over or undertaking the nutrients.

This study combined linear programming and multiple linear regression models to develop a cost-effective computing model for optimal diet to the people living with HIV. The mathematical model developed were incorporated in a MATLAB-based graphical user interface that could be used as a tool to compute the adequate amount of foods one should have to buy/take at a minimum cost while meeting all the nutritional requirements.

1.3. Research Justification

The results of this study would be useful to health professionals such as nurses, clinical officers and physicians in advising people living with HIV about the amounts of foods they should take at a minimum cost while meeting all the necessary nutrients recommended considering an individual's age, weight, height and gender. The research will add new knowledge to the community on how to get the right amount of foods so as to meet the recommended nutrients to the people living with HIV at minimum cost. Moreover, through the development of a mathematical model and a graphical user interface (GUI) from this study people living with HIV will be able to compute adequate amount of foods that would be available at a lower cost so as to reach optimal nutrients, and this would help the patients to manage their health better by meeting the recommended nutritional requirements and continue having an important contribution to social and economic sectors at the household and national level.

1.4. General Objective

The main objective of this study was to develop a cost-effective computing model for optimal diets for people living with HIV.

1.4.1. Specific Objectives

The specific objectives were:

1. To formulate a mathematical model for computing an optimal diet required by people living with HIV.
2. To analyze the multiple linear regression model.
3. To compare the cost when RDI, and multiple linear regression results were used to define the constraints in linear programming model.
4. To develop and implement a graphical user interface in MATLAB for a cost-effective computing model for the optimal diet for people living with HIV.

1.5. Research Questions

The research questions were developed based on the specific objectives; thus the research questions were:

1. What mathematical models could predict nutrients required in the human body by the factors of age, weight, height and gender and could be used to compute adequate amount of food that would achieve the amount of nutrients recommended for people living with HIV?
2. Is any difference between the means of model predicted values and RDI values?
3. How would the cost differ when RDI and model predicted values were used to limit the constraints in linear programming model?
4. Could I develop a graphic user interface with MATLAB that would be used to compute adequate amount of foods in order to achieve the amount of nutrients recommended for people living with HIV?

1.6. Dissertation Outline

This dissertation consists of four chapters.

Chapter 1 is about general introduction which consist of introduction, background, problem statement, research justification, research objectives and research questions as described above.

Chapter 2 is about modeling the diet decisions for people living with HIV in consideration of age, weight, height, and gender constraints. This chapter describes the formulation of the mathematical model, the predictive power of the multiple linear regression model developed, and the cost comparison incurred when the values of the multiple linear regression model developed and actual RDI values are used to limit the constraints in linear programming model in order to compute adequate amount of foods.

In Chapter 3 a MATLAB based Graphical User Interface was developed as a computational tool that could enable clients to compute adequate amounts of foods and lead them to achieve the amount of nutrients required at much less cost depending on the individual's age, weight, height, and gender. The developed mathematical models in Chapter 2 were incorporated into the GUI to enable the computation to take place.

Finally, Chapter 4 concludes the dissertation with a general discussion of the research findings, conclusion, and suggestions for further work on the research topic.

CHAPTER TWO

Modeling the Diet Decisions for People Living with HIV in Consideration of Age, Weight, Height, and Gender Constraints¹

Abstract: People living with HIV as well as AIDS patients, who do not receive proper and timely medical treatment, are open targets for all kinds of other infections owing mainly to their relatively weak immune systems. We emphasize upon the fact that, in most (if not all) such cases, poor nutrition intensifies the progression of the disease and that achieving basic nutritional recommendations is important at all stages of the disease. This chapter aims to develop a cost-effective computing model (mathematical model) in diet decisions for people living with HIV in consideration of age, weight, height, and gender constraints. The consideration of these factors tends to avoid undertaking/overtaking of the nutrients which may lead to more serious problems. This model combines multiple linear regression model and linear programming model. The multiple linear regression model predicts the nutrient requirements in the human body basing on the following factors: age, weight, height, and gender. The multiple linear regression model gives out the maximum allowable amount of nutrients (upper bound) and minimum amount of nutrients required (lower bound). These results are used to restrict some constraints in the linear programming model, while others are restricted to the maximum allowable amount of foods. From the linear programming model adequate amount of foods that achieve the nutrients recommended are computed. The linear programming problem formulated is solved by the two phase simplex method in MATLAB. Results show that multiple linear regression predicted values are close enough to the actual recommended dietary/daily intake values. The optimal nutrients are reached at much less cost when the multiple linear regression predicted values are used as nutrient recommendations to restrict the constraints in linear programming model compared to when actual recommended dietary/daily intake values are used. Since our model gives adequate amount of foods at much less cost than when the actual values are used, then this justifies that our goal has been successfully reached. The mathematical model developed could potentially be extended to

¹ This chapter is based on the research paper titled:

Yasin Kaunda Kowa, Yaw Nkansah-Gyekye, Emmanuel Mpolya. "Modeling Diet Decisions for People Living with HIV in Consideration of Age, Weight, Height, and Gender Constraints". *Applied Mathematics*, Vol. 5 No. 4, 2015, pp. 77-83. doi: 10.5923/j.am.20150504.01.

different groups of people who must manage their diets and therefore promises to have a wider applicability.

Keywords: regression analysis, linear programming, optimization, nutrition, people living with HIV

2.1. Introduction

HIV affects the immune system, making people infected with the virus at a higher risk of being infected by several other pathogens. It is important that those infected keep their bodies as healthy as possible by making sure that they get all the required nutrients. This will help to maintain and improve the performance of the immune system and hence increase the ability to fight against pathogens (Clark *et al.*, 2012). Poor nutrition worsens the effects of HIV by further failing the immune system, causing rapid progression of HIV to AIDS. It is known that food and nutritional intake can affect adherence to antiretroviral drugs (ARVs) as well as their effectiveness (Fawzi and Hunter, 1998; WHO, 2007).

Recommended Dietary Intake (RDI) is the average daily dietary intake level that is sufficient to meet the nutrient requirements of nearly all (97–98 %) healthy individuals in a particular life stage and gender group (Capra, 2006). Normally the RDI values are arranged according to age, weight, height, and gender of individuals' (Khan and Al Kanhal, 1998; Australian Government Department of Health and Ageing, National Health and Medical Research Council, 2005; Capra, 2006; Scientific Advisory Committee on Nutrition (SACN), 2011).

People living with HIV (PLWHIV) have additional requirements of energy compared to non-infected persons. An adult with early infection (asymptomatic stage) needs 10% to 20% of extra energy while at symptomatic stage needs 20% to 30% extra energy above the recommended daily/dietary intake (RDI) compared to a non-infected adult of the same age and sex. Children with early infection (asymptomatic stage) require 10% extra energy, but 20% to 30% extra energy at symptomatic stage compared to a non-infected child of the same age and sex. There is no strong evidence to suggest that HIV infection increases protein, vitamins and mineral requirements above the RDI of the non-infected individuals (World Health Organization (WHO), 2007; Tanzania Food and Nutrition Center (TFNC), 2009).

A linear programming optimization model has been employed to model the diet for Malaysian boarding school caterers with the aim of minimizing the budget provided by the Malaysian government and to maximize the variety of food intake (Sufahani and Ismail, 2014). This model also wanted to achieve the maximum nutritional requirement based on the Malaysian Recommended Daily Allowance (RDA) requirements. Also, a linear programming optimization model has been used to model food consumption to meet daily nutrients needed by the standard woman and the standard man at the minimal cost respecting the World Health Organization (WHO) standards. The cost of foods to be minimized, is defined as the objective function in the optimization model (Pasec *et al.*, 2011). The linear programming model is an influential tool for planning optimized diets, it is a useful mathematical technique that can be used to easily translate accepted nutritional recommendations into complete food-based guidelines basing on locally available foods and local market prices. Besides, linear programming is much more efficient than the experiential “trial and error” approach currently used for formulating diets (Briend *et al.*, 2003). Linear programming technique is a great approach for identifying a low cost nutritional diet (Smith, 1959).

Elsewhere, linear programming model has been used to determine the balanced and least costly diets for different age groups who suffer from malnutrition due to deficiencies in their food budget in the poor community in Greater Amman, Jordan (Akram *et al.*, 2003).

Czyzyk *et al.* (1996) formulated a diet problem with a linear programming model. This formulation included the diet problem web page where the user is shown the steps to convert the verbal formulation of the problem into a mathematical problem. According to Czyzyk *et al.* (1999), a linear programming model is used to find a one-day ration food for a person to meet their nutritional requirements while minimizing cost.

Multiple linear regression analysis is carried out to prophesy the values of a dependent variable Y given a set of N explanatory variables X_1, X_2, \dots, X_N (Tranmer and Elliot, 2008). Multiple regression analysis is a highly flexible system for observing the relationship of a collection of independent variables (or predictors) to a single dependent variable (or criterion). The independent variables may be quantitative or categorical (Aiken *et al.*, 2003).

Different researchers have described the importance of achieving nutrient recommendation for people living with HIV, but none (to our knowledge) has described how to achieve it using a simple model. Furthermore, different studies in diet modelling have not considered factors that determine nutrient required in the body during the optimization process. Without considering these factors over/under taking of the nutrients may result due to the fact that RDI values are determined by factors such as age, weight, height, and gender of individuals. This chapter aims to develop a mathematical model which combines multiple linear regression model and linear programming model. Multiple linear regression model will predict macro and micro nutrient required in the human body by an individual's age, weight, height and gender, so as to avoid over or under taking nutrients, and linear programming model will compute adequate amount of foods that will determine optimal nutrients. The developed mathematical model will act as a tool to enable people living with HIV to achieve the nutrients recommended.

2.2. Research Methodology

In this chapter, a linear programming and multiple linear regression models have been formulated and combined in order to form a cost-effective computing (mathematical) model that will be used to compute adequate amount of foods in order to meet the optimal nutrient needs of people living with HIV at a minimized cost in consideration of age, weight, height, and gender of an individual at an asymptomatic stage. The Recommended Dietary Intake (RDI) of nutrients used in this study is borrowed from (Khan and Al Kanhal, 1998; Australian Government Department of Health and Ageing, National Health and Medical Research Council, 2005; Scientific Advisory Committee on Nutrition (SACN), 2011) for normal persons. Energy requirements for PLWHIV have been obtained by adding 15% and 10% to a normal person's requirement for adults and children respectively. The food composition table is suggested by (TFNC, 2009).

2.3. Mathematical Model Formulation

The mathematical model is expected to predict the nutrient requirement in the human body and determine adequate amount of foods that will meet the optimal nutrients. The nutrient requirement will be predicted by the multiple linear regression model and the adequate amount of foods will be computed by the linear programming model.

2.3.1. Multiple Linear Regression Model

The regression analysis has assumptions to be met for its results to be considered valid, predictive and useful. In this chapter, four main assumptions have been considered before starting the regression analysis. According to Poole and O'Farrell (1971); Osborne and Waters (2002); Williams *et al.* (2013), it has been stated that the four assumptions which needs to be met include normality of errors (residuals), linearity of parameters in the model, assumptions about the model error, and assumption about measurement error (reliability).

2.3.1.1. Normality of Errors or Residuals

In this assumption, errors are assumed to be normally distributed in any combination of values of the predictor variables, the properties of the errors are investigated by calculating the residuals of a regression model in the sample data. It should be noted that it is not necessary for predictors to be normally distributed (Poole and O'Farrell, 1971; Seber and Lee, 2012; Williams *et al.*, 2013).

2.3.1.2. Linearity of Parameters in the Model

The response variable is assumed to be a linear function of the parameters such as $\beta_0, \beta_1, \beta_2, \beta_3, \dots, \beta_p$ but not necessarily a linear function of the predictor variables $X_0, X_1, X_2, X_3, \dots, X_p$ (Poole and O'Farrell, 1971; Seber and Lee, 2012; Williams *et al.*, 2013).

2.3.1.3. Assumptions about the Model Error

According to Osborne and Waters (2002); Weisberg (2005, 2014); Chatterjee and Hadi (2013); Williams *et al.* (2013), the assumptions about the model error include: mean of errors should be zero, independence of errors, homoscedasticity that is constant variation of errors and normal distribution of errors as has been named and explained above.

2.3.1.4. Assumption about Measurement Error (Reliability)

It is assumed that predictor variables are measured without errors, and errors in the response variable are considered not harmful to inferences concerning to unstandardized regression coefficients, so long as this measurement error is not correlated with the explanatory variable values (Poole and O'Farrell, 1971; Osborne and Waters, 2002; Williams *et al.*, 2013).

Apart from the four assumptions above, the regression analysis put into consideration other factors such as measure of goodness of fit (R^2), and the significance of the model and coefficients. We have performed a regression analysis of each nutrient as response variable on explanatory variables age (A), weight (W), height (H), and gender (G), in consideration of the assumptions above. Where A, W and H are continuous quantitative variables while G is a categorical variable. The regression analysis resulted into the following multiple linear regression model, but for more details on analysis see appendix 1 to appendix 26:

$$RDI_{\text{Energy}} = 0.002A^3 - 19.773A + 14.399W + 18.36H + 178.263G - 753.860 \quad (1)$$

$$RDI_{\text{Protein}} = 3.028 \times 10^{-5}A^3 + 25.676\ln(W) + 10.193G - 58.621 \quad (2)$$

$$RDI_{\text{Fiber}} = 2.247\ln(A) + 2.086 \times 10^{-5}W^3 + 2.309G + 12.641 \quad (3)$$

$$RDI_{\text{Vitamin A}} = 53.706\ln(A) - 9.815W - 15.252H + 0.093H^2 + 912.549 \quad (4)$$

$$RDI_{\text{Vitamin E}} = 2.930 \times 10^{-6}A^3 + 2.260 \times 10^{-5}W^3 + 0.444W + 0.001H^2 + 2.762 \quad (5)$$

$$RDI_{\text{Vitamin C}} = 0.850W - 0.003W^2 - 0.001H^2 + 32.492 \quad (6)$$

$$RDI_{\text{Thiamin}} = 1.860 \times 10^{-7}A^3 - 4.845 \times 10^{-7}W^3 + 1.844 \times 10^{-7}H^3 - 0.035G + 0.38 \quad (7)$$

$$RDI_{\text{Riboflavin}} = 0.110\ln(A) + 1.186 \times 10^{-7}H^3 + 0.286 \quad (8)$$

$$RDI_{\text{Niacin}} = 2.434 \times 10^{-6}A^3 - 0.001W^2 + 0.001H^2 - 0.107H + 8.002 \quad (9)$$

$$RDI_{\text{Vitamin B6}} = 7.226 \times 10^{-7}A^3 - 1.334 \times 10^{-6}W^3 + 0.023W + 0.192 \quad (10)$$

$$RDI_{\text{Folate}} = 17.576W - 0.133W^2 - 5.220H + 0.014H^2 + 303.834 \quad (11)$$

$$RDI_{\text{VitaminB12}} = 0.063W - 4.486 \times 10^{-6}W^3 - 0.006H^2 + 0.696 \quad (12)$$

$$RDI_{\text{Panthothenic}} = 1.680 \times 10^{-6}A^3 + 3.925 \times 10^{-5}W^3 - 0.004W + 1.698 \times 10^{-6}H^3 + 0.451G + 2.649 \quad (13)$$

$$RDI_{\text{Phosphorus}} = -0.005W^3 + 0.253H^2 - 42.095H + 2169.802 \quad (14)$$

$$RDI_{\text{Potassium}} = 252.852\ln(A) - 88.411W + 0.007W^3 + 283.435G + 1474.871 \quad (15)$$

$$RDI_{\text{Sodium}} = 14.035W - 0.108W^2 + 5.548H - 0.017H^2 - 204.433 \quad (16)$$

$$RDI_{\text{Zinc}} = 5.451 \times 10^{-6}A^3 + 6.699 \times 10^{-5}W^3 - 0.579W + 0.248H - 12.172 \quad (17)$$

$$RDI_{\text{Copper}} = 0.136\ln(A) + 4.568 \times 10^{-6}W^3 - 0.051W + 0.022H + 0.128G - 0.776 \quad (18)$$

$$RDI_{\text{Manganese}} = 2.292 \times 10^{-6}A^3 - 9.007 \times 10^{-6}W^3 + 2.250 \quad (19)$$

$$RDI_{\text{Upper Vitamin E}} = 12.965W - 0.042W^2 - 6.522 \times 10^{-5}H^3 - 43.248 \quad (20)$$

$$RDI_{\text{Upper Niacin}} = 2.989\ln(A) + 0.311W + 3.512 \quad (21)$$

$$RDI_{\text{Upper Vitamin B6}} = 3.419\ln(A) + 1.383W - 6.155 \times 10^{-5}W^3 - 0.389H + 30.112 \quad (22)$$

$$RDI_{\text{Upper Folate}} = 41.284\ln(A) + 40.128W - 0.154W^2 - 0.043H^2 + 132.850 \quad (23)$$

$$RDI_{\text{Upper Sodium}} = 129.946\ln(A) - 0.157W^2 + 0.074H^2 + 354.670 \quad (24)$$

$$RDI_{\text{Upper Zinc}} = 1.461W - 6.195 \times 10^{-5}W^3 - 6.192 \times 10^{-6}H^3 - 7.164 \quad (25)$$

$$RDI_{\text{Upper Copper}} = 0.805\ln(A) + 0.312W - 1.484 \times 10^{-5}W^3 - 0.074H + 3.286 \quad (26)$$

2.3.2. Linear Programming Model

The linear programming model involves food items as variable x_i , cost of food i as parameter c_i , nutrient j contained in food i is given as A_{ji} . Moreover, b_1 is the maximum allowable amount of nutrient j , while b_2 is the minimum amount of nutrient j required and $F_{\max i}$ is the maximum allowable amount of food i , from the multiple linear regression model we define $b_1 = \{RDI_{\max j}\}$ and $b_2 = \{RDI_{\min j}\}$ where $RDI_{\max j}$ is the set of multiple linear regression model for maximum recommended nutrient intake and $RDI_{\min j}$ is the set of multiple linear regression model for minimum recommended nutrient intake. Thus b_1 include equations numbered 20 through 26, where b_2 include equations numbered 1 to 19. Mathematically, a linear programming model is written as:

$$\begin{aligned} &\text{minimize} \quad \sum_{x_i} c_i x_i \\ &\text{subject to} \quad A_{ji} \leq b_1 \\ &\quad \quad \quad A_{ji} \geq b_2 \\ &\quad \quad \quad x_i \leq F_{\max i} \\ &\quad \quad \quad x_i \geq 0 \end{aligned} \tag{27}$$

2.4. Results

The results have been organized in two categories: the first is the predictive power of our multiple linear regression model and the second is the cost comparison incurred on buying the computed adequate amount of foods when the model predicted values and actual RDI values are used as the nutritional recommendations to limit the constraints in the linear programming model.

2.4.1. Multiple Linear Regression Model Predictive Power

In this part we want to see the predictive power of the multiple linear regression model developed, we compare the values predicted by the model to the actual RDI values. The comparison is considering the model predicted values and RDI values for a specific individual. The gender has

been arbitrary represented as 0 for women and 1 for men. We have randomly considered a man aged 24.5 years, weighing 71.5 kg, of height 178 cm and a woman of 60.5 years old, weighing 58 kg and of height 161 cm for this comparison. The results were as in the Table 2.1 below.

Table 2.1: Comparison of model predicted values to the actual RDI values

Model predicted value	RDI	Gender	Height (Cm)	Weight (kg)	Age (years)	Model predicted value	RDI	Gender	Height (cm)	Weight (kg)	Age (years)
		0	161	58	60.5			1	178	71.5	24.5
2285.0	2391.195					3268.2	3188.26	Energy			
52.3	46					61.6	64	Protein			
25.9	25					29.8	30	Fiber			
519.0	500					614.7	625	Vitamin A			
8.0	7					11.0	10	Vitamin E			
45.8	45					46.2	45	Vitamin C			
1.1	1.1					1.2	1.2	Thiamin			
1.2	1.1					1.3	1.3	Riboflavin			
13.9	14					15.6	16	Niacin			

Vitamin B6	1.3	1.4		1.5	1.4
Folate	400	395.0		400	398.3
Vitamin B12	2.4	2.5		2.4	2.5
Pantothenic	6	6.6		4	4.3
Phosphorus	1000	865.3		1000	975.0
Potassium	3800	3873.8		2800	2897.5
Sodium	690	695.9		690	698.9
Zinc	14	15.1		8	8.5
Copper	1.7	1.7		1.2	1.3
Manganese	5.5	5.6		5	4.5
Upper vitamin E	300	301.2		300	295.3
Upper niacin	35	35.3		35	33.8
Upper vitamin B6	50	48.2		50	49.7
Upper folate	1000	984.4		1000	997.0
Upper sodium	2300	2312.3		2300	2277.8

Upper zinc	40	39.7		40	39.6
Upper copper	10	9.6		10	9.9

From Table 2.1 above, we validate the model by testing for difference between the two means of RDI and the model predicted values. Let μ_1 be mean of RDI while that of model predicted values be μ_2 . We use the Two-Sample t Test for the Two-Tailed Hypotheses, to test for difference between the two means, we define the null hypothesis as $H_o : \mu_1 = \mu_2$; that is there is no difference between the two means. And we define the alternative hypothesis as $H_A : \mu_1 \neq \mu_2$; that is there is no difference between the means. The significance level used is 0.05, so we reject the null hypothesis if and only if the p – value is less than 0.05 otherwise we accept it.

From Table 2.1 we have two pairs, the first is for male aged 24.5 years old and the second is for woman aged 60.5 years old. In tables below, RDI24.51 and MODEL24.51 stands for RDI and model predicted values for male aged 24.5 years respectively, while RDI60.50 and MODEL60.50 stands for RDI and model predicted values for woman aged 60.5 years respectively.

We have used the SPSS software for this analysis, and the results are as follows:

Table2.2: Paired Samples Statistics

		Mean	N	Std. Deviation	Std. Error Mean
Pair 1	RDI24.51	524.4869	26	1017.30142	199.50922
	MODEL24.51	524.7385	26	1033.72105	202.72938
Pair 2	RDI60.50	449.1344	26	814.60914	159.75800
	MODEL60.50	447.9808	26	813.55301	159.55087

Table2.3:PairedSamples Correlations

	N	Correlation	Sig.
Pair 1 RDI24.51 and MODEL24.51	26	1.000	.000
Pair 2 RDI60.50 and MODEL60.50	26	.999	.000

Table2.4:Paired Samples Test

	Paired Differences					t	df	Sig. (2-tailed)
	Mean	Std. Deviation	Std. Error Mean	95% Confidence Interval of the Difference				
				Lower	Upper			
Pair 1 RDI24.51 - MODEL24.51	-.25154	34.95990	6.85620	-14.37215	13.86907	-.037	25	.971
Pair 2 RDI60.50 - MODEL60.50	1.15365	29.92359	5.86850	-10.93275	13.24005	.197	25	.846

From the paired sample test in Table 2.4; the p -value for the first pair (a man aged 24.5 years) is $0.971 > 0.05$ and that of second pair (a woman aged 60.5 year) has a p -value $0.846 > 0.05$, meaning that there is no means difference between the RDI and model predicted values in both case.

But also from the Table 2.1, model predicted values are close enough to the actual RDI values, which shows that the model developed has high predictive power. This convinced us that the multiple linear regression model could be used to predict the nutritional recommendation to the human body by factors age, weight, height, and gender. Results from multiple linear regression model were the minimum amount of nutrients recommended and maximum amount of nutrients

allowed. These results were used to limit/defines the constraints in the linear programming model, so as to find out adequate amount of foods that would meet the optimal nutrients required.

2.4.2. Linear Programming Model Results

A person who is infected with HIV and is not showing signs of illness does not need a specific HIV-diet. However, those infected with HIV should make every effort to adopt healthy and balanced nutrition patterns in order to meet their increased energy requirements and maintain their nutritional status, so as to enable their immune systems fight against opportunistic diseases (Food and Agriculture Organization (FAO) and World Health Organization (WHO), 2002). In this case, suppose there are ten foods which are easily found in a certain area, these foods are rice with coconut milk, banana with meat and coconut milk, beef, avocado, papaya juice, spinach, African doughnut, cassava porridge, dried fresh water fish and green salad. From this list an individual can select what to eat, for instance four individuals of different age, where men and woman are equal in numbers have selected all these. The formulated problem was solved by a two phase simplex method in MATLAB, we wrote the function *TwoPhaseSimplex* (see appendix 27) which calls the *totbl* and *lpx* functions in order to solve this problem by the two phase simplex method. The function *totbl* is used to display the tableau while the function *lpx* is used to exchange the pivot row and pivot column in the tableau (Ferris *et al.*, 2007). Then we have compared the cost incurred when we use the values of our multiple linear regression model and the actual RDI values as nutritional recommendations, to compute adequate amount of foods. The results from the problem were as follows:

Table 2.5: Cost comparison incurred for computing adequate amount of foods to reach the optimal nutrients using our model predicted values and using RDI values

Age (years)	Gender	RDI	Developed model
11	1	Rice with coconut milk =0.1681 kg	Rice with coconut milk=0.2500 kg Avocado = 0.2500 kg

		<p>Banana with meat and coconut milk = 0.2258 kg</p> <p>Papaya juice = 0.7644 kg</p> <p>Spinach = 0.05 kg</p> <p>African Doughnut = 0.250 kg</p> <p>Dried fresh water fish = 0.1663 kg</p> <p>Green salad = 0.0952 kg</p>	<p>Spinach = 0.0487 kg</p> <p>African Doughnut = 0.2500 kg</p> <p>Cassava porridge = 0.2500 kg</p> <p>Dried fresh water fish = 0.1334 kg</p> <p>Green salad = 0.1563 kg</p>
Total cost (Tsh)		3541.7816	2311.75
40.5	0	<p>Rice with coconut milk = 0.2500 kg</p> <p>Banana with meat and coconut milk = 0.0824 kg</p> <p>Avocado = 0.1250 kg</p> <p>Spinach = 0.0405 kg</p> <p>African Doughnut = 0.2500 kg</p> <p>Cassava porridge = 0.1000 kg</p> <p>Dried fresh water fish = 0.2000 kg</p> <p>Green salad = 0.2500 kg</p>	<p>Rice with coconut milk = 0.2500 kg</p> <p>Banana with meat and coconut milk = 0.1819 kg</p> <p>Avocado = 0.2500 kg</p> <p>Spinach = 0.1609 kg</p> <p>African Doughnut = 0.2500 kg</p> <p>Cassava porridge = 0.2500 kg</p> <p>Dried fresh water fish = 0.0677 kg</p> <p>Green salad = 0.2500 kg</p>

Total cost (Tsh)		2759.3391	2441.1
16	0	Rice with coconut milk = 0.2500 kg Banana with meat and coconut milk = 0.2500 kg Beef = 0.0555 kg Avocado = 0.1250 kg African Doughnut = 0.2500 kg Cassava porridge = 0.1000 kg Dried fresh water fish = 0.2000 kg Green salad = 0.2500 kg	Rice with coconut milk = 0.2500 kg Banana with meat and coconut milk = 0.2500 kg Avocado = 0.2500 kg African Doughnut = 0.2500 kg Cassava porridge = 0.2500 kg Dried fresh water fish = 0.1359 kg Green salad = 0.2500 kg
Total cost (Tsh)		3483.1124	2975.52
80.5	1	Not optimal	Rice with coconut milk = 0.2500 kg Banana with meat and coconut milk = 0.2500 kg Avocado = 0.2500 kg Spinach = 0.0102 kg African Doughnut = 0.2500 Kg Cassava porridge = 0.2500 Kg

			Dried fresh water fish = 0.1936 Kg Green salad = 0.2500 Kg
Total cost (Tsh)			3387.62

The cost comparisons from Table 2.5 above can be presented graphically in Figure 2.1 below.

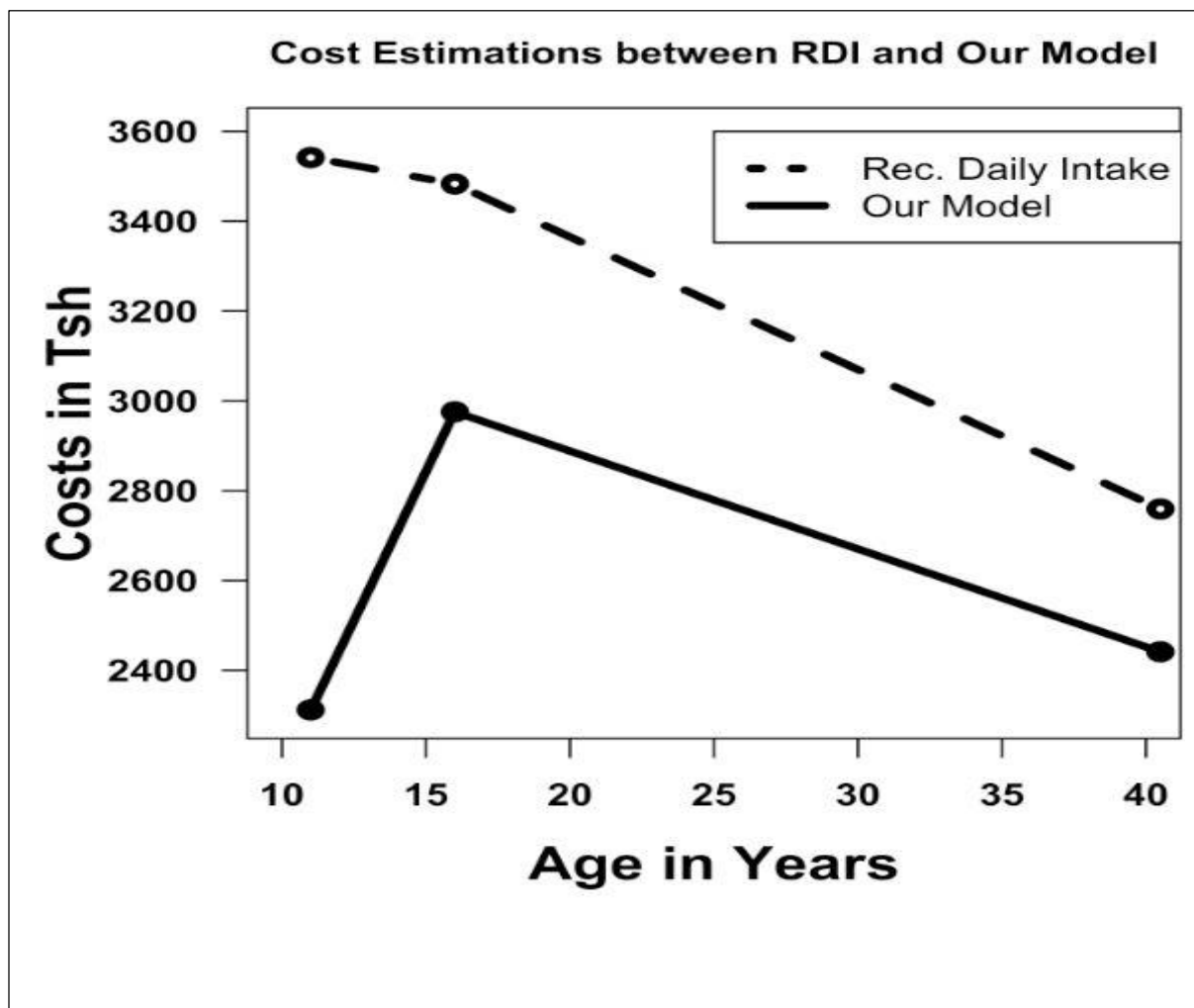


Figure 2.1: Cost comparison when RDI values and model predicted values were used to compute adequate amount of foods to achieve recommended nutrients

Table 2.5 and Figure 2.1 shows that when multiple linear regression and linear programming models are combined they provide optimal nutrients for a cost much less than when the actual RDI

nutrient recommendations values are used. Thus, our model achieves optimal nutrients for the patients at a much lower cost than the corresponding RDI nutrients. For instance a young man who is 11 years old has met the optimal nutrients by the cost of TSH 3541.7816 when actual RDI values are used, but using our model he meets the optimal nutrients at TSH 2311.75, this cost has decreased by approximately 34.73%. A woman whose age is 40.5 years old is meeting her nutritional requirements at TSH 2759.3391 when RDI values are used, but uses only TSH 2441.1 to meet her nutritional recommendation using our model. The cost will decrease by approximately 11.53%. Similarly a woman aged 16 years old will use TSH 3483.1124 to meet the nutritional recommendation when RDI values are used while she will use TSH 2975.52 when the values of our model are used, this cost has decreased by approximately 14.6%. The man whose age is 80.5 years will not meet the optimal nutrients from this food list if the RDI values are used, however, he will meet optimal nutrients using our model at a cost of TSH 3387.62. Thus, through this developed mathematical model, people living with HIV will be in a position to meet the amount of recommended nutrients at less cost.

2.5. Discussion and Conclusion

The chapter has developed a mathematical model that can be used to manage the nutritional requirements for people living with HIV at an asymptomatic stage at least cost. This mathematical model combines multiple linear regression and linear programming models. Multiple linear regression models predict the nutritional requirements in the human body by factors weight, age, height and gender, and its results are the maximum allowable amount of nutrients and minimum amount of nutrients required. These results are used in the linear programming model to limit the constraints set. The linear programming model computes the adequate amount of foods that determine the optimal nutrients.

We saw that the multiple linear regression models have high predictive powers as their values were close enough to the actual values of RDI. The optimality in the linear programming is reached even when these two models are combined and good enough the optimal solution is at least cost-saving compared to RDI. This model has been considered because individuals of the same age and sex may not necessarily have the same weight and height, which make the difference of nutrient required in the body. By predicting the nutritional requirements and providing optimal estimates, our model avoids over or undertakings of the nutrients for the patient thus ensuring a balanced

nutrient intake for them. The model reaches optimal nutritional recommendation at a cost lower than when RDI values are used, this will enable more people to have a balanced diet. In the absence of HIV/AIDS cure this will help individuals to live healthier and continue having a positive contribution at the household and national levels both socially and economically. The model developed can be extended to different groups of people who must manage their diets and therefore promises to have a wider applicability.

The study is limited to a few nutrients as we have considered nineteenth nutrients, this was done for demonstration purposes. The fact that our model has shown to produce useful information makes us eager to extend it by adding even more nutrients in the future. Another future plan is development of a graphical user interface (GUI) in different programming languages, web page or mobile application that will work as a computation tool by giving opportunity to clients just to select the foods they want to eat, submit their age, weight, height and gender, but also the cost of foods they have chosen and obtain back the adequate amount of foods that will meet their optimal nutrient requirements at a minimized cost.

CHAPTER THREE

A MATLAB Based Graphical User Interface for Computing Optimal Diet of People Living with HIV at Asymptomatic Stage²

Abstract: Achieving basic nutritional recommendations for people living with HIV is important at all stages as the virus affects the immune system and renders an infected person to contract other infections easily. Poor nutrition strengthens the effects of HIV as malnutrition adds to the deterioration of the immune system. This leads to a poor prognosis of the disease. Thus a tool to determine adequate amount and good combination of different available foods that ensures optimal nutrients will prove useful if designed. In this chapter, a MATLAB based Graphical User Interface (GUI) that will be used to compute adequate amount of foods, leading to the achievements of recommended nutrient intakes for these patients has been developed. The design and use of a GUI can be extended to various telecommunication devices and be accessible to a normal person and other groups in needs of diet management and therefore it will have a wider application.

Keywords

MATLAB, Graphical User Interface, HIV, optimal diet, computing

3.1. Introduction

It is important that people living with HIV (PLWHIV) keep their bodies as healthy as possible by making sure that they get all the nutrients a person needs as HIV affects their immune systems making them susceptible to opportunistic infections (Clark *et al.*, 2012). The effects of HIV are worsened by poor nutrition, which contributes to the deterioration of the immune system (World Health Organization (WHO), 2007). A healthy and balanced diet will help to maintain and improve the performance of the immune system for the body's protection against infection and therefore helps a person stay healthy (Akram *et al.*, 2003).

PLWHIV have additional requirements of energy compared to non-infected persons. An adult with early infection (asymptomatic stage) needs 10% to 20% of extra energy while at symptomatic

² This chapter is based on the research paper titled:

Yasin Kaunda Kowa, Emmanuel Mpolya, and Yaw Nkansah-Gyekye. "A MATLAB Based Graphical User Interface for Computing Optimal Diet of People Living with HIV at Asymptomatic Stage". *International Journal of Computer Applications* 121(20):31-35, July 2015

stage needs 20% to 30% extra energy above the recommended daily/dietary intake (RDI) compared to a non-infected adult of the same age and sex. Children with early infection (asymptomatic stage) require 10%, and 20% to 30% extra energy on symptomatic stage compared to non-infected children of the same age and sex. There is no strong evidence to suggest that HIV infection increases protein, vitamins and mineral requirements above the RDI of the non-infected individuals (Tanzania Food and Nutrition Center (TFNC), 2009; WHO, 2007). In this paper, PLWHIV who are at early stages of infection (asymptomatic stage) have been considered since at this stage their weights are equivalent to the non-infected (normal) people of the same age, height and sex and we want them to maintain their healthy status.

A graphical user interface (GUI) is a graphical display in one or more windows containing controls, called components, which enables a user to perform interactive tasks (Guruhappa *et al.*, 2013; Hunt *et al.*, 2014). The user of the GUI does not have to create a script or type commands at the command line to accomplish the tasks and need not understand the details of how the tasks are performed. Using the MATLAB programming language, one can create a GUI which contains controls such as menus, toolbars, buttons and sliders (Guruhappa *et al.*, 2013; Hunt *et al.*, 2014; Yang *et al.*, 2009).

This chapter aims to develop a MATLAB based graphical user interface that will enable people living with HIV to compute the adequate amount of foods that will meet the optimal nutrients (nutritional recommendations) required in their bodies. In the absence of cure this will help better manage their health and continue having important contributions to social and economic sectors at the household and national levels.

3.2. Research Methodology

In this chapter, a MATLAB based Graphical User Interface (GUI) has been developed by GUIDE (graphic user interface development environment) method. To achieve the goal intended, codes that combine multiple linear regression model and a linear programming model, and a two phase simplex method algorithm are added in the *function computation_callback*. A two phase simplex method will be used to compute the adequate amount of foods from a mathematical model incorporated into the GUI. The food composition table for the list of foods in the GUI has been suggested by (TFNC, 2009).

3.3. Mathematical Model Formulation

The mathematical model combines multiple linear regression and linear programming models; the regression model uses the factors age, weight, height, and gender to capture and predict the nutritional needs of the body. The results from the multiple linear regression model defines/limits the constraints in the linear programming model we have set. The constraints are restricted to the minimum amount of nutrients required and maximum allowable amount of nutrients, but also the maximum allowable amount of food.

3.3.1. Regression Analysis

According to Poole and O'Farrell (1971); Williams *et al.* (2013) who corrected Osborne and Waters (2002), for a multiple linear regression analysis (model) to be valid, four assumptions need to be met; these include normality of errors (residuals), linearity of parameters in the model, assumptions about the model error and assumption about measurement error (reliability).

3.3.1.1. Normality of Errors or Residuals

Here, it is assumed that errors are normally distributed in any combination of values of the predictor variables. The properties of the errors are investigated by calculating the residuals of a regression model in the sample data. It should be noted that it is not necessary for predictors to be normally distributed (Poole and O'Farrell, 1971; Williams *et al.*, 2013).

3.3.1.2. Linearity of Parameters in the Model

This means that the response variable is assumed to be a linear function of parameters such as $\beta_0, \beta_1, \beta_2, \beta_3, \dots, \beta_p$ but not necessarily a linear function of the predictor variables $X_1, X_2, X_3, \dots, X_p$ (Poole and O'Farrell, 1971; Seber and Lee, 2012; Williams *et al.*, 2013).

3.3.1.3. Assumptions about the Model Error

According to Chatterjee and Hadi (2013); Poole and O'Farrell (1971); Weisberg (2014); and Williams *et al.* (2013), this assumption includes mean of errors should be zero, independence of error, homoscedasticity, that is a constant variance of errors and normal distribution of errors as has been named and explained above.

3.3.1.4. Assumption about Measurement Error (Reliability)

The assumption assumes that predictor variables are measured without errors, error in the response variable are considered not harmful to inferences related to unstandardized regression coefficients, provided this measurement error is not correlated with the predictor variable value (Poole and O'Farrell, 1971; Osborne and Waters, 2002; Williams *et al.*, 2013).

Apart from the four assumptions above, the regression analysis has considered factors such as a measure of goodness of fit R^2 , and the significance of the model and coefficients.

From the regression analysis of each nutrient as a response variable on explanatory variables age (A), weight (W), height (H) and gender (G) see appendix 1 to appendix 26, the analysis resulted to the following regression models:

$$RDI_{\text{Energy}} = 0.002A^3 - 19.773A + 14.399W + 18.36H + 178.263G - 753.860 \quad (1)$$

$$RDI_{\text{Protein}} = 3.028 \times 10^{-5}A^3 + 25.676\ln(W) + 10.193G - 58.621 \quad (2)$$

$$RDI_{\text{Fiber}} = 2.247\ln(A) + 2.086 \times 10^{-5}W^3 + 2.309G + 12.641 \quad (3)$$

$$RDI_{\text{Vitamin A}} = 53.706\ln(A) - 9.815W - 15.252H + 0.093H^2 + 912.549 \quad (4)$$

$$RDI_{\text{Vitamin E}} = 2.930 \times 10^{-6}A^3 + 2.260 \times 10^{-5}W^3 + 0.444W + 0.001H^2 + 2.762 \quad (5)$$

$$RDI_{\text{Vitamin C}} = 0.850W - 0.003W^2 - 0.001H^2 + 32.492 \quad (6)$$

$$RDI_{\text{Thiamin}} = 1.860 \times 10^{-7}A^3 - 4.845 \times 10^{-7}W^3 + 1.844 \times 10^{-7}H^3 - 0.035G + 0.38 \quad (7)$$

$$RDI_{\text{Riboflavin}} = 0.110\ln(A) + 1.186 \times 10^{-7}H^3 + 0.286 \quad (8)$$

$$RDI_{\text{Niacin}} = 2.434 \times 10^{-6}A^3 - 0.001W^2 + 0.001H^2 - 0.107H + 8.002 \quad (9)$$

$$RDI_{\text{Vitamin B6}} = 7.226 \times 10^{-7}A^3 - 1.334 \times 10^{-6}W^3 + 0.023W + 0.192 \quad (10)$$

$$RDI_{\text{Folate}} = 17.576W - 0.133W^2 - 5.220H + 0.014H^2 + 303.834 \quad (11)$$

$$RDI_{\text{VitaminB12}} = 0.063W - 4.486 \times 10^{-6}W^3 - 0.006H^2 + 0.696 \quad (12)$$

$$RDI_{\text{Panthothenic}} = 1.680 \times 10^{-6}A^3 + 3.925 \times 10^{-5}W^3 - 0.004W + 1.698 \times 10^{-6}H^3 + 0.451G + 2.649 \quad (13)$$

$$RDI_{\text{Phosphorus}} = -0.005W^3 + 0.253H^2 - 42.095H + 2169.802 \quad (14)$$

$$RDI_{\text{Potassium}} = 252.852\ln(A) - 88.411W + 0.007W^3 + 283.435G + 1474.871 \quad (15)$$

$$RDI_{\text{Sodium}} = 14.035W - 0.108W^2 + 5.548H - 0.017H^2 - 204.433 \quad (16)$$

$$RDI_{\text{Zinc}} = 5.451 \times 10^{-6}A^3 + 6.699 \times 10^{-5}W^3 - 0.579W + 0.248H - 12.172 \quad (17)$$

$$RDI_{\text{Copper}} = 0.136\ln(A) + 4.568 \times 10^{-6}W^3 - 0.051W + 0.022H + 0.128G - 0.776 \quad (18)$$

$$RDI_{\text{Manganese}} = 2.292 \times 10^{-6}A^3 - 9.007 \times 10^{-6}W^3 + 2.250 \quad (19)$$

$$RDI_{\text{Upper Vitamin E}} = 12.965W - 0.042W^2 - 6.522 \times 10^{-5}H^3 - 43.248 \quad (20)$$

$$RDI_{\text{Upper Niacin}} = 2.989\ln(A) + 0.311W + 3.512 \quad (21)$$

$$RDI_{\text{Upper Vitamin B6}} = 3.419\ln(A) + 1.383W - 6.155 \times 10^{-5}W^3 - 0.389H + 30.112 \quad (22)$$

$$RDI_{\text{Upper Folate}} = 41.284\ln(A) + 40.128W - 0.154W^2 - 0.043H^2 + 132.850 \quad (23)$$

$$RDI_{\text{Upper Sodium}} = 129.946\ln(A) - 0.157W^2 + 0.074H^2 + 354.670 \quad (24)$$

$$RDI_{\text{Upper Zinc}} = 1.461W - 6.195 \times 10^{-5}W^3 - 6.192 \times 10^{-6}H^3 - 7.164 \quad (25)$$

$$RDI_{\text{Upper Copper}} = 0.805\ln(A) + 0.312W - 1.484 \times 10^{-5}W^3 - 0.074H + 3.286 \quad (26)$$

3.3.2. Linear Programming Model

Linear programming model involves food items as the variable x_i , parameter cost of food i as c_i , the nutrient j contained in food i is given as A_{ji} , b_1 the maximum allowable amount of nutrient j , b_2 the minimum amount of nutrient j required and F_{maxi} the maximum allowable amount of food i , from the multiple linear regression model, b_1 and b_2 are define as $b_1 = \{RDI_{maxj}\}$ and $b_2 = \{RDI_{minj}\}$, where RDI_{maxj} is the regression model for minimum recommended amount of nutrient j and RDI_{minj} is the regression model for maximum recommended amount of nutrient j . Mathematically this is written as:

$$\begin{aligned} & \text{minimize } \sum_{x_i} x_i c_i \\ & \text{subject to } A_{ij} \leq b_1 \\ & \quad A_{ij} \geq b_2 \\ & \quad x_i \leq F_{maxi} \\ & \quad x_i \geq 0 \end{aligned} \tag{27}$$

3.4. Graphical User Interface Development

A Graphical User Interface (GUI) in MATLAB was developed by the GUIDE method (appendix 28). GUIDE enables the rapid creation of a graphical user interface (GUI) for a MATLAB program (Hunt *et al.*, 2014; Kumar *et al.*, 2013; Lent, 2013). The two phase simplex method which is used for computing the adequate amount of foods from the mathematical model uses the codes from *TwoPhaseSimplex* (appendix 27) function which call the *totbl* and *lfx* functions, the last two functions are from (Ferris *et al.*, 2007). The *totbl* displays the tableau while the *lfx* exchange the pivot column and pivot row. In order to achieve the intended goal, the code in the *TwoPhaseSimplex* (appendix 27) function identifies the pivot column and pivot row automatically, but originally the pivot row and column were identified manually, also the extended codes displays the answer to the GUI automatically rather than reading from the tableau. The result from the two phase simplex method will either be optimal or not optimal, whatever the result obtained the user will be notified.

The GUI has five major parts: the first is the list of foods which contain a variety of foods that the user will choose. To select multiple foods she/he need to press the control button on the keyboard while the mouse click on the foods she/he want to eat.

The second part is to enter her/his particulars: these are the age in years, weight in kilograms, height in centimeters, and gender, where men have to enter 1 while women enter 0; note that 1 and 0 are arbitrary. Once she/he enters her/his particulars the mathematical model captures and predicts her/his nutrients requirement in the body. The third part is to enter the cost of each food that has been chosen (the cost should be per kilogram) and there should be a space between the costs.

The fourth is the computation part; after the first three parts the user has to click the compute button for computation to take place. After computation the status box will state whether her/his selection is optimal or not, it is optimal means she/he can take that combination, but if not optimal she/he needs to look for another combination, although she/he needs to reset by clicking the reset button before starting the computation again. The reset button clears the previous inputs and results.

The last part is the see the results; this contains the index of food component which display the index of foods which should be bought and the amount component indicates what amount of foods in kilograms she/he need to buy. Then the total cost component gives the total cost that will be incurred on buying these foods. But note that the index of foods will be X_1 for the first food that has been chosen, X_2 for the second food chosen, and so on up to X_n for the last food selected. The last part will happen if and only if the computation is optimal. The GUI developed looks like in Figure 3.1 below, the steps to be taken are ordered from numbers 1 to 5. The client selects foods she/he wants to eat from the list given (step 1). Then proceeds to enter his/her particulars (age, weight, height and gender) – step 2- and the costs of the foods selected (step 3). The client then proceeds to compute (step 4) and obtains the output (step 5).

3.4.1. Illustration

Suppose there are ten foods which are easily found in a certain area, these foods are rice with coconut milk, banana with meat and coconut milk, beef, avocado, papaya juice, spinach, donut African, cassava porridge, fish dried fresh water and salad green whose estimated costs per kilogram are (2000, 2500, 6000, 700, 1500, 600, 1000, 1000, 7000, 500) Tanzania shillings (TSH) respectively. From this list an individual can select what to eat, the GUI help individuals compute

the amount of foods that will lead to achievement of nutritional recommendation, for instance a man whose age is 40.5 years, weighing 69.7 kg and height of 176 cm decided to select all foods then after computation to take place the GUI will look like in Figure 3.2 below.

From Figure 3.2 the man aged 40.5 years, weighting 69.7 kg and of height 176 cm has selected ten foods, submitted his particulars and cost of foods selected. The computation shows that the selection is optimal as seen in status component and he needs to buy 0.25 kg of rice with coconut milk, 0.25 kg of banana with meat and coconut milk, approximately 0.081 kg of beef, 0.15 kg of avocado, approximately 0.169 kg of spinach, 0.2kg of donut African, 0.2 kg of cassava porridge, 0.25 kg of fish dried fresh water and 0.15 kg of salad green. But also we see that he should not buy papaya juice in order to meet the recommended nutrients since does not appear in the “see the results” part of the GUI. The total cost that will be incurred for buying these foods is TSH 4,044.79.

Figure 3.3 below shows that nine foods have been selected by a man aged 40.5years, weighing 69.7 kg, and of height 176 cm. The cost of foods selected have been submitted. The computation shows that the selected combination is not optimal as stated in the status component. But also there is no results in the See the Results component. This means that he should not take such combination otherwise the recommended nutrients will not be met.

3.5. Discussion and Conclusion

Meeting the recommended nutrients for PLWHIV is vital as it will help to maintain and improve the performance of the immune systems. This will protect them from contracting other diseases easily. The study has developed a MATLAB based Graphical User Interface that gives opportunity to PLWHIV to compute adequate amount of foods that will meet the recommended nutrients. The GUI asks the user to enter their age, weight, height and gender in order to predict their nutrient requirements, select the foods they want to eat, to submit the cost of each food of their choice. This will accommodate the fact that the cost of foods has been always dynamic.

The GUI will give back the index of foods to be bought, amount of foods to be bought and the total cost that will be incurred to buy all foods, provided such combination is optimal. If the combination is not optimal there will be no results in the See the Results component and it will be stated that not optimal on the status component.

The study may be limited to the index of foods rather than real names, this may lead to misinterpretation. This could be adjusted in the future towards a more comprehensive and user-

friendly system. Also, currently the access of the Graphical User Interface is limited and this will be improved in future through the development of a web page or a mobile application which will be easily accessed.

GRAPHICAL USER INTERFACE FOR COMPUTING OPTIMAL DIET

1. List of foods you can choose

2. Enter your particulars

3. Enter the cost of foods you have chosen

4. Computation

5. See the Results

Index of food	Amount of food (Kg)	Total cost of foods (Tsh)

Figure 3.1: Graphical user interface for computing optimal diet

GRAPHICAL USER INTERFACE FOR COMPUTING OPTIMAL DIET

1.List of foods you can choose

Rice with coconut milk
Banana with meat and coconut milk
Beef
Avocado
Papaya juice
spinach
Donut African
Cassava porridge
Fish dried Fresh water
Salad green

4.Computation

COMPUTE

RESET

STATUS
optimal

3.Enter the cost of foods you have chosen

2000 2500 6000 700 1500 600 1000 1000 7000 500

2.Enter your particulars

Age (years)
40.5

Weight (Kg)
69.7

Height (cm)
176

Gender
1=men
0=woman
1

5.See the Results

Index of food	Amount of food (Kg)	Total cost of foods (Tsh)
x1	0.25	4044.79
x10	0.15	
x2	0.25	
x3	0.0814142	
x4	0.15	
x6	0.16884	
x7	0.2	
x8	0.2	
x9	0.25	

Figure 3.2: Illustration of GUI computation for optimal diet

GRAPHICAL USER INTERFACE FOR COMPUTING OPTIMAL DIET

1. List of foods you can choose

Rice with coconut milk
Banana with meat and coconut milk
Beef
Avocado
Papaya juice
spinach
Donut African
Cassava porridge
Fish dried Fresh water
Salad green

2. Enter your particulars

Age (years)

40.5

Weight (Kg)

69.7

Height (cm)

176

Gender
1=men
0=woman

1

3. Enter the cost of foods you have chosen

2000 2500 6000 700 1500 600 1000 1000 500

4. Computation

COMPUTE

RESET

STATUS
not optimal

5. See the Results

Index of food	Amount of food (Kg)	Total cost of foods (Tsh)

Figure 3.3: Illustration of GUI computation for not optimal diet

CHAPTER FOUR

General Discussion, Conclusion and Recommendations

4.1. General Discussion

In this study, I formulated, analyzed and presented a cost-effective computing model (mathematical model) for computing optimal amounts of foods at minimized cost that would lead to the achievement of the recommended amount of nutrients for people living with HIV at the early stage of the infection. Also, I have developed a MATLAB based graphical user interface that could act as a tool to be used by clients to compute acceptable amount of foods.

As a result of this study I have developed a mathematical model that blends multiple linear regression and linear programming models which help in diet decision making. The multiple linear regression model capture and predict nutrient requirements in the body by the factors of age, weight, height and gender of an individual. The results from the multiple linear regression models, which are the minimum amount of nutrients required and the maximum amount of nutrients allowed were fed as constraints in the linear programming model.

The two phase simplex method was used to solve the formulated linear programming problem in the MATLAB. Specifically, we wrote the function *TwoPhaseSimplex* for the two phase simplex algorithm in MATLAB in order to solve this problem. This function calls the *totbl* and *lpx* functions which were developed by Ferris and colleagues (Ferris *et al.*, 2007). The function *totbl* display the tableau while the *lpx* is used to exchange the pivot row and pivot column in the simplex algorithm.

The mathematical model developed in chapter 2 was incorporated in the Graphical User Interface to extend usability of our model to individuals. The graphical user interface (GUI) was developed in MATLAB by the graphical user interface development environment (GUIDE) method; the GUIDE method enables the rapid creation of Graphical User Interface in MATLAB programming language. The Graphical User Interface developed has five major parts which include: (1) List of foods you want to choose; this contain a variety of foods that an individual selects. (2) Enter your particulars; this component asks an individual to enter her/his age, weight, height and gender so that to enable the developed mathematical model to capture her/his nutritional requirements in the body. (3) The third component is Enter the cost of foods you have chosen; here the client is asked to submit the cost of each food selected; and this cost should be per kilogram. (4) The fourth component is the Computation part. After the first three parts are well done, the client will need to

press the compute button so as to allow the computation to take place. But also this component contains two extra subcomponents: the status component that states whether the computation is optimal or not optimal, and the second subcomponent which is the reset button, that enables the client to clear the previous inputs and outputs. (5) The last component is the output or “See the results” section which has three sub-components including: index of food, amount of food and total cost of foods. The index of food component indicates the index of food the client will have to buy, the indexes are in the form of $X_1, X_2, X_3, \dots, X_n$, where X_1 stands for first food had been selected, X_2 for the second food selected up to X_n representing the last food choice. The amount of food component indicates the amount of foods with indices in the index component to be bought, and this amount will be in kilograms. The total cost component indicate the total cost that will be incurred to buy the foods stated in the index component, the cost is in Tanzania shillings.

4.2. Conclusion

From this study, we have seen that the multiple linear regression model is capable of predicting the nutrient requirement in the human body by the factors age, weight, height and gender. This is due to the fact that in comparison the values predicted from the model were close enough to the actual RDI values. From the linear programming model an optimal amount and combination of nutrients could be obtained by constraining the linear programming model with outputs from the multiple linear regression models. The optimal solution predicts much less cost compared to the foods suggested by the RDI. I therefore developed a cost-effective computing model for the optimal diet for people living with HIV at an asymptomatic stage. The Graphical User Interface was developed to serve as a computational tool to facilitate the computation of adequate amount of foods to achieve the optimal nutrients.

4.3. Recommendations

From the cost-effective computing model (mathematical model) developed and Graphical User Interface developed in this study, we propose the following further research questions.

- i. Could model predictability be improved through increasing the number of nutrients?
- ii. Could the GUI accessibility be extended by incorporation into web browsers and mobile applications?

References

- Aiken, L.S., West, S.G. and Pitts, S.C. (2003). Multiple linear regression. *Handbook of psychology*
- Akram, D.-S., Bagch, K., Becker, G., Bhandari, N., Bland, R., Brandes, N. and Brazdov, Z. (2003). Cecilia Acuin. *Special Issue Based on a World Health Organization Expert Consultation on Complementary Feeding* 24:130.
- Australian Government Department of Health and Ageing, National Health and Medical Research Council. 2005. Nutrient Reference Values for Australia and New Zealand.
<https://www.nrv.gov.au/> Accessed on 25 May, 2015
- Briend, A., Ferguson, E., and Darmon, N. (2001). Local food price analysis by linear programming: a new approach to assess the economic value of fortified food supplements. *Food and Nutrition Bulletin*, 22(2), 184-189.
- Briend, A., Darmon, N., Ferguson, E., and Erhardt, J. G. (2003). Linear programming: a mathematical tool for analyzing and optimizing children's diets during the complementary feeding period. *Journal of Pediatric Gastroenterology and Nutrition*, 36(1), 12-22.
- Capra, S. (2006). Nutrient reference values for Australia and New Zealand: Including recommended dietary intakes. Commonwealth of Australia,
- Chatterjee, S. and Hadi, A. S. (2013). Regression analysis by example. John Wiley & Sons,
- Clark, R.A., Maupin Jr, R. T., Hayes, J. and Hammer, J. H. (2012). A woman's guide to living with HIV infection. JHU press,
- Czyzyk, J. and Wisniewski, T. J. (1996). The Diet Problem: A WWW-based Interactive Case Study in Linear Programming. *Mathematics and Computer Science Division, Argonne National Laboratory*, <http://citeseerx.ist.psu.edu/viewdoc/summary>.
- Czyzyk, J., Wisniewski, T. and Wright, S. J. (1999). Optimization case studies in the NEOS guide. *SIAM review*, 41(1), 148-163.
- Darmon, N., Ferguson, E., and Briend, A. (2002). Linear and nonlinear programming to optimize the nutrient density of a population's diet: an example based on diets of preschool children in rural Malawi. *The American journal of clinical nutrition*, 75(2), 245-253.
- Fawzi, W. W. and Hunter, D. J. (1998). Vitamins in HIV disease progression and vertical transmission. *Epidemiology* 9(4): 457-466.
- Ferris, M. C., Mangasarian, O. L. and Wright, S. J. (2007). *Linear programming with MATLAB*
<http://research.cs.wisc.edu/math-prog/lpbook/> Accessed on May 28 2015 at 18:12

- Ferris, M. C., Mangasarian, O. L. and Wright, S. J. (2007). *Linear programming with MATLAB* (Vol. 7). SIAM.
- Food and Agriculture Organization (FAO) and World Health Organization (WHO).2002. Living well with HIV/AIDS – a manual on nutritional care and support for people living with HIV/AIDS. <http://www.fao.org/docrep/005/y4168e/y4168e06.htm> Accessed on May 26 2015
- Guruhappa, H., Sree, S., Madhu, C. and Srivastava, K. (2013). A Graphical User Interface (GUI) in MATLAB to Compute the Thermal Lithospheric Thickness and its Error Bounds. *interface* 10:11.
- Hunt, B. R., Lipsman, R. L. and Rosenberg, J. M. (2014). A guide to MATLAB: for beginners and experienced users. Cambridge University Press,
- Kumar, A., Kamboj, D., Choudhary, J., Yadav, N. and Batra, V. (2013). GUI Based Device Controller Using MATLAB.
- Langseth, L. (1999). Nutrition and Immunity in man, International Life Science Institute, Europe.
- Lent, C. S. (2013). Learning to Program with MATLAB: Building GUI Tools: Building GUI Tools. Wiley Global Education,
- Mbabazi, D. (2008). Population Dynamic Type Models in HIV Infections, African Institute for Mathematical Sciences (AIMS).
- Osborne, J. and Waters, E. (2002). Four assumptions of multiple regression that researchers should always test. *Practical assessment, research & evaluation* 8:1-9.
- Pasic, M., Catovic, A., Bijelonja, I. and Crnovrsanin, S. (2011). Linear Programming Local Cost Nutrition Optimization Model. In Annals of DAAAM for 2011 & Proceedings of the 22nd International DAAAM Symposium. 0389-0390.
- Poole, M. A. and O'Farrell, P.N. (1971). The assumptions of the linear regression model. *Transactions of the Institute of British Geographers* 145-158.
- Scientific Advisory Committee on Nutrition (SACN).2011. Dietary Reference Values for Energy. <https://www.sacn.gov.uk> Accessed on May 25 2015
- Seber, G.A. and Lee, A.J. (2012). Linear regression analysis. John Wiley & Sons,
- Smith, V. E. (1959). Linear programming models for the determination of palatable human diets. *Journal of Farm Economics*, 272-283.

Sufahani, S. and Ismail, Z. (2014). A New Menu Planning Model for Malaysian Secondary Schools using Optimization Approach. *Applied Mathematical Sciences* 8:7511-7518.

United Nations University and World Health Organization. (2004). *Human Energy Requirements: Report of a Joint FAO/WHO/UNU Expert Consultation: Rome, 17-24 October 2001* (No. 1). Food & Agriculture Org.

Tanzania Food and Nutrition Center (TFNC). (2009). National guidelines for nutrition care and support for people living with hiv In Tanzania Food and Nutrition Center, Dar es Salaam.

Tranmer, M. and Elliot, M. (2008). Multiple linear regression. *The Cathie Marsh Centre for Census and Survey Research (CCSR)*.

Weisberg, S. (2005). Applied linear regression. John Wiley & Sons,

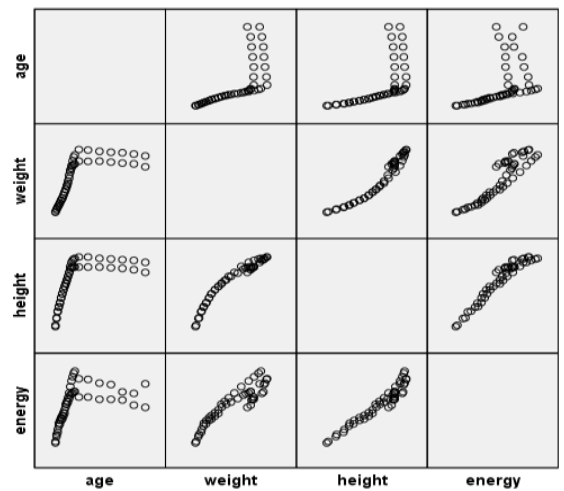
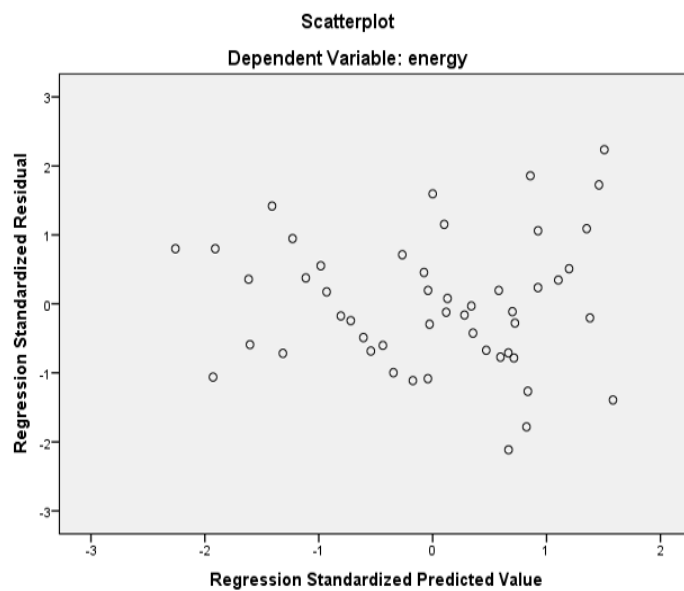
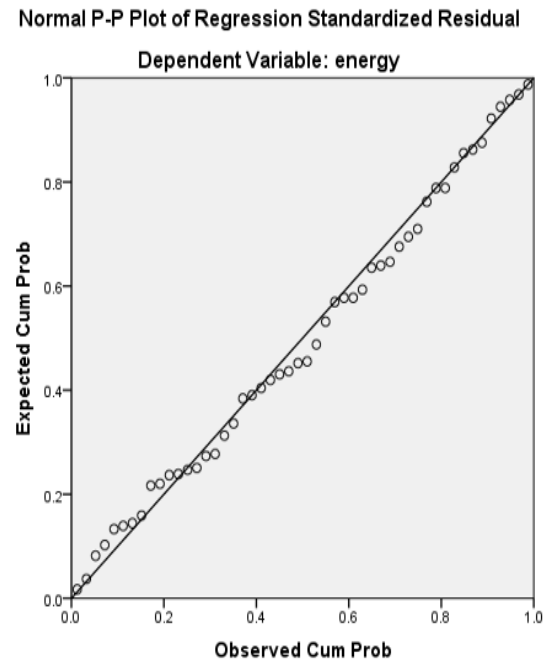
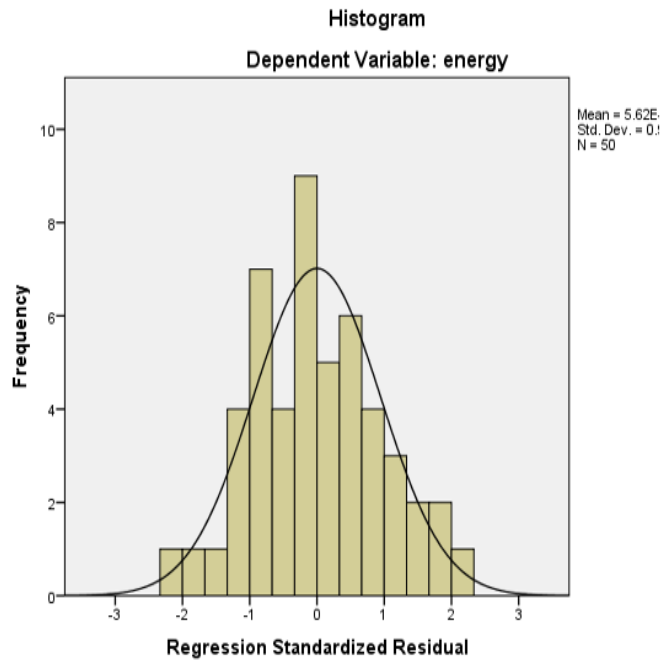
Weisberg, S. (2014). Applied linear regression. John Wiley & Sons,

Williams, M.N., Grajales, C. A. G and Kurkiewicz, D. (2013). Assumptions of multiple regression: correcting two misconceptions. *Pract. Assess. Res. Eval* 18:

World Health Organization (WHO). (2007). Nutrient requirements for people living with HIV/AIDS: report of a technical consultation; 2003. In.

Appendices

Appendix 1: summary of multiple linear regression analysis results in graphs and tables for energy



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.992 ^a	.984	.982	91.87955

a. Predictors: (Constant), gender, age_cube, height, weight, age

b. Dependent Variable: energy

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	22464786.48	5	4492957.296	532.224	.000 ^b
	Residual	371441.472	44	8441.852		
	Total	22836227.95	49			

a. Dependent Variable: energy

b. Predictors: (Constant), gender, age_cube, height, weight, age

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-753.860	142.343		-5.296	.000	-1040.734	-466.985
	age	-19.773	2.999	-.623	-6.594	.000	-25.816	-13.729
	age_cube	.002	.000	.323	4.396	.000	.001	.003
	weight	14.399	3.102	.428	4.642	.000	8.148	20.649
	height	18.367	1.710	.805	10.739	.000	14.920	21.814
	gender	178.263	26.462	.132	6.737	.000	124.933	231.593

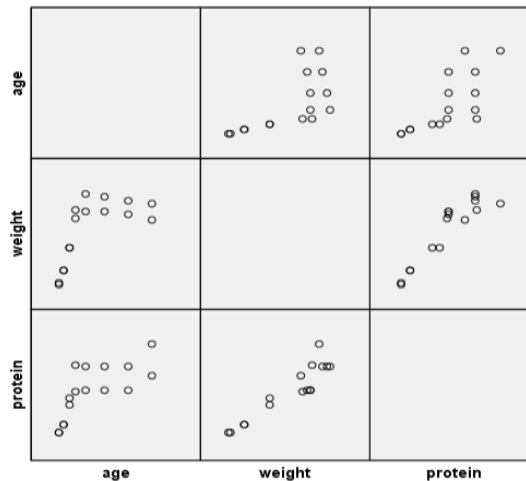
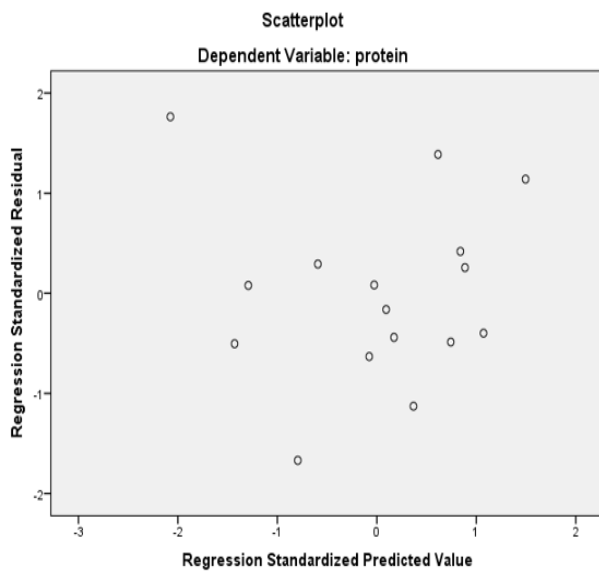
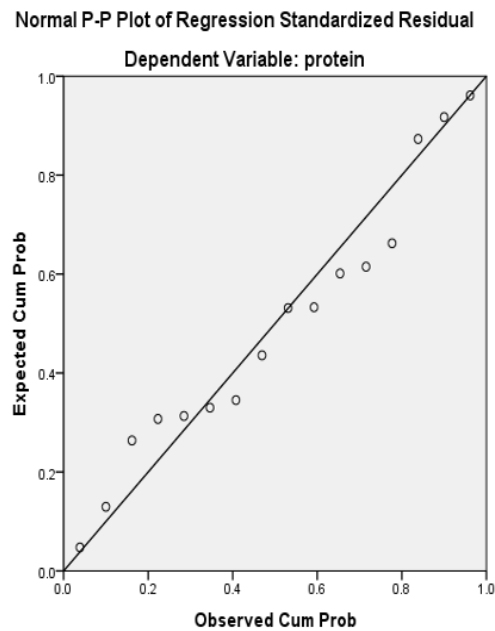
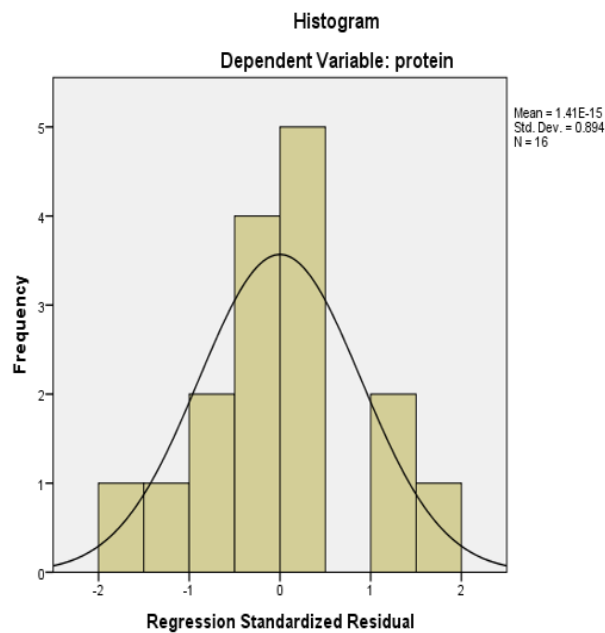
a. Dependent Variable: energy

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	715.1306	3316.2107	2244.4372	677.10046	50
Residual	-194.28136	205.45631	.00000	87.06571	50
Std. Predicted Value	-2.259	1.583	.000	1.000	50
Std. Residual	-2.115	2.236	.000	.948	50

a. Dependent Variable: energy

Appendix 2: summary of multiple linear regression analysis results in graphs and tables for protein



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.969 ^a	.939	.924	5.62055

a. Predictors: (Constant), gender, age_cube, log_weight

b. Dependent Variable: protein

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	5843.851	3	1947.950	61.662	.000 ^b
	Residual	379.087	12	31.591		
	Total	6222.938	15			

a. Dependent Variable: protein

b. Predictors: (Constant), gender, age_cube, log_weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-58.621	9.290		-6.310	.000	-78.861	-38.381
	age_cube	3.028E-005	.000	.266	3.434	.005	.000	.000
	log_weight	25.676	2.563	.778	10.020	.000	20.093	31.260
	gender	10.193	2.823	.258	3.611	.004	4.043	16.344

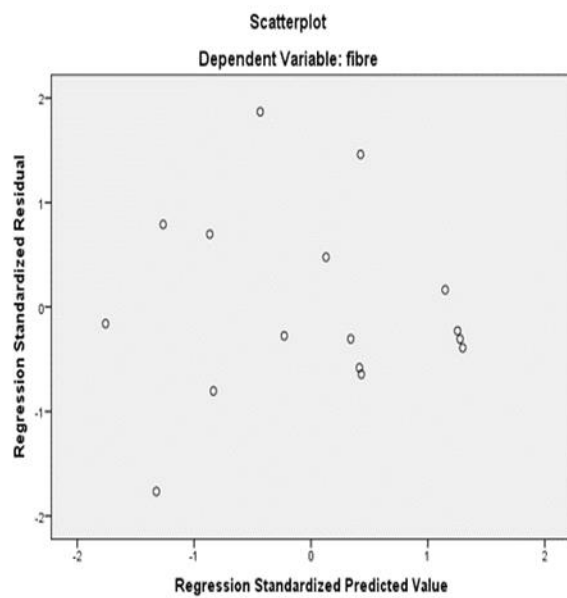
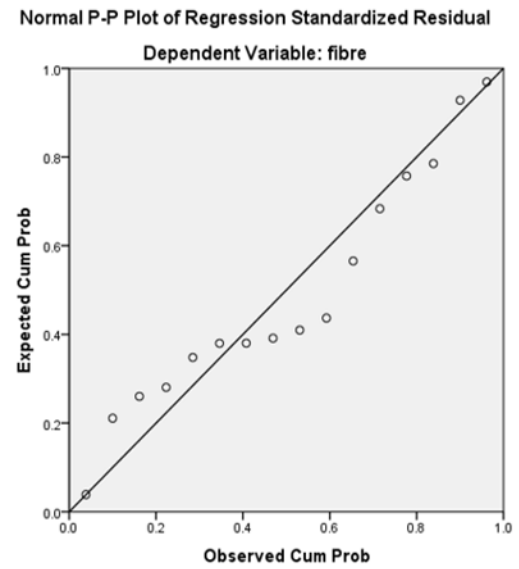
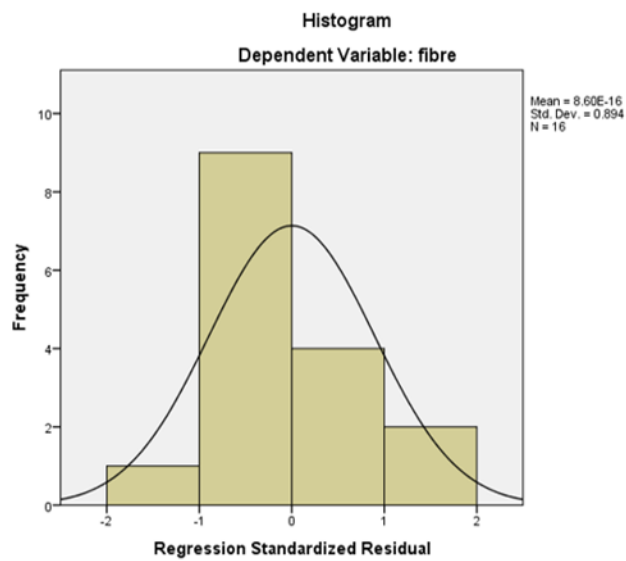
a. Dependent Variable: protein

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	4.0892	74.5878	45.0625	19.73804	16
Residual	-9.38100	9.91078	.00000	5.02717	16
Std. Predicted Value	-2.076	1.496	.000	1.000	16
Std. Residual	-1.669	1.763	.000	.894	16

a. Dependent Variable: protein

Appendix 3: summary of multiple linear regression analysis results in graphs and tables for fiber



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.972 ^a	.945	.931	1.44377

a. Predictors: (Constant), gender, log_age, weight_cube

b. Dependent Variable: fibre

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	428.736	3	142.912	68.560	.000 ^b
	Residual	25.014	12	2.084		
	Total	453.750	15			

a. Dependent Variable: fibre

b. Predictors: (Constant), gender, log_age, weight_cube

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	12.641	1.195		10.577	.000	10.037	15.245
	log_age	2.247	.542	.489	4.145	.001	1.066	3.428
	weight_cube	2.086E-005	.000	.480	3.906	.002	.000	.000
	gender	2.309	.811	.217	2.848	.015	.542	4.075

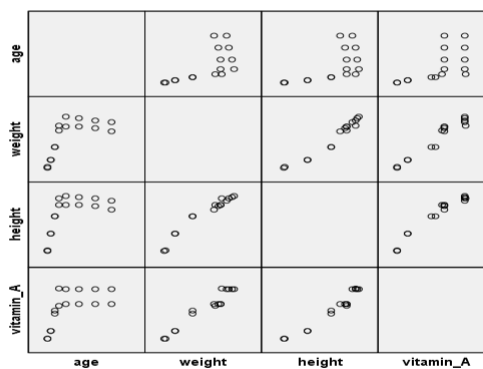
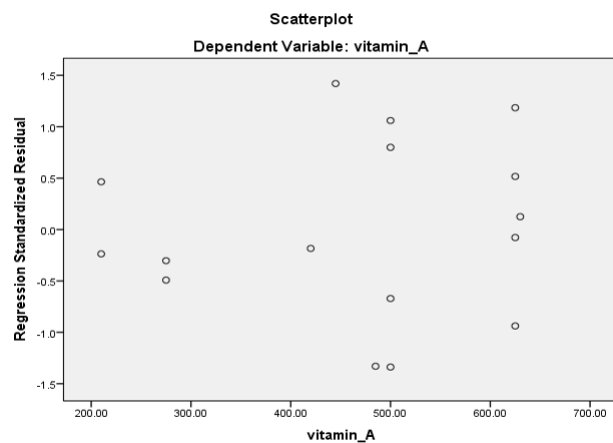
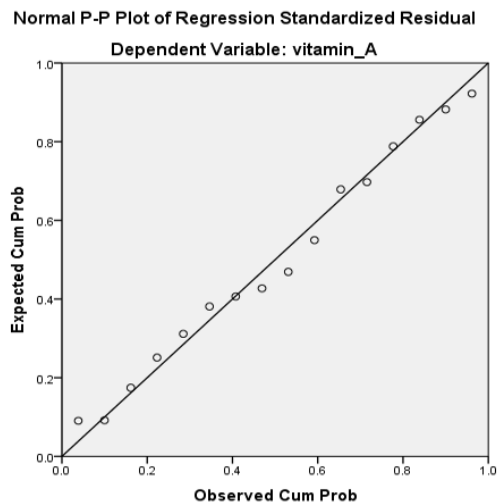
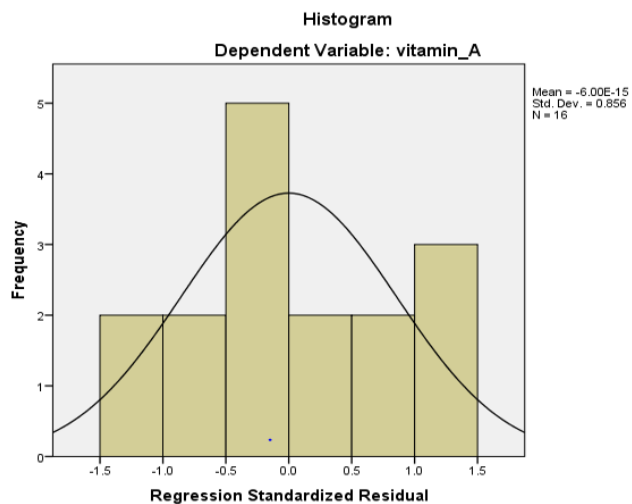
a. Dependent Variable: fibre

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	14.2298	30.5644	23.6250	5.34625	16
Residual	-2.54963	2.69791	.00000	1.29135	16
Std. Predicted Value	-1.757	1.298	.000	1.000	16
Std. Residual	-1.766	1.869	.000	.894	16

a. Dependent Variable: fibre

Appendix 4: summary of multiple linear regression analysis results in graphs and tables for vitamin A



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.996 ^a	.992	.989	15.58836

a. Predictors: (Constant), log_age, height_square, weight, height

b. Dependent Variable: vitamin_A

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	336920.783	4	84230.196	346.631	.000 ^b
	Residual	2672.967	11	242.997		
	Total	339593.750	15			

a. Dependent Variable: vitamin_A

b. Predictors: (Constant), log_age, height_square, weight, height

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	912.549	146.757		6.218	.000	589.540	1235.558
	weight	-9.815	2.074	-1.375	-4.732	.001	-14.381	-5.250
	height	-15.252	2.588	-3.117	-5.894	.000	-20.947	-9.556
	height_square	.093	.013	5.076	7.112	.000	.064	.122
	log_age	53.706	11.360	.427	4.728	.001	28.704	78.708

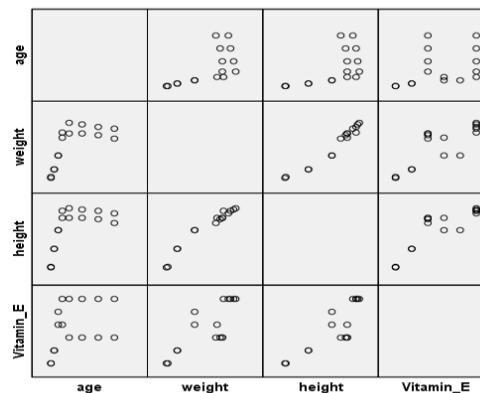
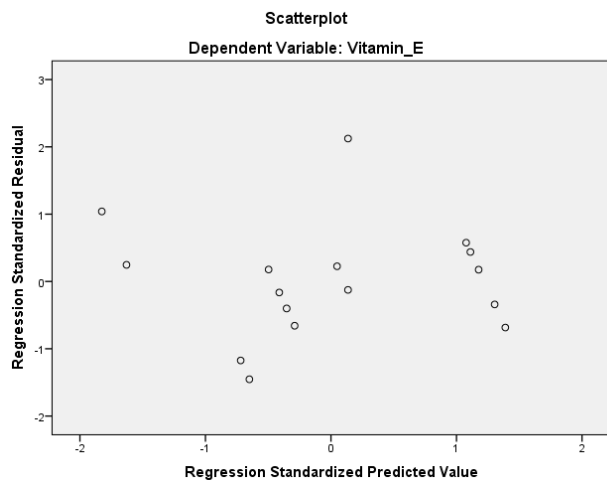
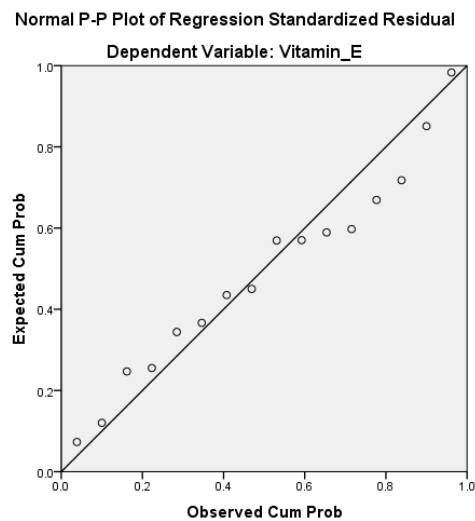
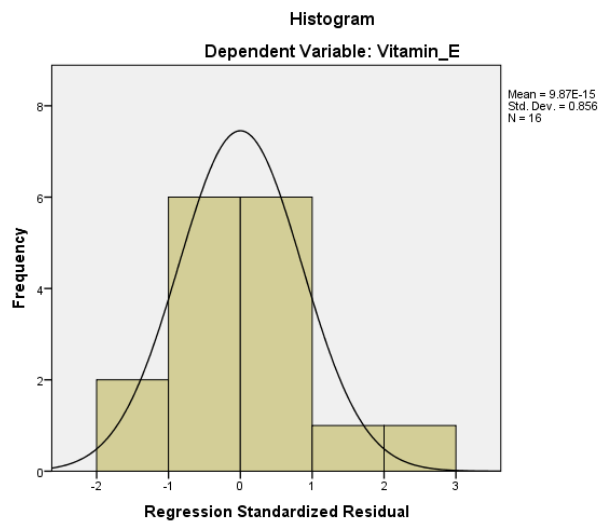
a. Dependent Variable: vitamin_A

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	202.7651	639.6144	465.6250	149.87123	16
Residual	-20.85366	22.13431	.00000	13.34907	16
Std. Predicted Value	-1.754	1.161	.000	1.000	16
Std. Residual	-1.338	1.420	.000	.856	16

a. Dependent Variable: vitamin_A

Appendix 5: summary of multiple linear regression analysis results in graphs and tables for vitamin E



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.978 ^a	.957	.941	.44465

a. Predictors: (Constant), weight_cube, age_cube, height_square, weight

b. Dependent Variable: Vitamin_E

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	48.263	4	12.066	61.026	.000 ^b
	Residual	2.175	11	.198		
	Total	50.438	15			

a. Dependent Variable: Vitamin_E

b. Predictors: (Constant), weight_cube, age_cube, height_square, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	2.762	.508		5.442	.000	1.645	3.880
	weight	-.444	.058	-.5107	-7.613	.000	-.573	-.316
	height_square	.001	.000	.4359	9.023	.000	.001	.001
	age_cube	2.930E-006	.000	.286	3.567	.004	.000	.000
	weight_cube	2.260E-005	.000	1.561	5.986	.000	.000	.000

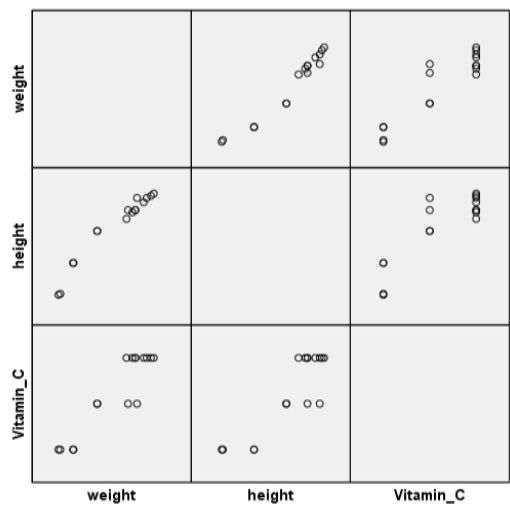
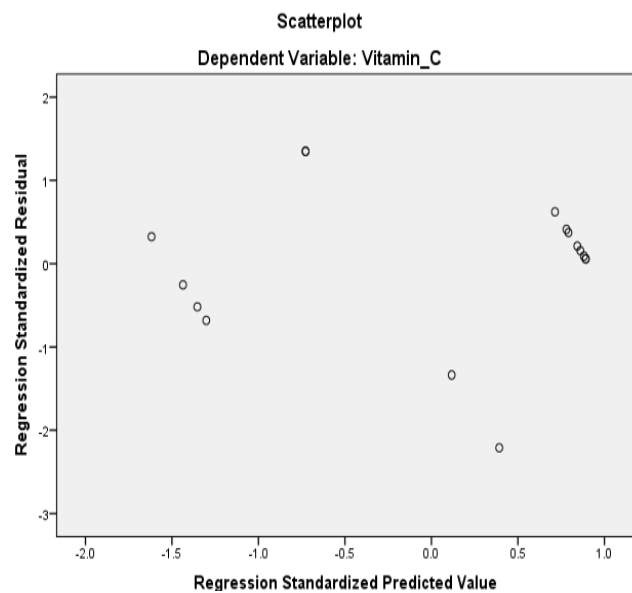
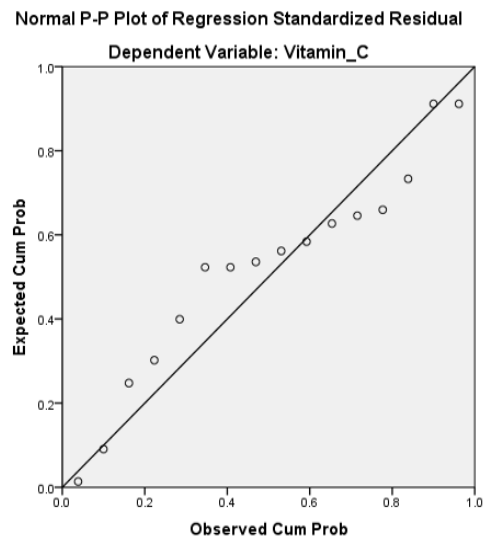
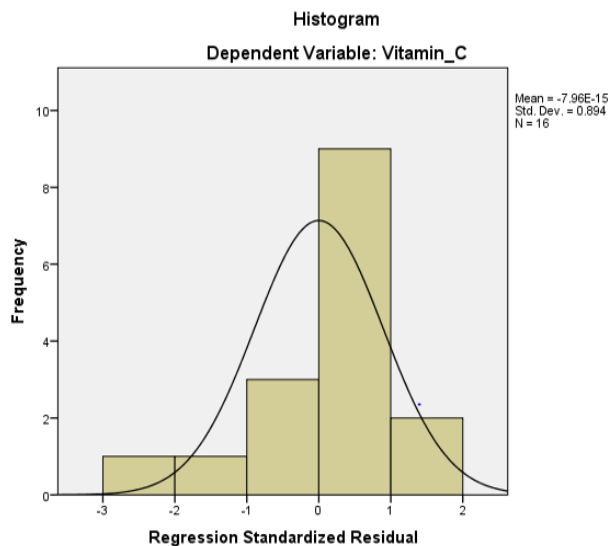
a. Dependent Variable: Vitamin_E

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	4.5375	10.3043	7.8125	1.79374	16
Residual	-.64634	.94422	.00000	.38078	16
Std. Predicted Value	-1.826	1.389	.000	1.000	16
Std. Residual	-1.454	2.124	.000	.856	16

a. Dependent Variable: Vitamin_E

Appendix 6: summary of multiple linear regression analysis results in graphs and tables for vitamin C



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.963 ^a	.927	.908	1.29606

a. Predictors: (Constant), height_square, weight_square, weight

b. Dependent Variable: Vitamin_C

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	254.843	3	84.948	50.571	.000 ^b
	Residual	20.157	12	1.680		
	Total	275.000	15			

a. Dependent Variable: Vitamin_C

b. Predictors: (Constant), height_square, weight_square, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	32.492	1.685		19.287	.000	28.821	36.162
	weight	.850	.196	4.183	4.332	.001	.422	1.278
	weight_square	-.003	.001	-1.436	-2.547	.026	-.006	-.001
	height_square	-.001	.000	-1.872	-3.432	.005	-.002	.000

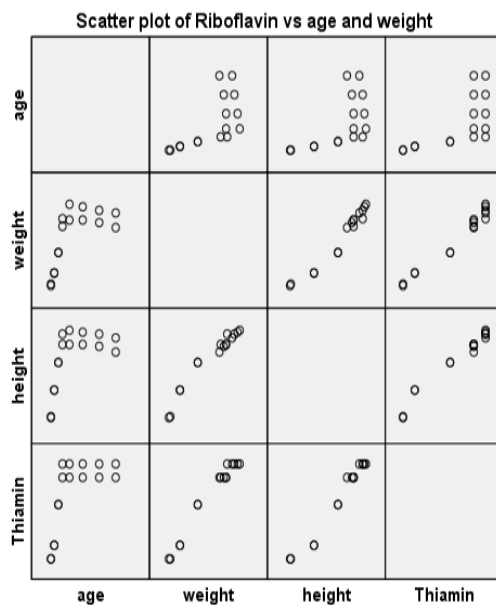
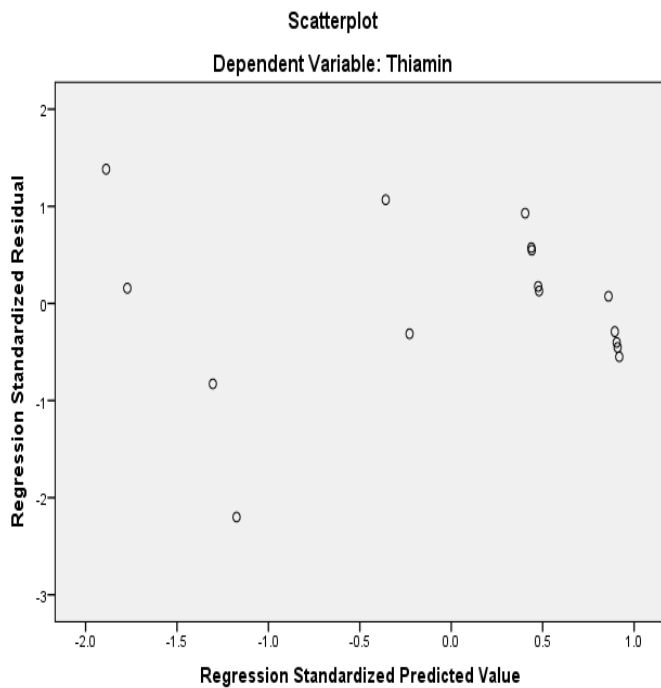
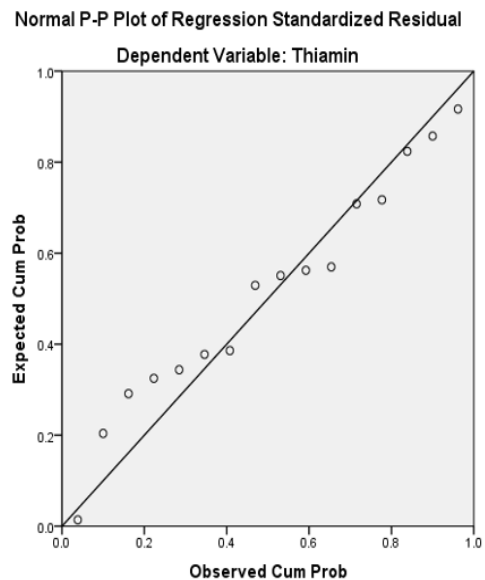
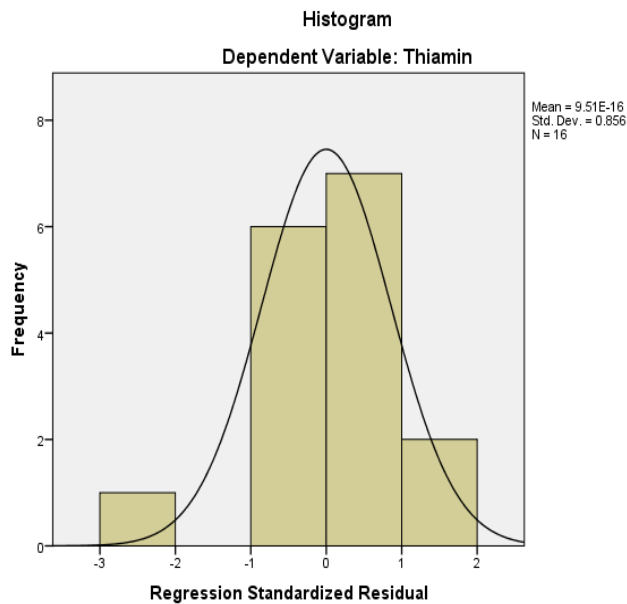
a. Dependent Variable: Vitamin_C

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	34.5804	44.9254	41.2500	4.12183	16
Residual	-2.86559	1.74893	.00000	1.15923	16
Std. Predicted Value	-1.618	.892	.000	1.000	16
Std. Residual	-2.211	1.349	.000	.894	16

a. Dependent Variable: Vitamin_C

Appendix 7: summary of multiple linear regression analysis results in graphs and tables for Thiamin



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.997 ^a	.993	.991	.02521

a. Predictors: (Constant), gender, age_cube, height_cube, weight_cube

b. Dependent Variable: Thiamin

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.067	4	.267	419.809	.000 ^b
	Residual	.007	11	.001		
	Total	1.074	15			

a. Dependent Variable: Thiamin

b. Predictors: (Constant), gender, age_cube, height_cube, weight_cube

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	.380	.022		17.251	.000	.331	.428
	weight_cube	-4.845E-007	.000	-.229	-3.004	.012	.000	.000
	age_cube	1.860E-007	.000	.124	4.761	.001	.000	.000
	height_cube	1.844E-007	.000	1.175	15.889	.000	.000	.000
	gender	-.035	.013	-.067	-2.597	.025	-.064	-.005

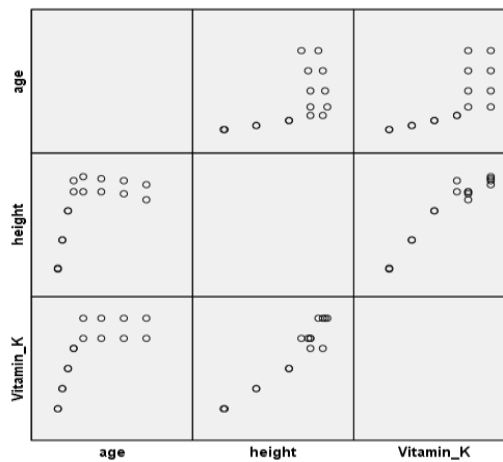
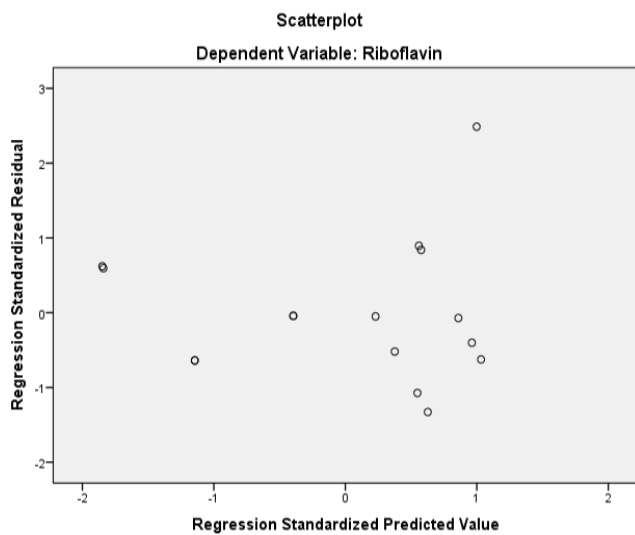
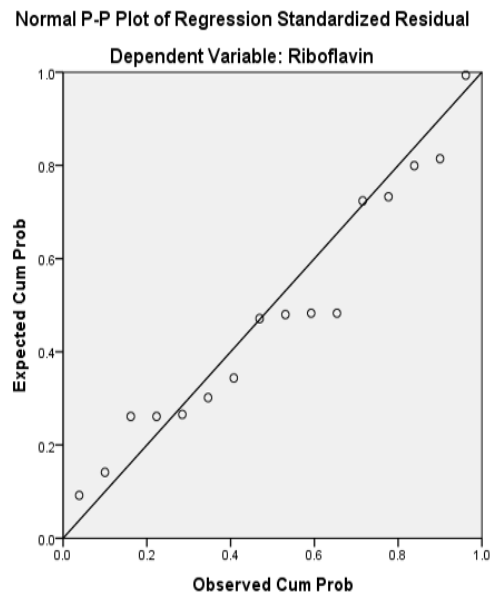
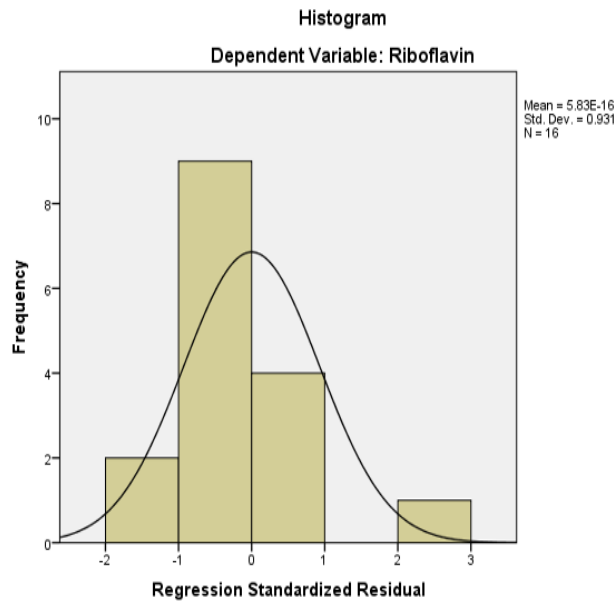
a. Dependent Variable: Thiamin

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	.4651	1.2139	.9688	.26676	16
Residual	-.05547	.03485	.00000	.02159	16
Std. Predicted Value	-1.888	.919	.000	1.000	16
Std. Residual	-2.200	1.382	.000	.856	16

a. Dependent Variable: Thiamin

Appendix 8: summary of multiple linear regression analysis results in graphs and tables for Riboflavin



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.961 ^a	.923	.911	.09981

a. Predictors: (Constant), log_age, height_cube

b. Dependent Variable: Riboflavin

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.545	2	.772	77.542	.000 ^b
	Residual	.129	13	.010		
	Total	1.674	15			

a. Dependent Variable: Riboflavin

b. Predictors: (Constant), log_age, height_cube

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	.286	.067		4.274	.001	.142	.431
	height_cube	1.186E-007	.000	.606	4.285	.001	.000	.000
	log_age	.110	.039	.394	2.788	.015	.025	.195

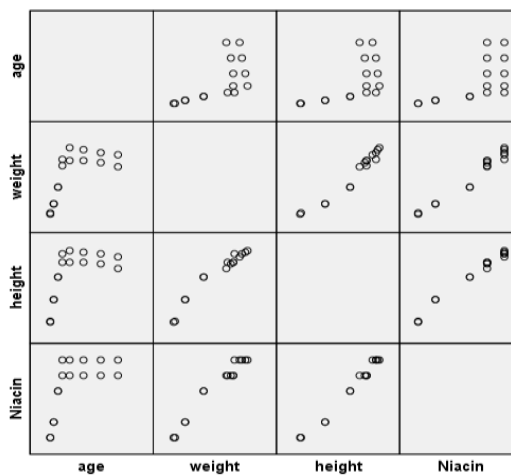
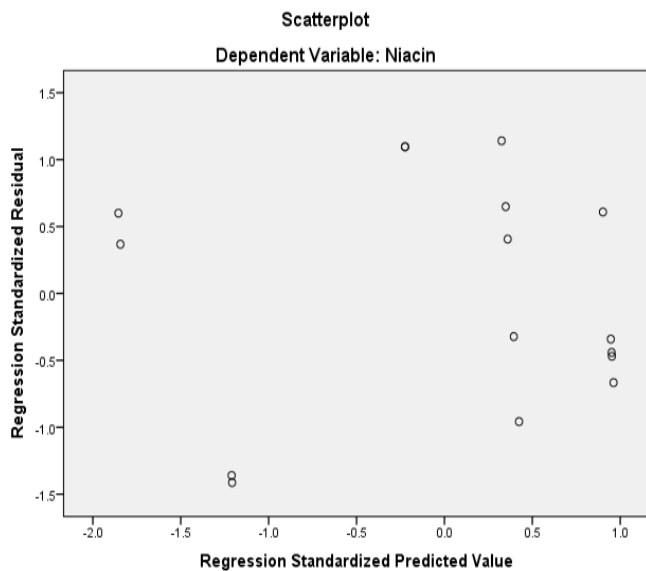
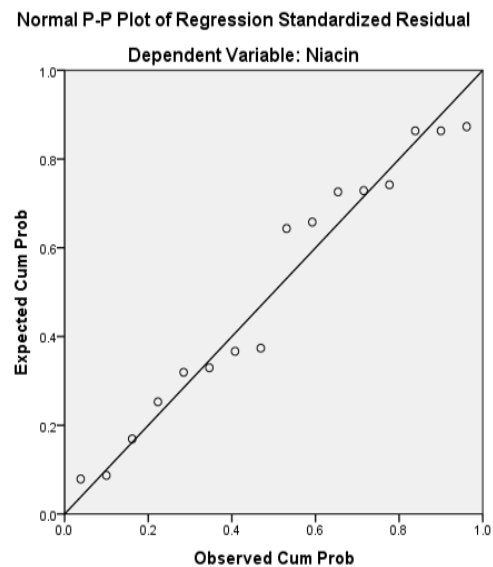
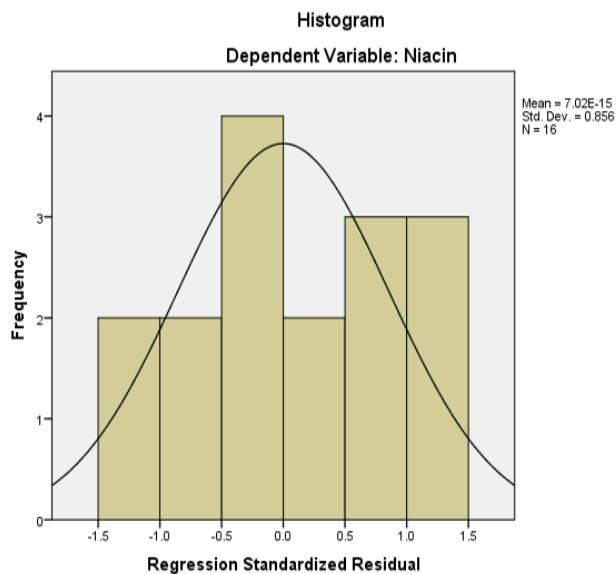
a. Dependent Variable: Riboflavin

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	.4380	1.3625	1.0313	.32092	16
Residual	-.13261	.24823	.00000	.09292	16
Std. Predicted Value	-1.849	1.032	.000	1.000	16
Std. Residual	-1.329	2.487	.000	.931	16

a. Dependent Variable: Riboflavin

Appendix 9: summary of multiple linear regression analysis results in graphs and tables for Niacin



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.999 ^a	.998	.998	.17141

a. Predictors: (Constant), age_cube, height_square, weight_square, height

b. Dependent Variable: Niacin

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	197.427	4	49.357	1679.814	.000 ^b
	Residual	.323	11	.029		
	Total	197.750	15			

a. Dependent Variable: Niacin

b. Predictors: (Constant), age_cube, height_square, weight_square, height

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	8.002	1.594		5.021	.000	4.495	11.510
	height	-.107	.029	-.905	-3.655	.004	-.171	-.043
	weight_square	-.001	.000	-.353	-4.549	.001	-.001	.000
	height_square	.001	.000	2.194	7.093	.000	.001	.001
	age_cube	2.434E-006	.000	.120	8.121	.000	.000	.000

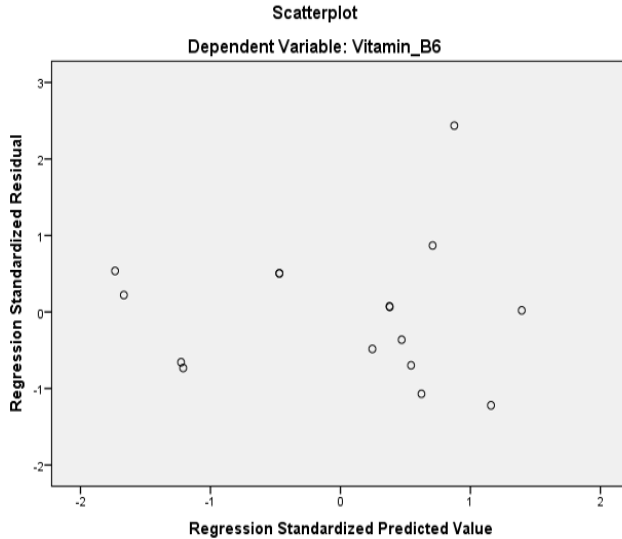
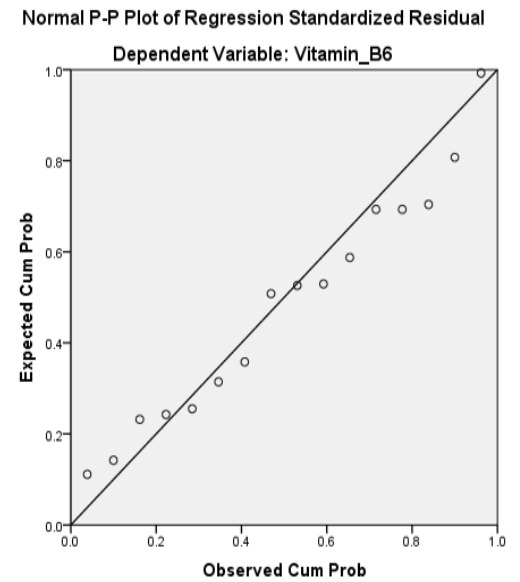
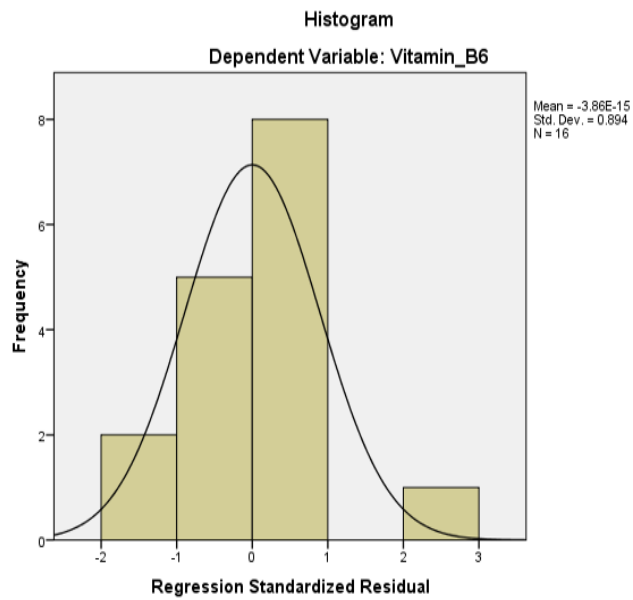
a. Dependent Variable: Niacin

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	5.8970	16.1141	12.6250	3.62792	16
Residual	-.24216	.19561	.00000	.14679	16
Std. Predicted Value	-1.855	.962	.000	1.000	16
Std. Residual	-1.413	1.141	.000	.856	16

a. Dependent Variable: Niacin

Appendix 10: summary of multiple linear regression analysis results in graphs and tables for Vitamin B6



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.982 ^a	.964	.955	.08545

a. Predictors: (Constant), weight, age_cube, weight_cube

b. Dependent Variable: Vitamin_B6

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2.372	3	.791	108.267	.000 ^b
	Residual	.088	12	.007		
	Total	2.459	15			

a. Dependent Variable: Vitamin_B6

b. Predictors: (Constant), weight, age_cube, weight_cube

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	.192	.087		2.208	.047	.003	.381
	weight_cube	-1.334E-006	.000	-.417	-2.238	.045	.000	.000
	age_cube	7.226E-007	.000	.319	5.340	.000	.000	.000
	weight	.023	.004	1.197	6.283	.000	.015	.031

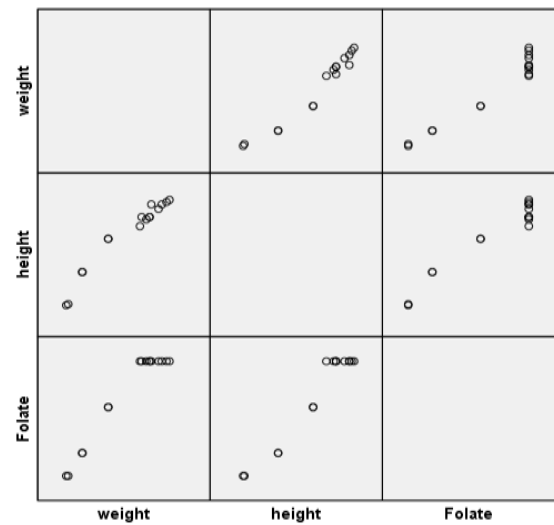
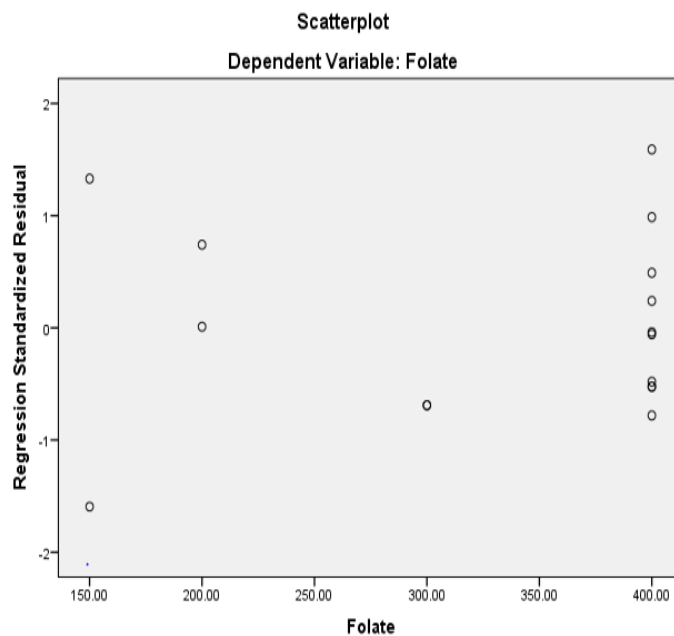
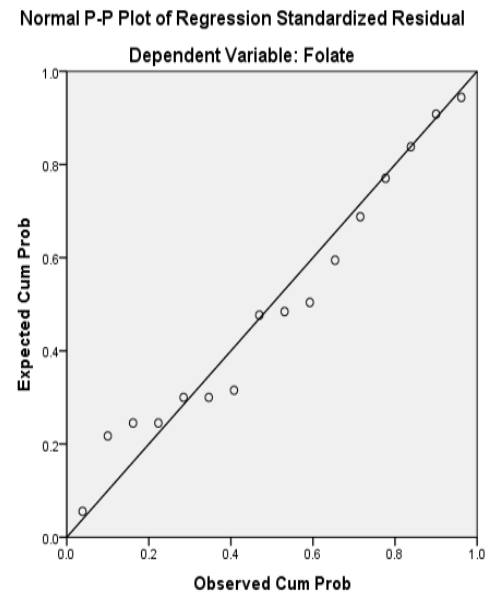
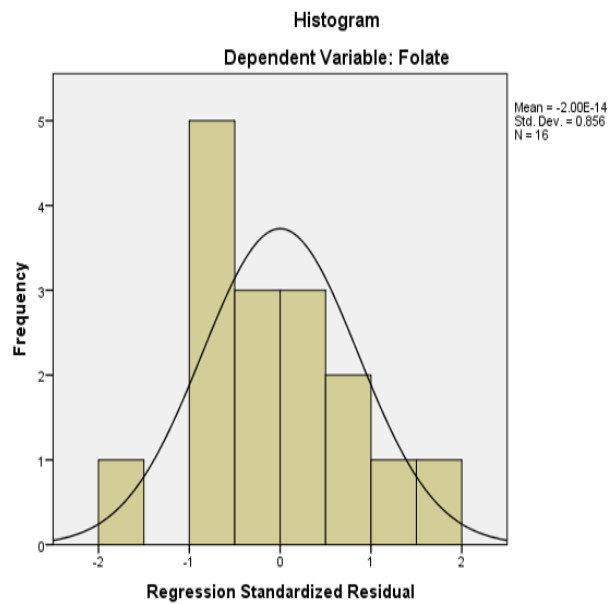
a. Dependent Variable: Vitamin_B6

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	.4542	1.6983	1.1438	.39764	16
Residual	-.10431	.20810	.00000	.07643	16
Std. Predicted Value	-1.734	1.395	.000	1.000	16
Std. Residual	-1.221	2.435	.000	.894	16

a. Dependent Variable: Vitamin_B6

Appendix 11: summary of multiple linear regression analysis results in graphs and tables for Folate



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.999 ^a	.998	.998	4.94189

a. Predictors: (Constant), height_square, weight_square, weight, height

b. Dependent Variable: Folate

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	149106.354	4	37276.589	1526.332	.000 ^b
	Residual	268.646	11	24.422		
	Total	149375.000	15			

a. Dependent Variable: Folate

b. Predictors: (Constant), height_square, weight_square, weight, height

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	303.834	57.003		5.330	.000	178.372	429.296
	height	-5.220	1.071	-1.609	-4.876	.000	-7.577	-2.864
	weight	17.576	1.058	3.711	16.612	.000	15.247	19.904
	weight_square	-.133	.011	-2.345	-12.622	.000	-.156	-.110
	height_square	.014	.004	1.156	3.409	.006	.005	.023

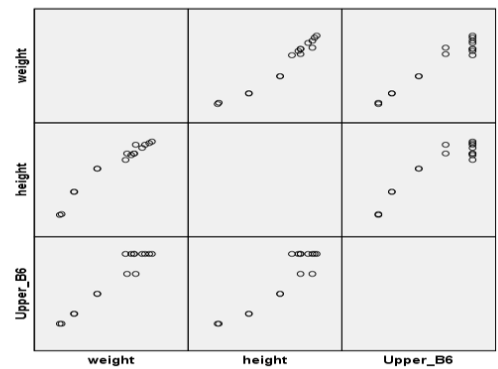
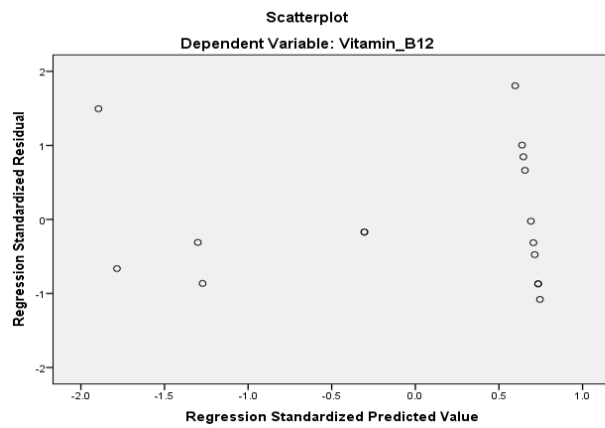
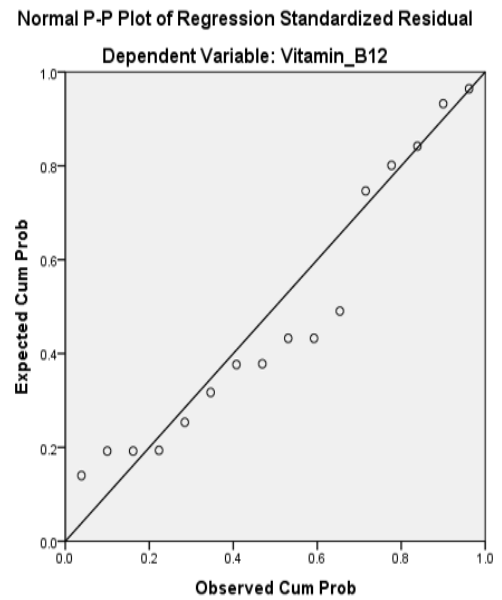
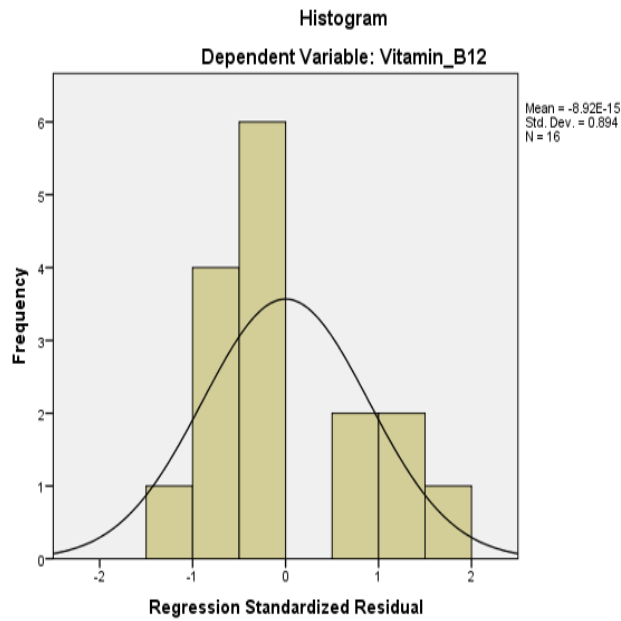
a. Dependent Variable: Folate

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	143.4329	403.8613	331.2500	99.70167	16
Residual	-7.87781	7.85271	.00000	4.23199	16
Std. Predicted Value	-1.884	.728	.000	1.000	16
Std. Residual	-1.594	1.589	.000	.856	16

a. Dependent Variable: Folate

Appendix 12: summary of multiple linear regression analysis results in graphs and tables for Vitamin B12



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.999 ^a	.998	.997	.03069

a. Predictors: (Constant), weight_cube, height, weight

b. Dependent Variable: Vitamin_B12

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	5.366	3	1.789	1899.427	.000 ^b
	Residual	.011	12	.001		
	Total	5.378	15			

a. Dependent Variable: Vitamin_B12

b. Predictors: (Constant), weight_cube, height, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	.696	.107		6.499	.000	.463	.929
	weight	.063	.004	2.200	17.394	.000	.055	.070
	height	-.006	.002	-.331	-4.151	.001	-.010	-.003
	weight_cube	-4.486E-006	.000	-.949	-15.338	.000	.000	.000

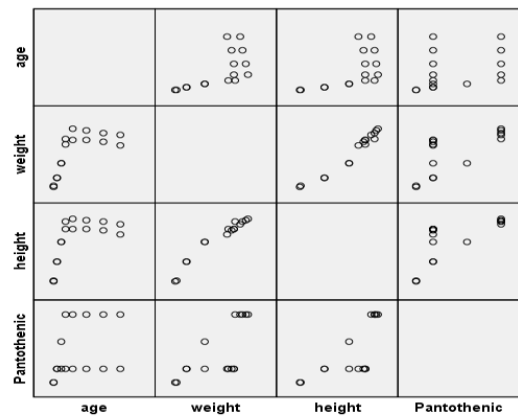
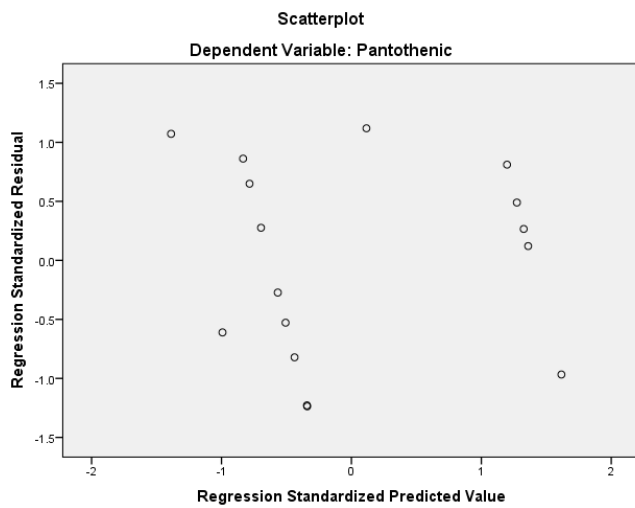
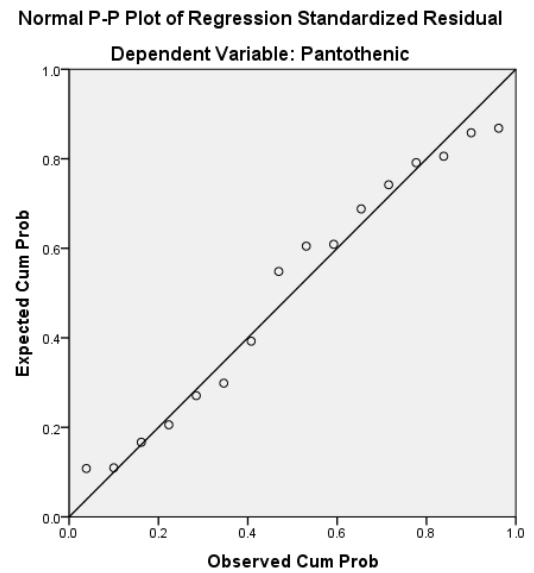
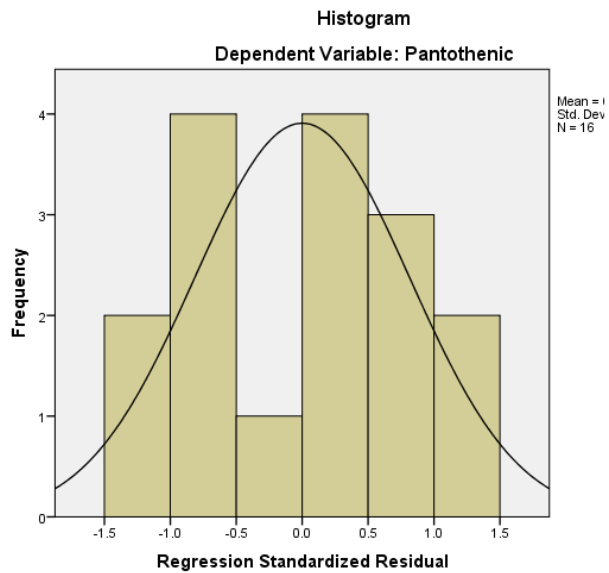
a. Dependent Variable: Vitamin_B12

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	.8542	2.4332	1.9875	.59812	16
Residual	-.03315	.05540	.00000	.02745	16
Std. Predicted Value	-1.895	.745	.000	1.000	16
Std. Residual	-1.080	1.805	.000	.894	16

a. Dependent Variable: Vitamin_B12

Appendix 13: summary of multiple linear regression analysis results in graphs and tables for Pantothenic



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.982 ^a	.964	.947	.23293

a. Predictors: (Constant), gender, age_cube, height_cube, weight_cube, weight_square

b. Dependent Variable: Pantothenic

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	14.707	5	2.941	54.214	.000 ^b
	Residual	.543	10	.054		
	Total	15.250	15			

a. Dependent Variable: Pantothenic

b. Predictors: (Constant), gender, age_cube, height_cube, weight_cube, weight_square

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	2.649	.204		12.967	.000	2.194	3.105
	age_cube	1.680E-006	.000	.298	3.611	.005	.000	.000
	weight_cube	3.925E-005	.000	4.929	5.031	.001	.000	.000
	weight_square	-.004	.001	-7.114	-5.096	.000	-.006	-.002
	height_cube	1.698E-006	.000	2.873	5.944	.000	.000	.000
	gender	.451	.179	.231	2.518	.030	.052	.851

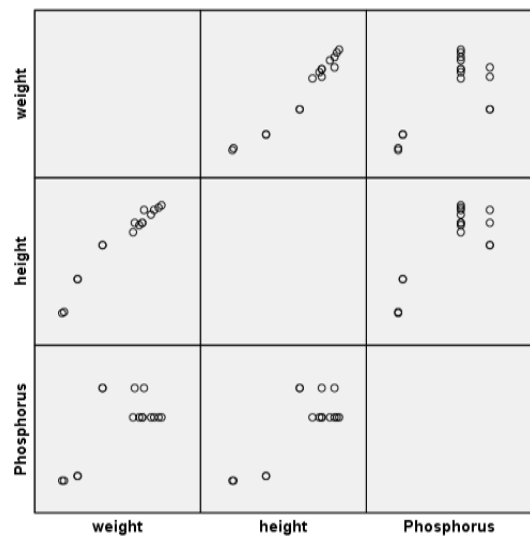
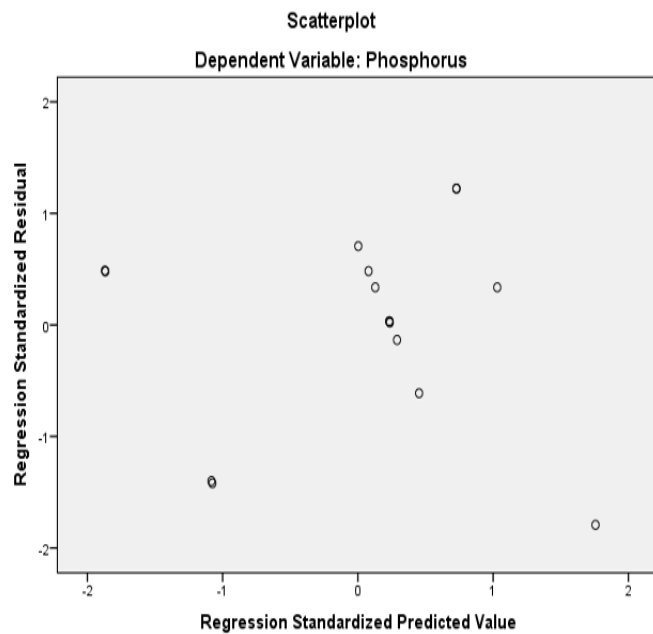
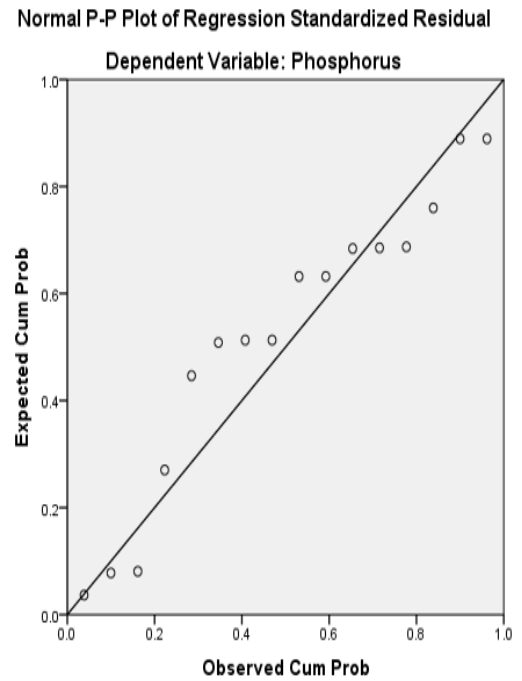
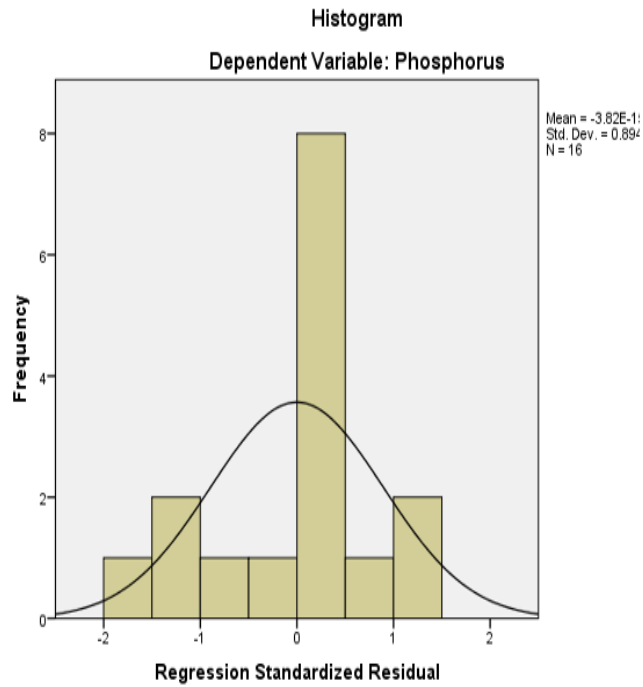
a. Dependent Variable: Pantothenic

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	3.2504	6.2253	4.6250	.99020	16
Residual	-.28811	.26056	.00000	.19019	16
Std. Predicted Value	-1.388	1.616	.000	1.000	16
Std. Residual	-1.237	1.119	.000	.816	16

a. Dependent Variable: Pantothenic

Appendix 14: summary of multiple linear regression analysis results in graphs and tables for Phosphorus



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.956 ^a	.915	.894	94.54595

a. Predictors: (Constant), height, weight_cube, height_square

b. Dependent Variable: Phosphorus

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1153032.764	3	384344.255	42.997	.000 ^b
	Residual	107267.236	12	8938.936		
	Total	1260300.000	15			

a. Dependent Variable: Phosphorus

b. Predictors: (Constant), height, weight_cube, height_square

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	2169.802	891.433		2.434	.031	227.536	4112.067
	weight_cube	-.005	.001	-.2110	-5.518	.000	-.007	-.003
	height_square	.253	.070	7.166	3.605	.004	.100	.406
	height	-42.095	15.941	-4.466	-2.641	.022	-76.829	-7.362

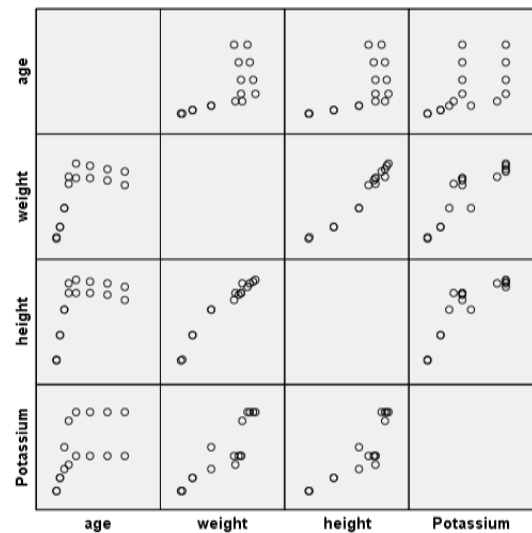
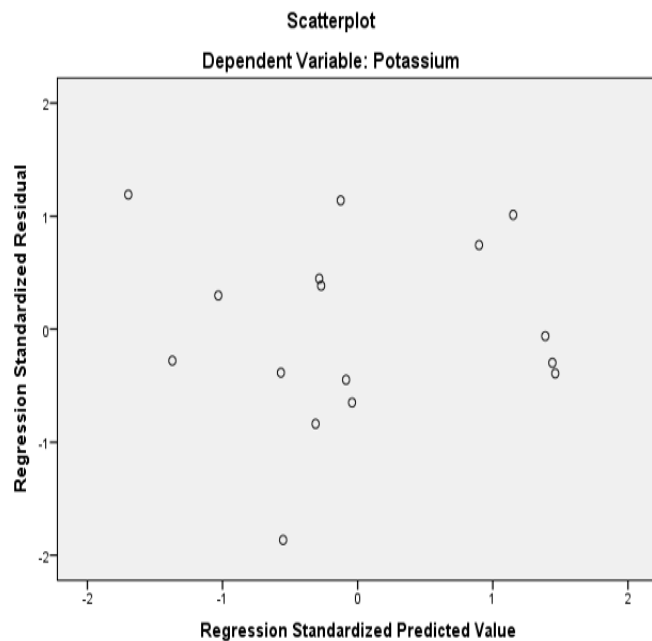
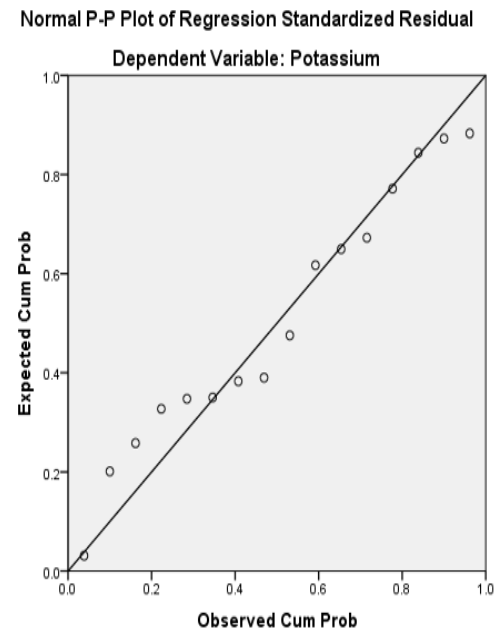
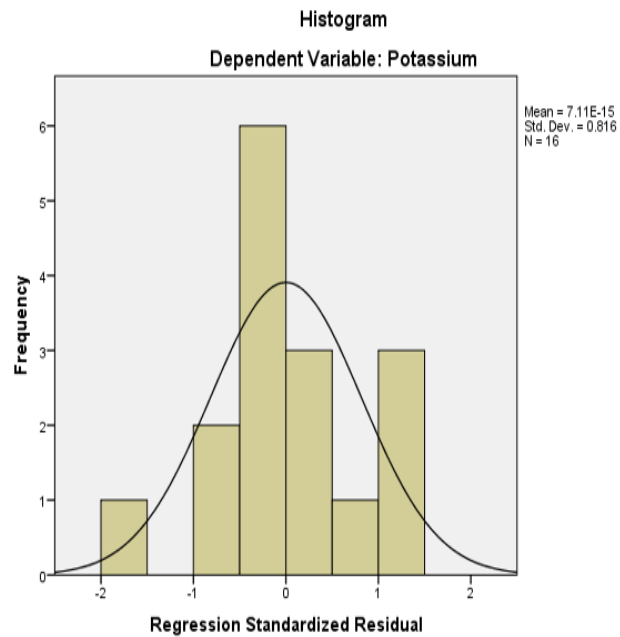
a. Dependent Variable: Phosphorus

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	413.8795	1419.5618	932.5000	277.25232	16
Residual	-169.56177	115.55907	.00000	84.56447	16
Std. Predicted Value	-1.871	1.757	.000	1.000	16
Std. Residual	-1.793	1.222	.000	.894	16

a. Dependent Variable: Phosphorus

Appendix 15: summary of multiple linear regression analysis results in graphs and tables for potassium



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.984 ^a	.968	.952	142.07707

a. Predictors: (Constant), gender, log_age, weight_cube, height_square, weight

b. Dependent Variable: Potassium

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	6162516.053	5	1232503.211	61.058	.000 ^b
	Residual	201858.947	10	20185.895		
	Total	6364375.000	15			

a. Dependent Variable: Potassium

b. Predictors: (Constant), gender, log_age, weight_cube, height_square, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	1474.871	170.705		8.640	.000	1094.517	1855.224
	weight	-88.411	28.703	-2.860	-3.080	.012	-152.366	-24.456
	height_square	.160	.043	2.021	3.758	.004	.065	.256
	weight_cube	.007	.002	1.343	3.581	.005	.003	.011
	log_age	252.852	88.432	.465	2.859	.017	55.814	449.890
	gender	283.435	103.335	.225	2.743	.021	53.192	513.679

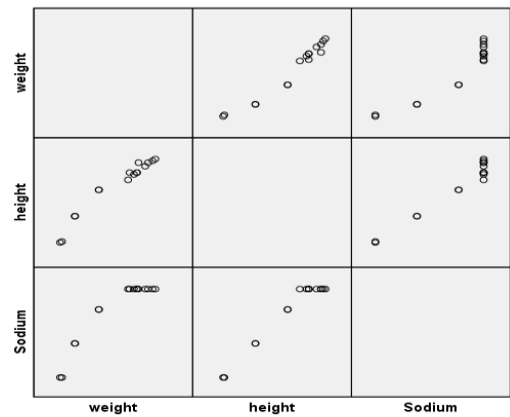
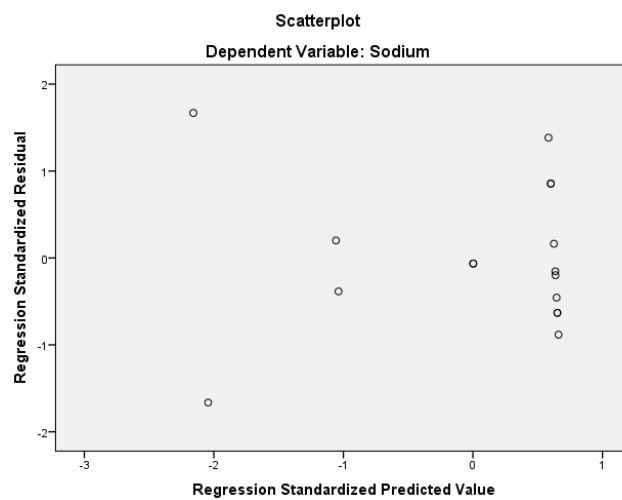
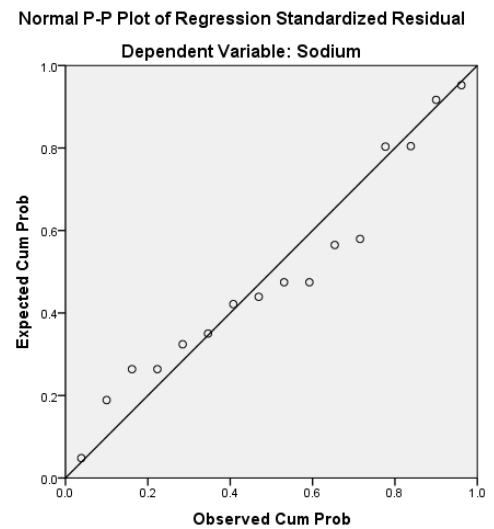
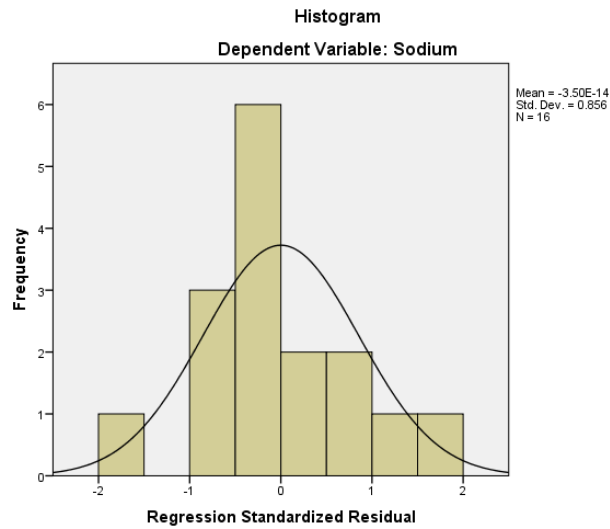
a. Dependent Variable: Potassium

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	1830.8689	3855.6531	2918.7500	640.96365	16
Residual	-264.94971	169.13107	.00000	116.00544	16
Std. Predicted Value	-1.697	1.462	.000	1.000	16
Std. Residual	-1.865	1.190	.000	.816	16

a. Dependent Variable: Potassium

Appendix 16: summary of multiple linear regression analysis results in graphs and tables for Sodium



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	1.000 ^a	.999	.999	4.87907

a. Predictors: (Constant), height_square, weight_square, weight, height

b. Dependent Variable: Sodium

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	305738.141	4	76434.535	3210.817	.000 ^b
	Residual	261.859	11	23.805		
	Total	306000.000	15			

a. Dependent Variable: Sodium

b. Predictors: (Constant), height_square, weight_square, weight, height

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-204.433	56.278		-3.633	.004	-328.300	-80.566
	weight	14.035	1.045	2.071	13.436	.000	11.736	16.334
	weight_square	-.108	.010	-1.333	-10.404	.000	-.131	-.085
	height	5.548	1.057	1.195	5.249	.000	3.221	7.874
	height_square	-.017	.004	-.994	-4.248	.001	-.026	-.008

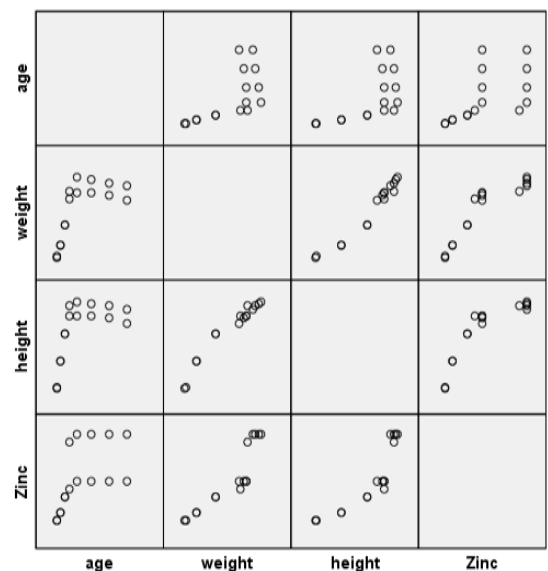
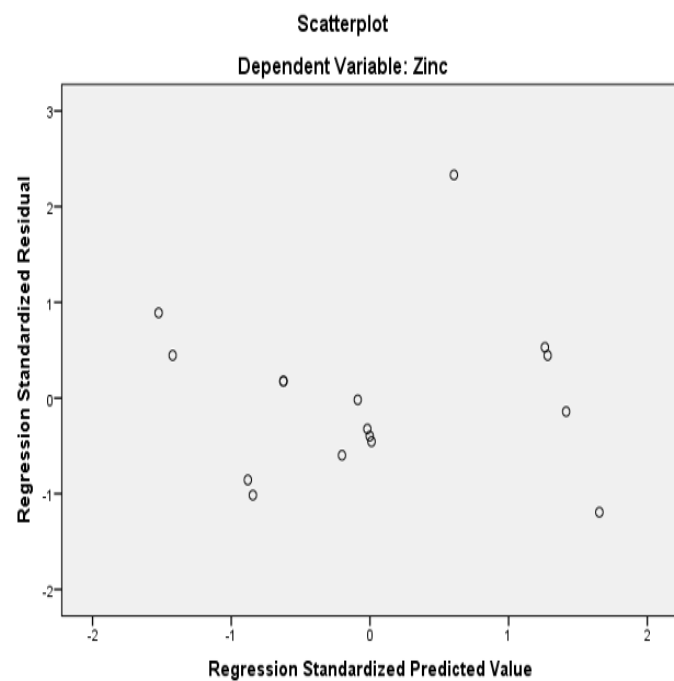
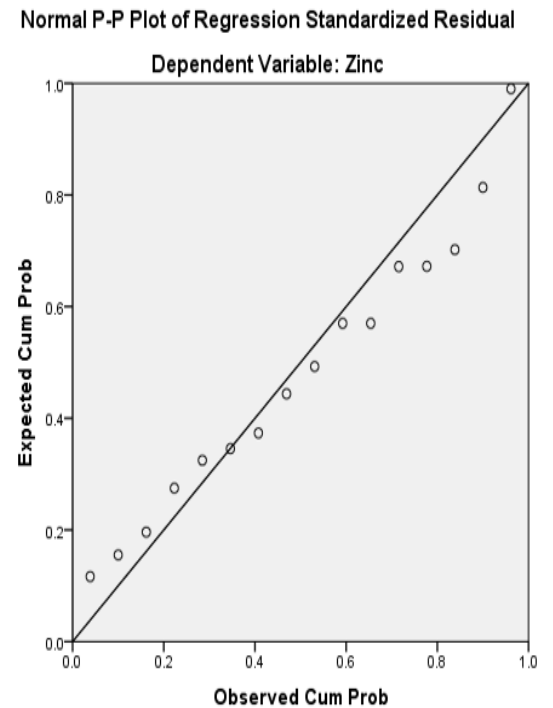
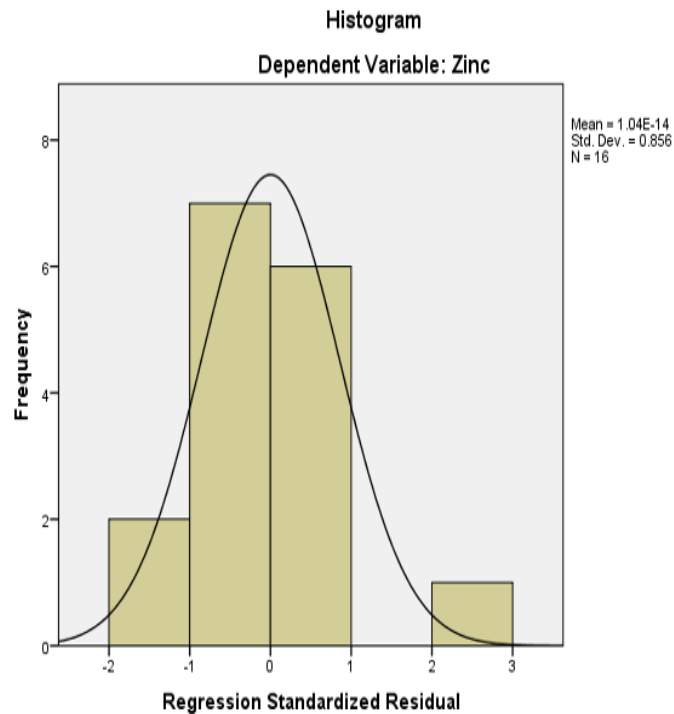
a. Dependent Variable: Sodium

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	291.8587	694.3030	600.0000	142.76744	16
Residual	-8.11628	8.14129	.00000	4.17819	16
Std. Predicted Value	-2.158	.661	.000	1.000	16
Std. Residual	-1.663	1.669	.000	.856	16

a. Dependent Variable: Sodium

Appendix 17: summary of multiple linear regression analysis results in graphs and tables for Zinc



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.982 ^a	.963	.950	.92661

a. Predictors: (Constant), weight, age_cube, weight_cube, height

b. Dependent Variable: Zinc

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	248.305	4	62.076	72.300	.000 ^b
	Residual	9.445	11	.859		
	Total	257.750	15			

a. Dependent Variable: Zinc

b. Predictors: (Constant), weight, age_cube, weight_cube, height

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-12.172	3.528		-3.450	.005	-19.937	-4.407
	age_cube	5.451E-006	.000	.235	3.301	.007	.000	.000
	weight_cube	6.699E-005	.000	2.047	6.908	.000	.000	.000
	height	.248	.053	1.839	4.700	.001	.132	.364
	weight	-.579	.125	-2.944	-4.617	.001	-.855	-.303

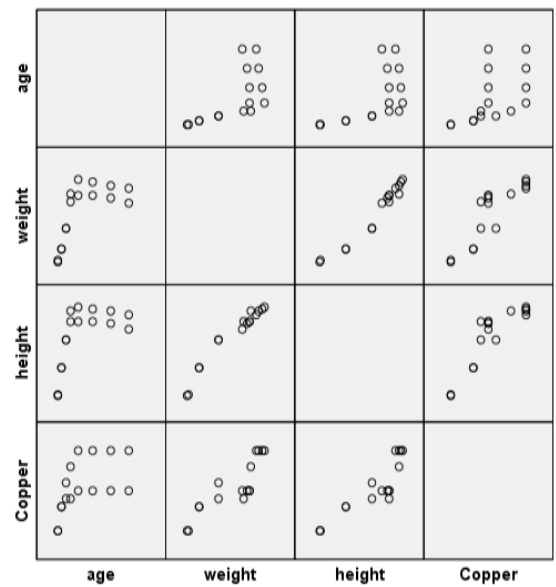
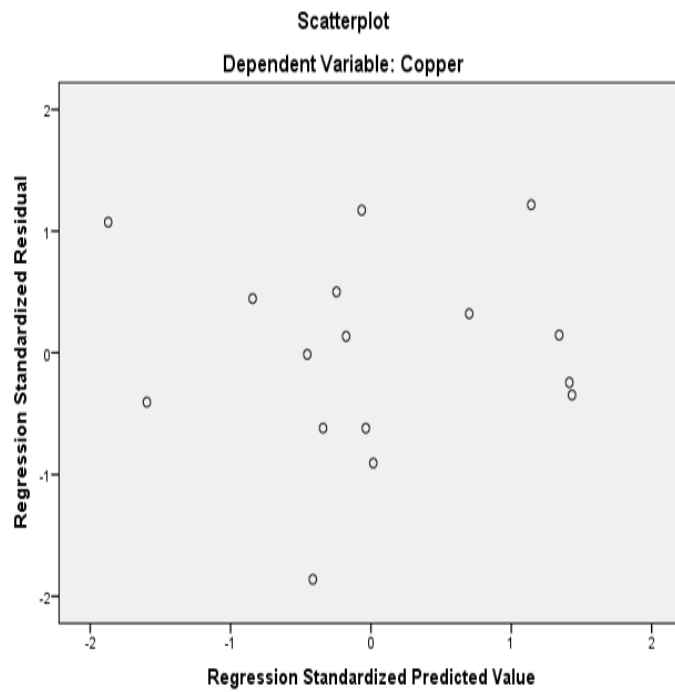
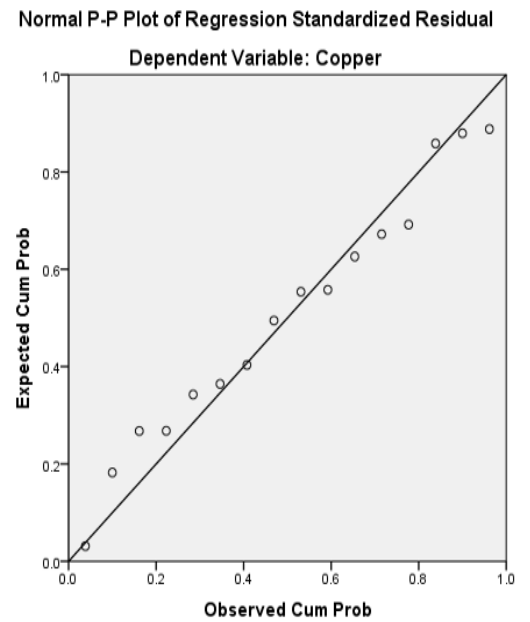
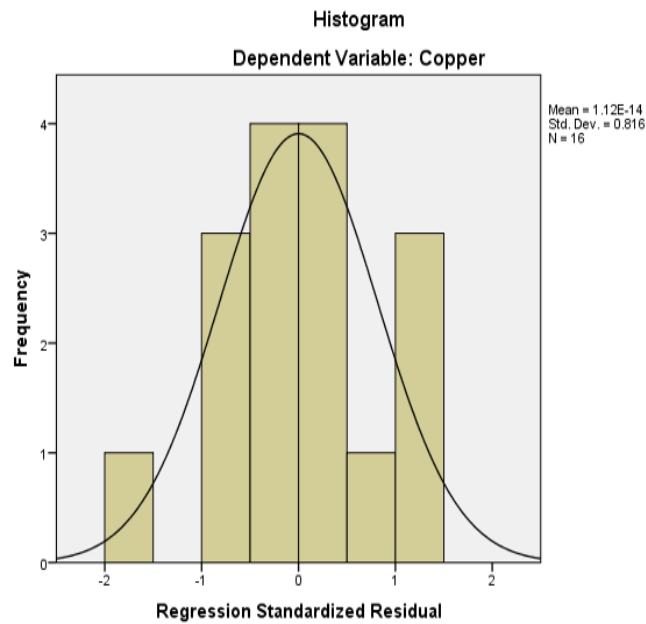
a. Dependent Variable: Zinc

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.1750	15.1054	8.3750	4.06862	16
Residual	-1.10541	2.16032	.00000	.79350	16
Std. Predicted Value	-1.524	1.654	.000	1.000	16
Std. Residual	-1.193	2.331	.000	.856	16

a. Dependent Variable: Zinc

Appendix 18: summary of multiple linear regression analysis results in graphs and tables for Copper



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.989 ^a	.977	.966	.06116

a. Predictors: (Constant), gender, log_age, weight_cube, height, weight

b. Dependent Variable: Copper

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1.623	5	.325	86.766	.000 ^b
	Residual	.037	10	.004		
	Total	1.660	15			

a. Dependent Variable: Copper

b. Predictors: (Constant), gender, log_age, weight_cube, height, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-.776	.242		-3.213	.009	-1.315	-.238
	log_age	.136	.034	.491	4.010	.002	.061	.212
	weight_cube	4.568E-006	.000	1.739	5.062	.000	.000	.000
	weight	-.051	.011	-3.248	-4.654	.001	-.076	-.027
	height	.022	.004	2.041	5.659	.000	.013	.031
	gender	.128	.042	.198	3.027	.013	.034	.221

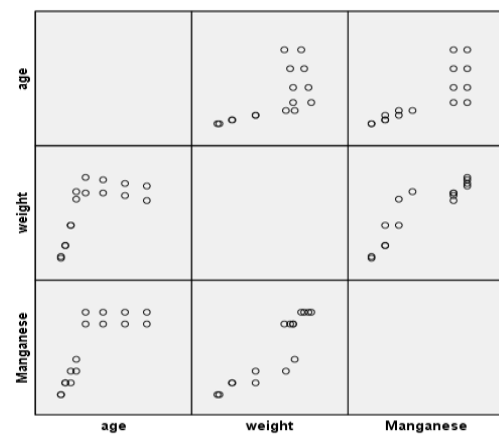
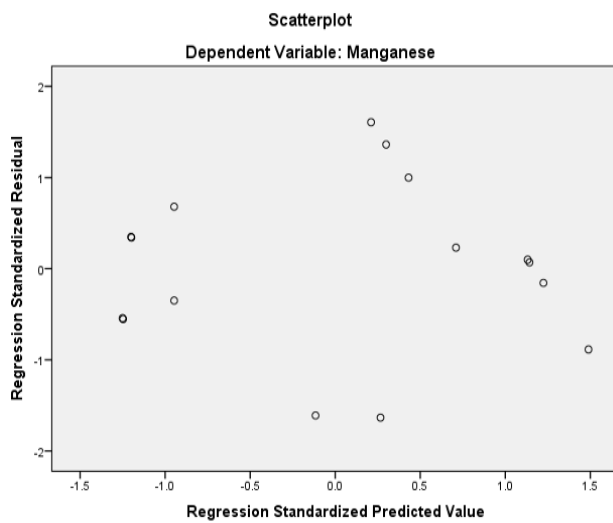
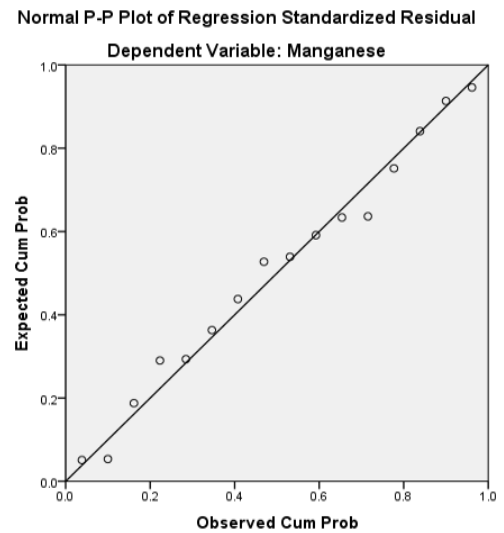
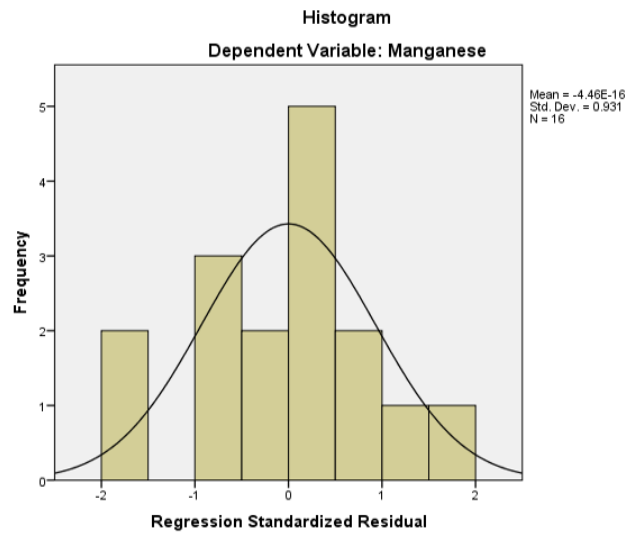
a. Dependent Variable: Copper

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	.6343	1.7212	1.2500	.32890	16
Residual	-.11384	.07444	.00000	.04993	16
Std. Predicted Value	-1.872	1.433	.000	1.000	16
Std. Residual	-1.861	1.217	.000	.816	16

a. Dependent Variable: Copper

Appendix 19: summary of multiple linear regression analysis results in graphs and tables for Manganese



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.947 ^a	.898	.882	.48582

a. Predictors: (Constant), weight_cube, age_cube

b. Dependent Variable: Manganese

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	26.869	2	13.435	56.922	.000 ^b
	Residual	3.068	13	.236		
	Total	29.938	15			

a. Dependent Variable: Manganese

b. Predictors: (Constant), weight_cube, age_cube

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	2.250	.201		11.178	.000	1.815	2.685
	age_cube	2.292E-006	.000	.290	3.066	.009	.000	.000
	weight_cube	9.007E-006	.000	.807	8.534	.000	.000	.000

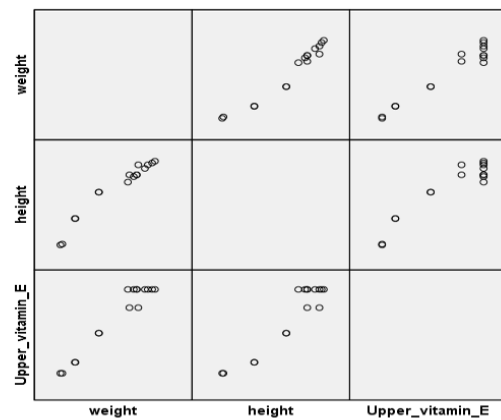
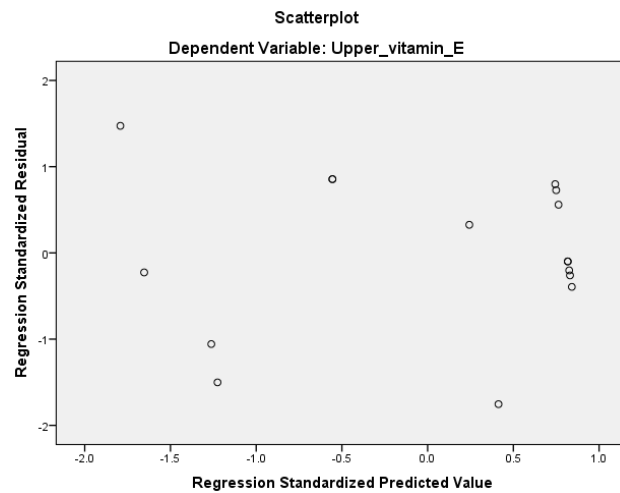
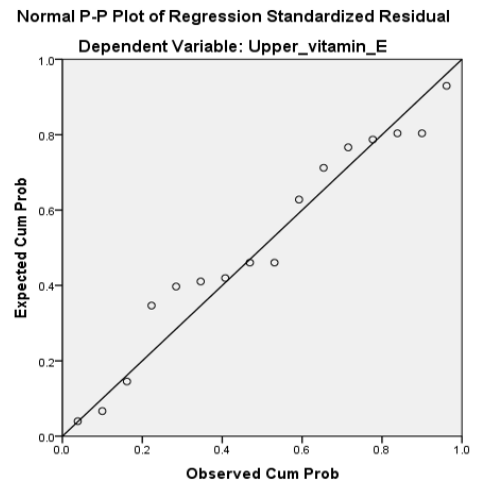
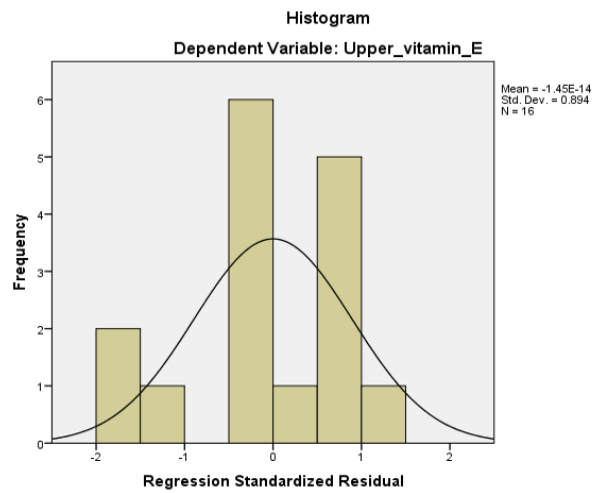
a. Dependent Variable: Manganese

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	2.2639	5.9307	3.9375	1.33839	16
Residual	-.79388	.78038	.00000	.45227	16
Std. Predicted Value	-1.250	1.489	.000	1.000	16
Std. Residual	-1.634	1.606	.000	.931	16

a. Dependent Variable: Manganese

Appendix 20: summary of multiple linear regression analysis results in graphs and tables for Upper vitamin E



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.997 ^a	.995	.993	7.57691

a. Predictors: (Constant), weight_square, height_cube, weight

b. Dependent Variable: Upper_vitamin_E

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	128911.086	3	42970.362	748.488	.000 ^b
	Residual	688.914	12	57.410		
	Total	129600.000	15			

a. Dependent Variable: Upper_vitamin_E

b. Predictors: (Constant), weight_square, height_cube, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-43.248	9.876		-4.379	.001	-64.766	-21.731
	weight	12.965	.865	2.939	14.994	.000	11.081	14.849
	height_cube	-6.522E-005	.000	-1.197	-8.665	.000	.000	.000
	weight_square	-.042	.007	-.793	-6.218	.000	-.057	-.027

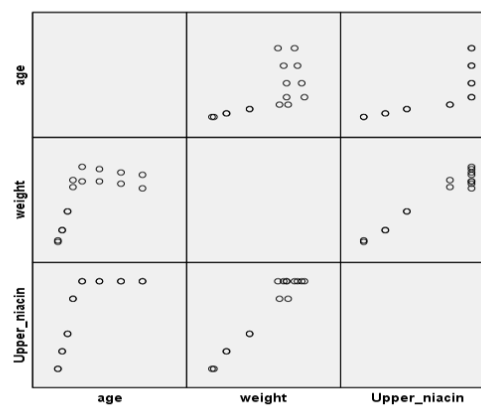
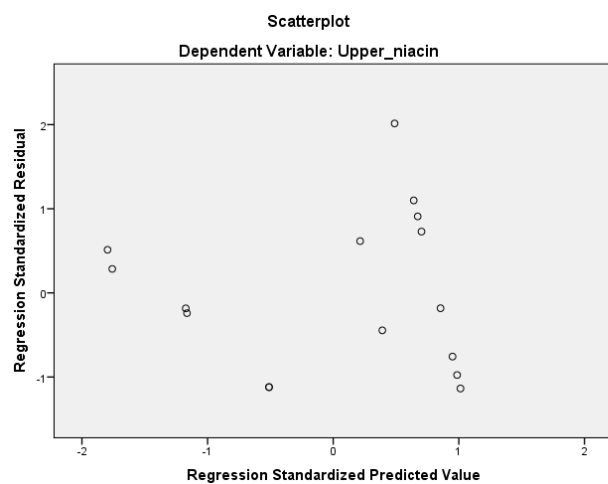
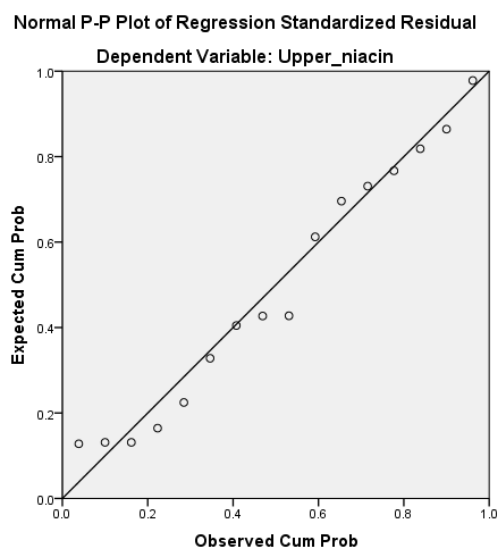
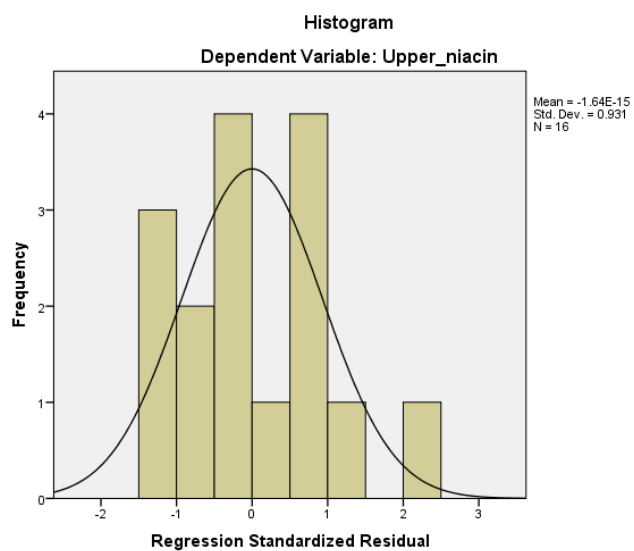
a. Dependent Variable: Upper_vitamin_E

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	58.8349	302.9878	225.0000	92.70422	16
Residual	-13.28832	11.16512	.00000	6.77699	16
Std. Predicted Value	-1.792	.841	.000	1.000	16
Std. Residual	-1.754	1.474	.000	.894	16

a. Dependent Variable: Upper_vitamin_E

Appendix 21: summary of multiple linear regression analysis results in graphs and tables for Upper niacin



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.988 ^a	.977	.973	1.64126

a. Predictors: (Constant), weight, log_age

b. Dependent Variable: Upper_niacin

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1458.732	2	729.366	270.765	.000 ^b
	Residual	35.018	13	2.694		
	Total	1493.750	15			

a. Dependent Variable: Upper_niacin

b. Predictors: (Constant), weight, log_age

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	3.512	1.104		3.180	.007	1.126	5.898
	log_age	2.989	.762	.359	3.921	.002	1.342	4.635
	weight	.311	.043	.656	7.177	.000	.217	.404

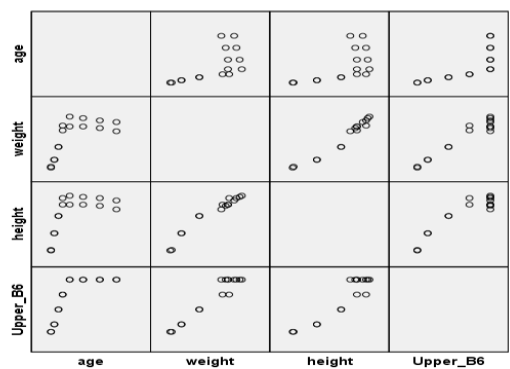
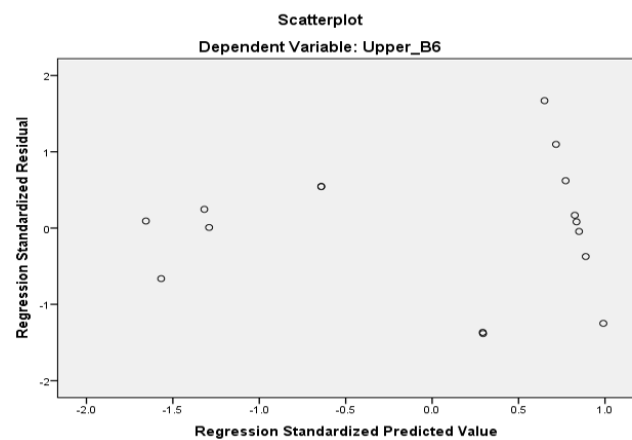
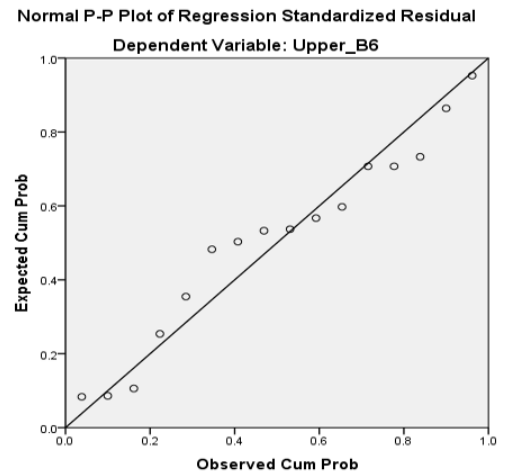
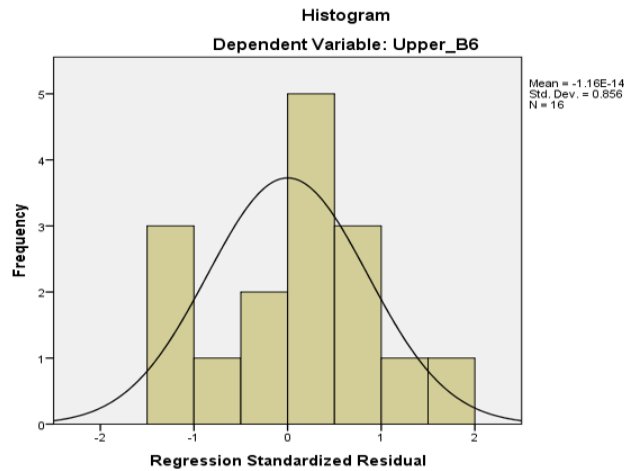
a. Dependent Variable: Upper_niacin

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	9.1590	36.8653	26.8750	9.86148	16
Residual	-1.86530	3.30644	.00000	1.52793	16
Std. Predicted Value	-1.796	1.013	.000	1.000	16
Std. Residual	-1.137	2.015	.000	.931	16

a. Dependent Variable: Upper_niacin

Appendix 22: summary of multiple linear regression analysis results in graphs and tables for Upper vitamin B6



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.995 ^a	.990	.987	1.63819

a. Predictors: (Constant), weight_cube, log_age, height, weight

b. Dependent Variable: Upper_B6

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2964.230	4	741.057	276.136	.000 ^b
	Residual	29.520	11	2.684		
	Total	2993.750	15			

a. Dependent Variable: Upper_B6

b. Predictors: (Constant), weight_cube, log_age, height, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	30.112	5.983		5.033	.000	16.944	43.280
	weight	1.383	.241	.2062	5.747	.000	.853	1.912
	height	-.389	.088	-.847	-4.412	.001	-.583	-.195
	log_age	3.419	.904	.290	3.782	.003	1.429	5.409
	weight_cube	-6.155E-005	.000	-.552	-3.328	.007	.000	.000

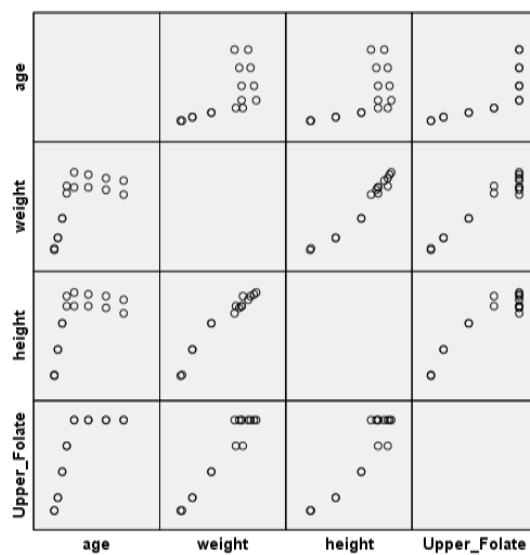
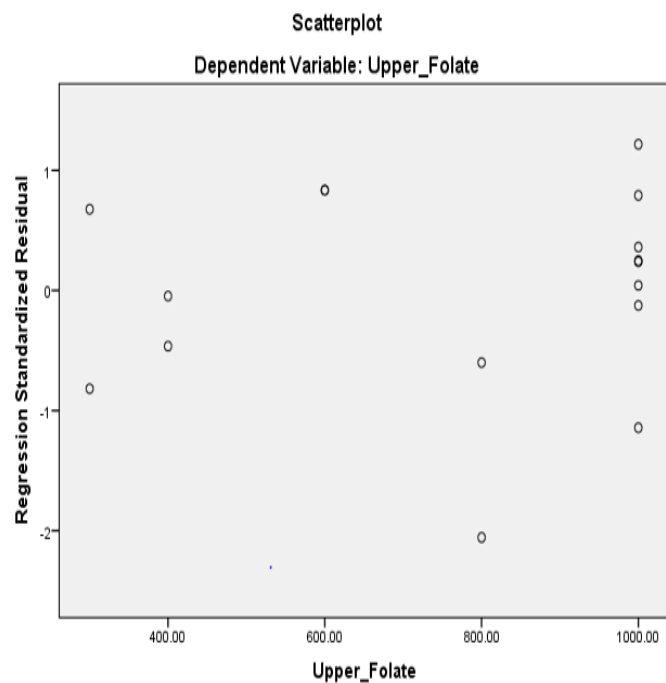
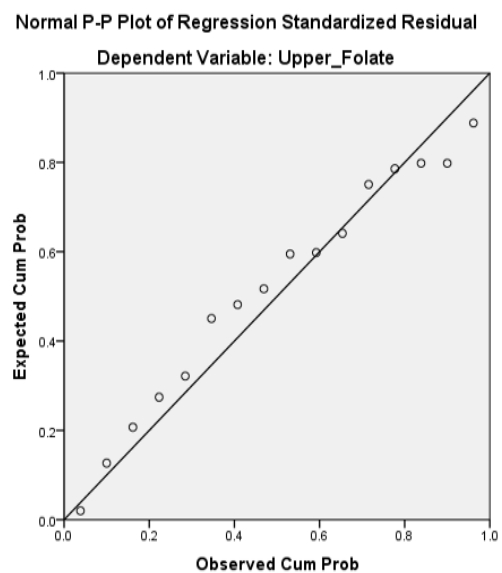
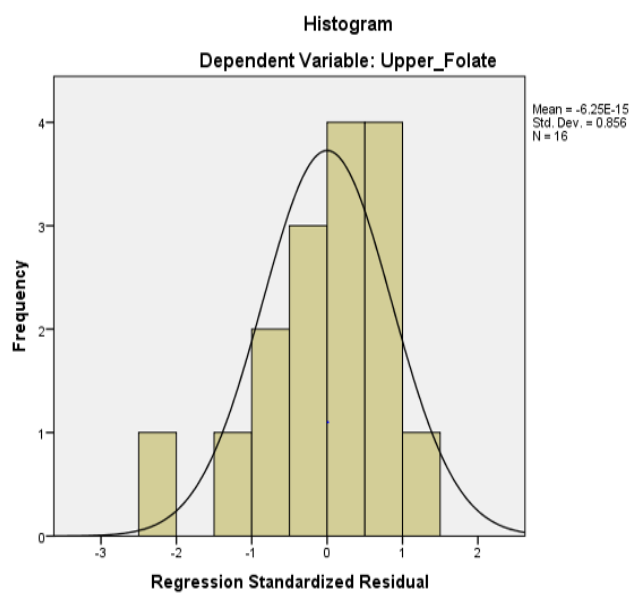
a. Dependent Variable: Upper_B6

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	14.8479	52.0463	38.1250	14.05757	16
Residual	-2.26515	2.73677	.00000	1.40286	16
Std. Predicted Value	-1.656	.990	.000	1.000	16
Std. Residual	-1.383	1.671	.000	.856	16

a. Dependent Variable: Upper_B6

Appendix 23: summary of multiple linear regression analysis results in graphs and tables for Upper folate



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.997 ^a	.995	.993	24.25377

a. Predictors: (Constant), height_square, log_age, weight_square, weight

b. Dependent Variable: Upper_Folate

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	1191029.302	4	297757.325	506.179	.000 ^b
	Residual	6470.698	11	588.245		
	Total	1197500.000	15			

a. Dependent Variable: Upper_Folate

b. Predictors: (Constant), height_square, log_age, weight_square, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	132.850	31.740		4.186	.002	62.991	202.709
	weight	40.128	5.450	2.993	7.363	.000	28.132	52.123
	log_age	41.284	15.236	.175	2.710	.020	7.750	74.818
	weight_square	-.154	.034	-.959	-4.509	.001	-.229	-.079
	height_square	-.043	.006	-1.252	-6.672	.000	-.057	-.029

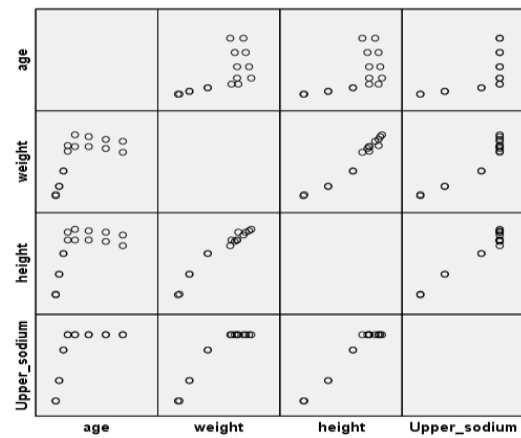
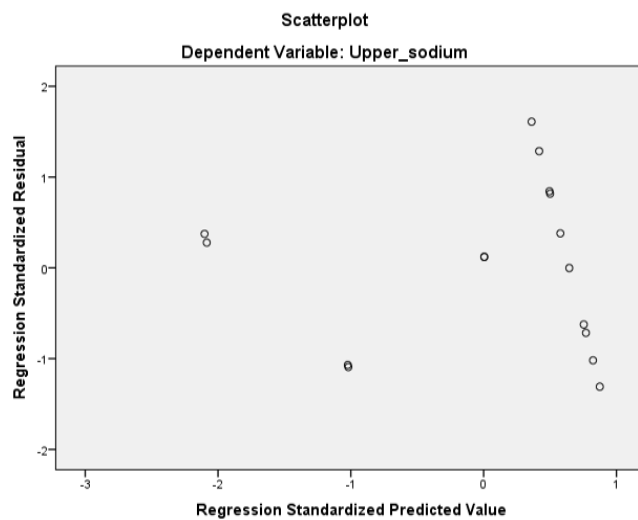
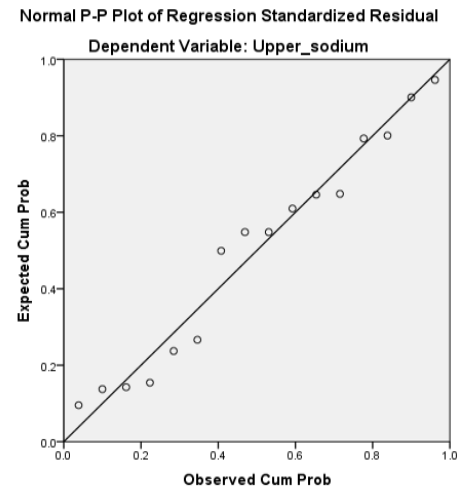
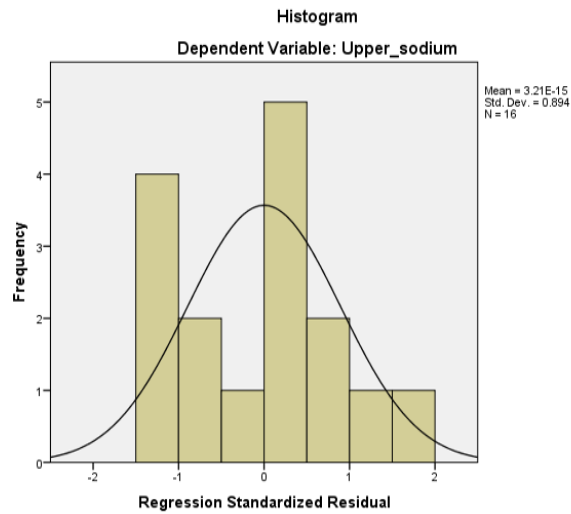
a. Dependent Variable: Upper_Folate

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	283.5881	1027.6979	762.5000	281.78352	16
Residual	-49.85039	29.51669	.00000	20.76969	16
Std. Predicted Value	-1.700	.941	.000	1.000	16
Std. Residual	-2.055	1.217	.000	.856	16

a. Dependent Variable: Upper_Folate

Appendix 24: summary of multiple linear regression analysis results in graphs and tables for Upper sodium



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.988 ^a	.976	.970	85.33255

a. Predictors: (Constant), log_age, weight_square, height_square

b. Dependent Variable: Upper_sodium

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	3530120.265	3	1176706.755	161.599	.000 ^b
	Residual	87379.735	12	7281.645		
	Total	3617500.000	15			

a. Dependent Variable: Upper_sodium

b. Predictors: (Constant), log_age, weight_square, height_square

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	354.670	107.992		3.284	.007	119.375	589.964
	weight_square	-.157	.043	-.562	-3.648	.003	-.250	-.063
	height_square	.074	.010	1.229	7.540	.000	.052	.095
	log_age	129.946	36.116	.317	3.598	.004	51.256	208.636

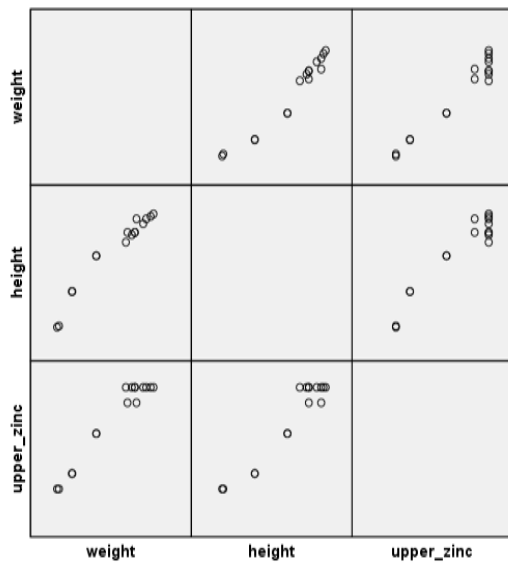
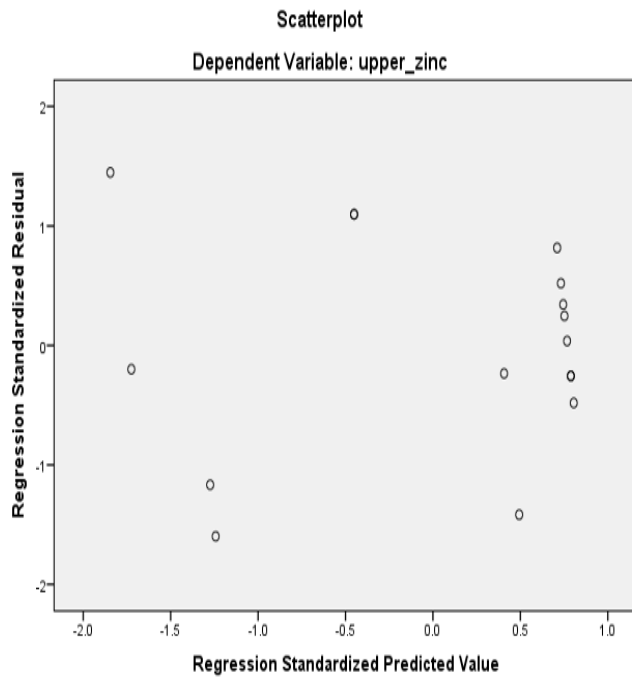
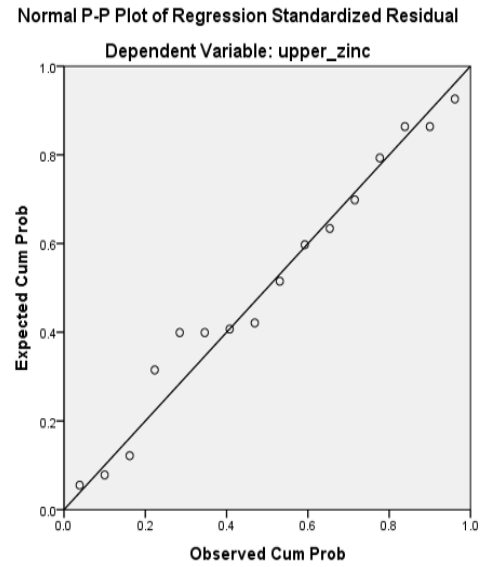
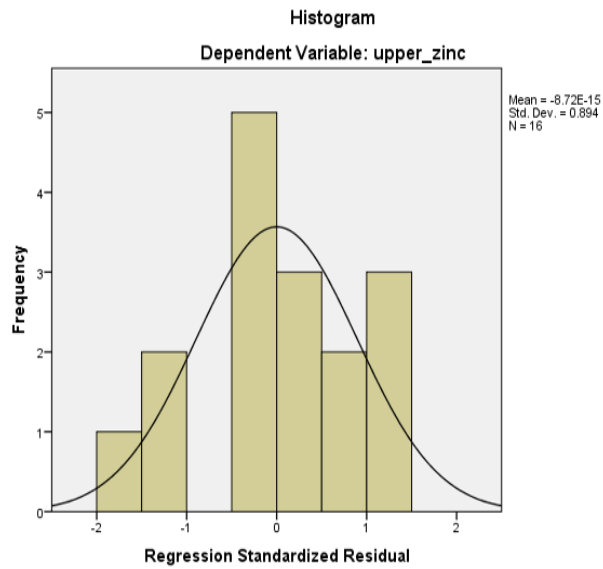
a. Dependent Variable: Upper_sodium

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	968.0610	2411.6948	1987.5000	485.11993	16
Residual	-111.69484	137.32472	.00000	76.32376	16
Std. Predicted Value	-2.101	.874	.000	1.000	16
Std. Residual	-1.309	1.609	.000	.894	16

a. Dependent Variable: Upper_sodium

Appendix 25: summary of multiple linear regression analysis results in graphs and tables for upper zinc



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.998 ^a	.996	.995	.96060

a. Predictors: (Constant), height_cube, weight_cube, weight

b. Dependent Variable: upper_zinc

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	2594.677	3	864.892	937.304	.000 ^b
	Residual	11.073	12	.923		
	Total	2605.750	15			

a. Dependent Variable: upper_zinc

b. Predictors: (Constant), height_cube, weight_cube, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-7.164	.982		-7.296	.000	-9.304	-5.025
	weight	1.461	.087	2.336	16.773	.000	1.272	1.651
	weight_cube	-6.195E-005	.000	-.595	-9.302	.000	.000	.000
	height_cube	-6.192E-006	.000	-.801	-6.518	.000	.000	.000

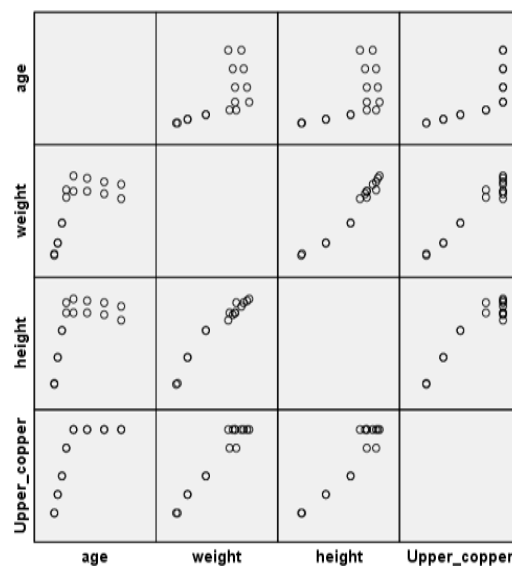
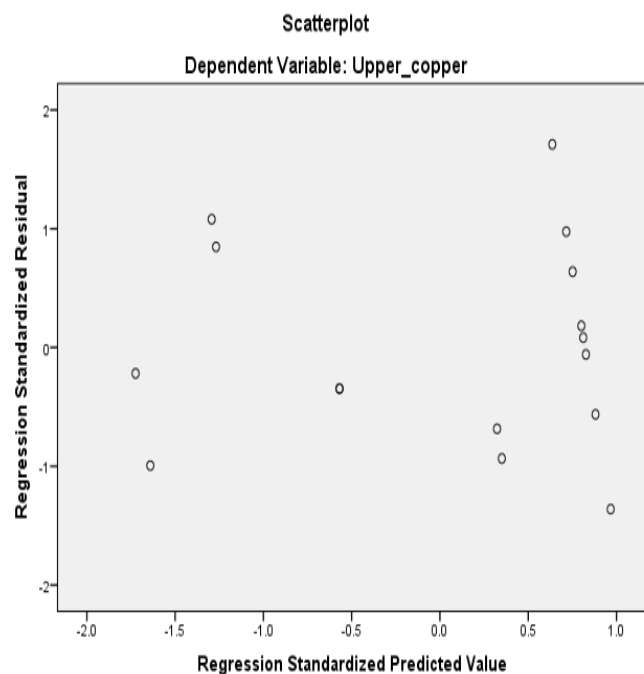
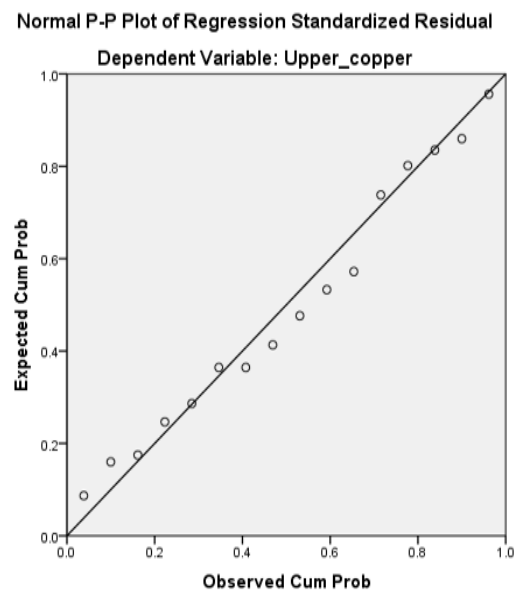
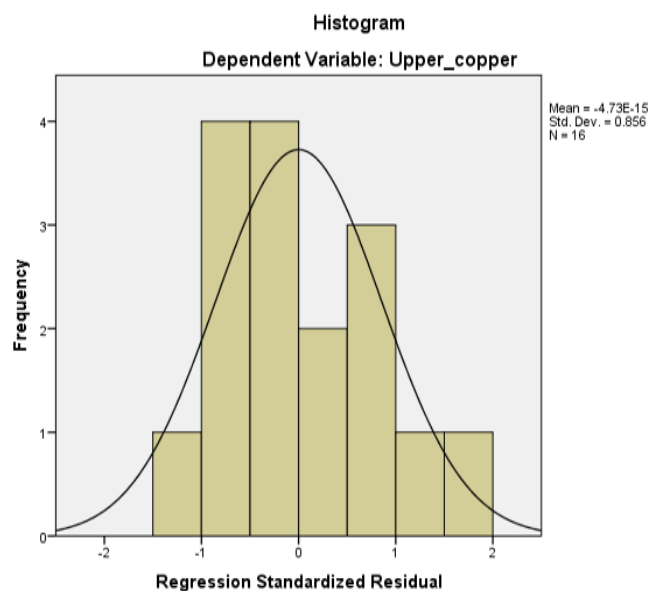
a. Dependent Variable: upper_zinc

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	5.6093	40.4624	29.8750	13.15213	16
Residual	-1.53472	1.39065	.00000	.85918	16
Std. Predicted Value	-1.845	.805	.000	1.000	16
Std. Residual	-1.598	1.448	.000	.894	16

a. Dependent Variable: upper_zinc

Appendix 26: summary of multiple linear regression analysis results in graphs and tables for upper copper



Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.996 ^a	.992	.989	.37716

a. Predictors: (Constant), height, log_age, weight_cube, weight

b. Dependent Variable: Upper_copper

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	184.185	4	46.046	323.700	.000 ^b
	Residual	1.565	11	.142		
	Total	185.750	15			

a. Dependent Variable: Upper_copper

b. Predictors: (Constant), height, log_age, weight_cube, weight

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	3.286	1.377		2.386	.036	.254	6.318
	log_age	.805	.208	.274	3.866	.003	.347	1.263
	weight_cube	-1.484E-005	.000	-.534	-3.486	.005	.000	.000
	weight	.312	.055	1.866	5.626	.000	.190	.434
	height	-.074	.020	-.643	-3.623	.004	-.118	-.029

a. Dependent Variable: Upper_copper

Residuals Statistics^a

	Minimum	Maximum	Mean	Std. Deviation	N
Predicted Value	1.0826	10.5136	7.1250	3.50414	16
Residual	-.51355	.64493	.00000	.32298	16
Std. Predicted Value	-1.724	.967	.000	1.000	16
Std. Residual	-1.362	1.710	.000	.856	16

a. Dependent Variable: Upper_copper

Appendix 27: MATLAB code for the function TwoPhaseSimplex

```
function K=TwoPhaseSimplex(A,b,c)
T=totbl(A,b,c); %Display the tableau
H=T.nonbas; %identifies none basic variables in the tableau
[m,n]=size(T.val); %find size of structure variable T
a=T.val(:,n)<0; % identifies any negative elements in the last column
               %of tableau
T=addcol(T,a,'x0',n); %This add artificial variable in tableau
T=addrow(T,[zeros(1,n-1) 1 0],'z0',m+1); %This adds artificial objective
                                         %function in the tableau

i=1:m-1;
ab=T.val(1:m,n+1); % Takes the last column in the tableau
ac= ab<0 ; %Find out the elements less than zero in the last column
ad=ab(ac); %Identifies the elements less than zero
ae=min(ad); %Find out the most negative element
af=find(ab==ae); %Find the in which the most negative element belongs
T=ljx(T,af,n); %Special pivoting
d=min(T.val(m+1,1:n)); %Find the minimum element in the last after special
                       %pivoting
counter=0; %Initialize the number of counting the number of loops

while d<0
    counter=counter+1; %Increament the number of loops
    s=find(T.val(m+1,1:n)==d); % Find the pivot column
    %v(counter)=s
    if length(s)>1 %Test whether there are multiple pivot and if it is,
                  %takes the first one
        counter=1;
        s=min(s);
    end
    aj=-T.val(1:m-1,n+1)./T.val(i,s); %Find the ratio of last column to
                                     %the pivot column
    ak=aj>0; %Find the ratios greater than zero
    al=aj(ak); %Identifies ratios greater than zero
    am=min(al); %Find the minimum ratio
    r=find(aj==am); %Find the pivot row
    T=ljx(T,r,s);
    d=min(T.val(m+1,1:n));
end
if T.val(m+1,n+1)>0
    disp('infesible')
    return
end
T=delcol(T,'x0'); T=delrow(T,'z0');
d1=min(T.val(m,1:n-1));
counterr=0;
```

```

while d1<0
    counterr=counterr+1;
    s1=find(T.val(m,1:n-1)==d1);
    if length(s1)>1
        s1=min(s1);
    end
    aj1=-T.val(1:m-1,n)./T.val(i,s1);
    ak1=aj1>0;
    if ak1==0
        disp('unbounded')
        return
    end
    al1=aj1(ak1);
    am1=min(al1);
    r1=find(aj1==am1);
    if length(r1)>1
        r1=min(r1);
    end
    T=ljx(T,r1,s1);
    d1=min(T.val(m,1:n-1));
end
B=T.bas;
s=intersect(B,H);
[l,y]=size(s);
j=1:l;
[~,ky]=ismember(s(j),B);
s
values=T.val(ky,n)
Totalcost=T.val(m,n)
end

```

Appendix 28: MATLAB code for the Graphical User Interface

```
function varargout = food_computing_model2(varargin)
% FOOD_COMPUTING_MODEL2 MATLAB code for food_computing_model2.fig
%     FOOD_COMPUTING_MODEL2, by itself, creates a new
%     FOOD_COMPUTING_MODEL2 or raises the existing singleton*.
%
%     H = FOOD_COMPUTING_MODEL2 returns the handle to a new
%     FOOD_COMPUTING_MODEL2 or the handle to
%     the existing singleton*.
%
%     FOOD_COMPUTING_MODEL2('CALLBACK',hObject,eventData,handles,...)
%     calls the local
%     function named CALLBACK in FOOD_COMPUTING_MODEL2.M with the
%     given input arguments.
%
%     FOOD_COMPUTING_MODEL2('Property','Value',...) creates a new
%     FOOD_COMPUTING_MODEL2 or raises the
%     existing singleton*. Starting from the left, property value pairs
%     are
%     applied to the GUI before food_computing_model2_OpeningFcn gets
%     called. An
%     unrecognized property name or invalid value makes property
%     application
%     stop. All inputs are passed to food_computing_model2_OpeningFcn
%     via varargin.
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only
%     one instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
%
% Edit the above text to modify the response to help food_computing_model2
%
% See also: GUIDE, GUIDATA, GUIHANDLES
%
% Edit the above text to modify the response to help food_computing_model2
%
% Last Modified by GUIDE v2.5 24-May-2015 07:18:22
%
% Begin initialization code - DO NOT EDIT
```

```

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',   @food_computing_model2_OpeningFcn, ...
                  'gui_OutputFcn',    @food_computing_model2_OutputFcn, ...
                  'gui_LayoutFcn',    [] , ...
                  'gui_Callback',     []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before food_computing_model2 is made visible.
function food_computing_model2_OpeningFcn(hObject, eventdata, handles, ...
    varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)
% varargin    command line arguments to food_computing_model2 (see VARARGIN)

% Choose default command line output for food computing model2
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes food_computing_model2 wait for user response (see UIRESUME)
% uiwait(handles.figure1);

```

```

% --- Outputs from this function are returned to the command line.
function varargout = food_computing_model2_OutputFcn(hObject, eventdata, ...
    handles)
% varargout    cell array for returning output args (see VARARGOUT);
% hObject     handle to figure
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on selection change in food_list.
function food_list_Callback(hObject, eventdata, handles)
% hObject     handle to food_list (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns food_list
% contents as cell array
%           contents{get(hObject,'Value')} returns selected item from
% food_list

% --- Executes during object creation, after setting all properties.
function food_list_CreateFcn(hObject, eventdata, handles)
% hObject     handle to food_list (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.
%       See ISPC and COMPUTER.

function cost_Callback(hObject, eventdata, handles)
% hObject     handle to cost (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of cost as text
%       str2double(get(hObject,'String')) returns contents of cost as a
% double
% --- Executes during object creation, after setting all properties.

```

```

function cost_CreateFcn(hObject, eventdata, handles)
% hObject    handle to cost (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
% --- Executes on button press in computation.
function computation_Callback(hObject, eventdata, handles)
% hObject    handle to computation (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
ko='optimal';
ka='not optimal';
MM=[170 130.3 267 160 58.3 23 316.6 85.5 355 36 89 61;
    2.9 1.5 16.9 2 0.4 3 5.2 0.4 58.6 3.5 1.1 3.5;
    0.7 2.5 0 6.7 1.2 2.4 1.4 0.5 0 4.4 2.6 0;
    0 91 0 7 90 819 0 4.3 0 399 3 27;
    0.1 0.1 0 2.1 0.7 1 0.2 0.2 0 3 0.1 0.1;
    0 11 0 10 41.33 10 0 18.1 0 32 8.7 0.5;
    0 0.1 0.1 0.1 0 0.1 1 0.1 0.1 0.1 0 0;
    0 0.1 0.1 0.1 0 0.2 0 0 0.3 0.1 0.1 0.1;
    0.5 0.8 2.2 1.7 0.2 0.5 0.6 0.5 8.1 0.8 0.7 0.1;
    0.1 0.2 0.2 0.3 0 0.2 0 0.2 0.4 0.2 0.4 0;
    2.6 26 4 81 25.3 146 0.9 14.5 28 113 20 7;
    0 0 1.3 0 0 0 0 0 12 0 0 0.4;
    0.5 0.2 0.3 1.4 0.2 0.2 0.2 0.2 1.5 0.1 0.3 0.4;

    91.1 36.1 132 52 3.5 56 54.3 25.4 1300 17 22 95;
    152 449.8 230 599 172 466 53.8 87.7 953 22 358 155;
    6 7 36 7 2.1 70 1.1 0.9 312 1.8 1 0;
    0.9 0.2 2.7 0.6 0.1 0.8 0.4 0.3 5.2 0.1 0.2 0;
    0.2 0.1 0.1 0.2 0 0.2 0.1 0 0.5 0.1 0.1 0;
    1 0.2 0 0.1 0 0.9 0.3 0 2.1 0.4 0.3 0;
    -0.1 -0.1 -0 -2.1 -0.7 -1 -0.2 -0.2 0 -3 -0.1 -0.1;
    -0.5 -0.8 -2.2 -1.7 -0.2 -0.5 -0.6 -0.5 -8.1 -0.8 -0.7 -0.1;
    -0.1 -0.2 -0.2 -0.3 -0 -0.2 0 -0.2 -0.4 -0.2 -0.4 0;
    -2.6 -26 -4 -81 -25.3 -146 -0.9 -14.5 -28 -113 -20 -7;
    -6 -7 -36 -7 -2.1 -70 -1.1 -0.9 -312 -1.8 -1 0;
    -0.9 -0.2 -2.7 -0.6 -0.1 -0.8 -0.4 -0.3 -5.2 -0.1 -0.2 0;
    -0.2 -0.1 -0.1 -0.2 0 -0.2 -0.1 0 -0.5 -0.1 -0.1 0]*10;

```

```

[yk,ky]=size(MM); R=-eye(ky);
MM=[MM;R]; a=get(handles.food_list,'value'); A1=MM(:,a);
AA=get(handles.age,'string');A=str2num(AA);WW=get(handles.weight,'string');
W=str2num(WW); HH=get(handles.height,'string'); H=str2num(HH);
GG=get(handles.gender,'string'); G=str2num(GG);
b=[0.002*A^3-19.773*A+14.399*W+18.367*H+178.263*G-753.86;%Energy
    3.028e-5*A^3+25.676*log(W)+10.193*G-58.621;%Protein
    2.247*log(A)+2.086e-5*W^3+2.309*G+12.641;%Fiber
    53.706*log(A)-9.815*W-15.25*H+0.093*(H^2)+912.549;%vitamin A
    2.930e-6*A^3-0.444*W+2.260e-5*W^3+0.001*H^2+2.762;%vitamin E
    0.850*W-0.003*W^2-0.001*H^2+32.492;%vitamin C
    (1.860e-7)*A^3-(4.845e-7)*W^3+(1.844e-7)*(H^3)-0.035*G+0.38;%Thiamin
    0.110*log(A)+1.186e-7*H^3+0.286;%Riboflavin
    2.434e-6*A^3-0.001*W^2+0.001*H^2-0.107*H+8.002;%Niacin
    7.226e-7*A^3-1.334e-6*W^3+0.023*W+0.192;%vitaminB6
    17.576*W-0.133*W^2-5.220*H+0.014*H^2+303.834;%folate
    0.063*W-4.486e-6*W^3-0.006*H+0.696;%vitamin B12
    1.680e-6*A^3+3.925e-5*W^3-0.004*W^2+1.698e-6*H^3+0.451*G+2.649;
    %pantothenic
    -0.005*W^3+0.253*H^2-42.095*H+2169.802;%phosphorus
    252.852*log(A)-88.411*W+0.007*W^3+0.160*H^2+283.435*G+1474.871;
    %potassium
    14.035*W-0.198*W^2+5.548*H-0.017*H^2-204.433;%sodium
    5.451e-6*A^3+6.699e-5*W^3-0.579*W+0.248*H-12.172;%zinc
    0.136*log(A)+4.56e-6*W^3-0.051*W+0.022*H+0.128*G-0.776;%copper
    2.292e-6*A^3+9.007e-6*W^3+2.250;%manganese
    -(12.965*W-0.042*W^2-6.522e-5*H^3-43.248);%super vitaminE
    -(2.989*log(A)+0.311*W+3.512);%upper niacin
    -(3.419*log(A)+1.383*W-6.15e-5*W^3-0.389*H+30.112);%upper vitamin B6
    -(41.284*log(A)+40.128*W-0.154*W^2-0.043*H^2+132.850);%upper folate
    -(129.946*log(A)-0.157*W^2+0.074*H^2+354.670);%upper sodium
    -(1.461*W-6.195e-5*W^3-6.192e-6*H^3-7.164);%upper zinc

    -(0.805*log(A)+0.312*W-1.484e-5*W^3-0.074*H+3.286);%upper copper
    -0.250;-0.250;-0.25;-0.20;-0.500;-0.180;-0.1500;-0.1500;-0.2500;...
    -0.100;-0.10;-0.250];

cc=get(handles.cost,'string'); c=str2num(cc); T=totbl(A1,b,c);
H=T.nonbas; [m,n]=size(T.val); a=T.val(:,n)<0;%Find out negative values
T=addcol(T,a,'x0',n);%Add artificial variable
T=addrow(T,[zeros(1,n-1) 1 0],'z0',m+1);% Add artificial objective function
i=1:m-1;

```



```

ab=T.val(1:m,n+1); % Takes the last column in the tableau
ac= ab<0 ; %Find out the elements less than zero in the last column
ad=ab(ac); %Identifies the elements less than zero
ae=min(ad); %Find out the most negative element
af=find(ab==ae) %Find the in which the most negative element belongs
T=ljx(T,af,n); %Special pivoting
d=min(T.val(m+1,1:n));%Find the minimum element in the last after special
%pivoting
counter=0; %Initialize the number of counting the number of loops
while d<0
    counter=counter+1; %Increament the number of loops
    s=find(T.val(m+1,1:n)==d)% Find the pivot column
    %v(counter)=s
    if length(s)>1 %Test whether there are multiple pivot and if it is,
        %takes the first one
        counter==1;
        s=min(s);
    end
    aj=-T.val(1:m-1,n+1)./T.val(i,s); %Find the ratio of last column to
    %the pivot column
    ak=aj>0; %Test if there are ratios greater than zero
    al=aj(ak);%Identifies ratios greater than zero
    am=min(al);%Find the minimum ratio greater than zero
    r=find(aj==am)%Find the pivot row
    T=ljx(T,r,s);
    d=min(T.val(m+1,1:n));
end
if T.val(m+1,n+1)>0
    disp('infesible')
    set(handles.remark,'string',ka)
    return
end

T=delcol(T,'x0'); T=delrow(T,'z0'); d1=min(T.val(m,1:n-1));
counterr=0;
while d1<0
    counterr=counterr+1;
    s1=find(T.val(m,1:n-1)==d1)
    if length(s1)>1
        s1=min(s1);
    end
    aj1=-T.val(1:m-1,n)./T.val(i,s1); ak1=aj1>0;
    if ak1==0
        disp('unbounded')
        return
    end
end

```

```

    all=aj1(ak1);    am1=min(all);    r1=find(aj1==am1)
    if length(r1)>1
        r1=min(r1);
    end
    T=ljx(T,r1,s1);    d1=min(T.val(m,1:n-1));
end
B=T.bas; s=intersect(B,H); [l,y]=size(s); j=1:l;
[~,ky]=ismember(s(j),B);
set(handles.index,'string',s);
set(handles.amount,'string',T.val(ky,n));
set(handles.total_cost,'string',T.val(m,n));
set(handles.remark,'string',ko);
function age_Callback(hObject, eventdata, handles)
% hObject    handle to age (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of age as text
%        str2double(get(hObject,'String')) returns contents of age as a
%double

% --- Executes during object creation, after setting all properties.
function age_CreateFcn(hObject, eventdata, handles)
% hObject    handle to age (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.

if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
function weight_Callback(hObject, eventdata, handles)
% hObject    handle to weight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hints: get(hObject,'String') returns contents of weight as text
%        str2double(get(hObject,'String')) returns contents of weight as a
%double

% --- Executes during object creation, after setting all properties.
function weight_CreateFcn(hObject, eventdata, handles)
% hObject    handle to weight (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

```

```

% Hint: edit controls usually have a white background on Windows.
%     See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function height_Callback(hObject, eventdata, handles)
% hObject    handle to height (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of height as text
%        str2double(get(hObject,'String')) returns contents of height as a
%        double
% --- Executes during object creation, after setting all properties.
function height_CreateFcn(hObject, eventdata, handles)
% hObject    handle to height (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%     See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function gender_Callback(hObject, eventdata, handles)
% hObject    handle to gender (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of gender as text
%        str2double(get(hObject,'String')) returns contents of gender as a
%        double
% --- Executes during object creation, after setting all properties.
function gender_CreateFcn(hObject, eventdata, handles)
% hObject    handle to gender (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%     See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),...
    get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

% --- Executes on button press in ResetComp.
function ResetComp_Callback(hObject, eventdata, handles)
% hObject      handle to ResetComp (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
set(handles.age, 'string', []);
set(handles.weight, 'string', []);
set(handles.height, 'string', []);
set(handles.gender, 'string', []);
set(handles.index, 'string', []);
set(handles.amount, 'string', []);
set(handles.total_cost, 'string', []);
set(handles.cost, 'string', []);
set(handles.remark, 'string', []);
set(handles.food_list, 'value', [])

% --- Executes during object creation, after setting all properties.
function uipanel4_CreateFcn(hObject, eventdata, handles)
% hObject      handle to uipanel4 (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      empty - handles not created until after all CreateFcns called

```