COMPUTER PROGRAM ON DRIS, MDRIS AND CND

- BIVARIATE AND MULTIVARIATE ANALYSES TOOLS FOR MONITORING THE SOIL AND PLANT NUTRIENT IMBALANCES

Senthil-Kumar Selvaradjou, Luca Montanarella and Aruna Geetha

2005

EUR 21505 EN
This document may be cited as follows:

COMPUTER PROGRAM ON DRIS, MDRIS AND CND

- BIVARIATE AND MULTIVARIATE ANALYSES TOOLS FOR MONITORING
  THE SOIL AND PLANT NUTRIENT IMBALANCES

Senthil-Kumar Selvaradjou¹, Luca Montanarella¹ and Aruna Geetha²

¹ Institute of Environment & Sustainability, Joint Research Centre, Ispra, Italy
² Tamil Nadu Agricultural University, Coimbatore, India
MISSION OF THE INSTITUTE FOR ENVIRONMENT AND SUSTAINABILITY

The mission of the Institute of Environment and Sustainability is to provide scientific and technical support to EU strategies for the protection of the environment and sustainable development. Employing an integrated approach to the investigation of air, water and soil contaminants, its goals are sustainable management of water resources, protection and maintenance of drinking waters, good functioning of aquatic ecosystems and good ecological quality of surface waters.

LEGAL NOTICE

Neither the European Commission nor any person acting on behalf of the Commission is responsible for the use, which might be made of the following information.

A great deal of additional information on the European Union is available on the Internet. It can be accessed through the Europa server (http://europa.eu.int)
## CONTENTS

<table>
<thead>
<tr>
<th>SNo</th>
<th>TITLE</th>
<th>Page No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>II</td>
<td>DIAGNOSIS AND RECOMMENDATION INTEGRATED SYSTEM</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>i BRIEF DESCRIPTION OF METHODOLOGY – DRIS AND MDRIS MODELS</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>ii ALGORITHM FOR DRIS AND MDRIS NUTRIENT MODELS</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>iii FLOW CHART OF THE COMPUTER PROGRAM - DRIS MODEL</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>iv COMPUTER PROGRAM - DRIS MODEL</td>
<td>10</td>
</tr>
<tr>
<td>III</td>
<td>MODIFIED - DIAGNOSIS AND RECOMMENDATION INTEGRATED SYSTEM</td>
<td>17</td>
</tr>
<tr>
<td></td>
<td>i FLOW CHART OF THE COMPUTER PROGRAM - MDRIS</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>ii COMPUTER PROGRAM - MDRIS MODEL</td>
<td>20</td>
</tr>
<tr>
<td>IV</td>
<td>COMPOSITIONAL NUTRITIVE DIAGNOSIS APPROACH</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>i BRIEF DESCRIPTION OF METHODOLOGY – CND MODEL</td>
<td>29</td>
</tr>
<tr>
<td></td>
<td>ii ALGORITHM FOR CND NUTRIENT MODEL</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>iii FLOW CHART OF THE COMPUTER PROGRAM – CND MODEL</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>iv COMPUTER PROGRAM – CND MODEL</td>
<td>32</td>
</tr>
<tr>
<td>V</td>
<td>OUTPUT DISPLAY OF A MODEL ON PROGRAM EXECUTION</td>
<td>35</td>
</tr>
<tr>
<td>VI</td>
<td>REFERENCES</td>
<td>47</td>
</tr>
</tbody>
</table>
CHAPTER - I

INTRODUCTION
The intensive cropping system pushing up the agricultural output level parallel with the present demographic transition imparts a cruel attack on the scarce and precious soil resources. The path of agricultural dynamism has closed the eye towards soil quality, which would challenge the agricultural production pertaining to human survival, is not a distant future. The unprincipled exploitation of soil resource has threatened soil quality, which is moving forward to touch the barren point. The need of the hour is not only to exploit the soil resource to fulfill the growing demand for food but also to sustain and conserve it. The present pace of soil degradation and environmental concerns of high input intensive agriculture are the major issues pertaining to soil fertility management and in developing strategies for sustainable agriculture through integrated and balanced nutrient management.

Indiscriminate use of unbalanced NPK application aggravates the micronutrient disorders, which act additively along with other biotic and abiotic stresses to limit the crop production. According to the law of minimum which governs maximizing the returns for the inputs, the largest response to a given input comes where there is no other limiting factor and the magnitude of response will increase as more and more limiting factors are corrected. Developing suitable management strategies based on these needs would help us in exploiting the full potentials of the soil by scooping out the obstacle of nutrient imbalance in the crops without spoiling the soil health.

In this context, the most popular nutrient diagnostic model of Beaufils (1973) with the acronym "DRIS" (Diagnosis and Recommendation Integrated System) and the most recent Compositional Nutritional Diagnosis model of Parent et al (1992) serves the purpose to diagnose the nutritional imbalance provoked inside the crop plant due to improper nourishment of the soil. But, the main limitations of these analysis methods were the involvement of extensive and voluminous rigorous computational steps. Due to the advancement in the computer software and hardware resources in the present era, calculations involving more than 4-5 nutrients in DRIS and CND which were considered to be Herculean task in the past, is now transformed to be comparatively simple and feasible. The computer programs for calculating Diagnosis and Recommendation Integrated System (DRIS), Modified Diagnosis and Recommendation Integrated System (MDRIS) and Compositional Nutritional Diagnosis (CND) approaches were developed in Microsoft VISUAL FOXPRO – 6.0. Using these programs and carefully following the guidelines given in this book, the nutrient norms and indices upto 12 nutrients can be calculated in easy steps.
CHAPTER II

DRIS Model

(DIAGNOSIS AND RECOMMENDATION INTEGRATED SYSTEM)
brief description of methodology - DRIS and MDRIS models

DRIS / MDRIS (Diagnosis Recommendation Integrated System (DRIS) of Beaufils, (1973) / Modified Diagnosis Recommendation Integrated System (M-DRIS) of Beaufils, (1973)) provides a means of ordering nutrient ratios or products into meaningful expressions called DRIS / MDRIS indices. Essentially, a nutrient index is a mean of the deviations of the ratios constraining a given nutrient from their respective optimum or norm values. The first step in implementing DRIS / MDRIS is the establishment of these standard values or norms. This is done using a survey data in which yield data are collected from cropping enterprise and nutrients concentration from the plant analysis data (index tissue of the plant) in order to build up a data bank representative of the crops.

Using yield and plant tissue nutrient concentration from the survey data, DRIS norms and coefficients of variations (CVs) are derived according to the procedure by Walworth and Sumner (1987). The statistical Critical Value Approach (CVA) of Cate and Nelson (1971) is used to derive the cut off for the high yielding and low yielding populations. Mean values for each nutrient expression together with their associated CVs and variances are then calculated for the two populations. The mean values (high yielding population) of nutrient expressions are ultimately chosen as diagnostic norms. The selection of nutrient ratio expression values with relatively large variance ratios (variance of low yielding population / variance of high yielding population) were done. DRIS indices are calculated for nutrients A – N using the following generalized equations:

\[ \text{A index} = \frac{f(A/B) + f(A/C) + f(A/D) + \ldots + f(A/N)}{Z} \]

\[ \text{B index} = \frac{f(A/B) + f(B/C) + f(B/D) + \ldots + f(B/N)}{Z} \]

\[ \text{N index} = \frac{f(N/B) + f(N/C) + f(N/D) + \ldots + f(N/M)}{Z} \]

where

\[ f(A/B) = \begin{cases} 
\frac{A/B}{a/b} - 1 & \text{if } A/B > a/b \times 1000 \frac{CV}{CV} \\
1 - \frac{a/b}{A/B} & \text{if } A/B < a/b 
\end{cases} \]

in which A/B is the value of the ratio of the two element in the tissue under diagnosis and a / b is the value of the corresponding norms, Z is the number of functions and CV is the coefficient of variation associated with each nutrient ratio norm a / b – a / n. In the case of MDRIS the yield is included as one of the nutrient parameter and is attached to the denominator in the expression for calculation of functions to be used for indices calculation.
ii. Algorithm for DRIS and MDRIS nutrient models

SET ENVIRONMENT
INPUT VARIABLE DECLARATION
GET - Input file(source file : EXCEL input FILE), Sample number,
no of parameters/nutrients studied, Enter output file
CREATE database tables
DECLARE ARRAYS wholefile(sample number, total parameters +1), class sum of square (sample no),
class sum of square1 (sample no), class sum of square2 (sample no),
class sum of square3 (sample no), class sum of square4 (sample no)....
DECLARE Memory variables
USE whole data file
APPEND FROM input file
COPY TO ARRAY whole file [sample number, total parameters +1]
SORT ARRAY -descending order of yield data e.g. (ASORT (wholefile, 1, -1,1))
APPEND FROM ARRAY whole file database TO whole file array
- Critical value calculation for yield cut off using Cate and Nelson's statistical class sum of
  square technique
- Creation of high and low population nutrient data arrays based on the cut off derived from
  above method
- Calculation for both combinations of various nutrient ratios for high and low populations
  separately in arrays
- Calculation of mean, variance, CV for both combinations of nutrient ratios for high and low
  population data separately in arrays
- Calculation of variance ratios between low population and high population files and selection
  of opt ratio for indices calculation
- Sending the parameters like mean, CV, Variance, Variance ratios, choice of ratio for indices
- Calculation, deficiency cut off, low level cut off, optimum level cut off, and high or sufficient
  level cutoff for both high and low population file
- Calculation of index functions for various nutrient ratios for both high and low population files
- Calculation of DRIS index value based on the index function, NII for both high and low
  population in arrays
- Transferring data from array to subsequent DRIS index EXCEL files

CLOSE ALL DATABASES
CLEAR ALL memory variables
iii. Flow chart of the computer program – DRIS model

**Database developed in Excel with uniform units of the data for input**

- Data acquisition from database
- Set Environment
- Declare arrays and variables
- Create database tables

**DATABASE CREATION**

Visual FOXPRO

**DATA INPUT**

**DATA PROCESSING**

- Yield cutoff using CV (using Cate & Nelson method)
- Calculation of indices
- Creation high low population database
- Calculation of mean, SD, CV, variance
- Calculation of variance ratios and optimum ratios
- Calculation of deficiency cutoff, optimum level and sufficiency cutoff values
- Calculation of nutrient Index functions
- Calculation of DRIS index and NII values

**DATA OUTPUT**

Microsoft EXCEL

- Transferring data of DRIS index into EXCEL file
- Close all databases and memory variable
iv. Computer program - DRIS model

SET ENVIRONMENT
CLEAR; CLEAR ALL; CLOSE ALL; SET SCOREBOARD OFF; SET STATUS OFF; SET TALK OFF; SET CONFIRM OFF; SET SAFETY OFF
CLOSE ALL
CLEAR

F1 = SPACE(20); T1=SPACE(20); T2=SPACE(20); XX1=SPACE(40)
STORE 0 TO SAMPLENO, PARANO, INDNO

@ 7, 10 SAY "ENTER INPUT FILE NAME" FONT 'COMIC SANS MS', 13 STYLE 'S'
@ 7, 50 GET F1 FONT 'COMIC SANS MS', 13 STYLE 'S'
@ 11, 10 SAY "ENTER TOTAL NO. OF SAMPLES" FONT 'COMIC SANS MS', 13 STYLE 'S'
@ 11, 50 GET SAMPLENO FONT 'COMIC SANS MS', 13 STYLE 'S'
@ 15, 10 SAY "ENTER NO. OF PARAMETERS" FONT 'COMIC SANS MS', 13 STYLE 'S'
@ 15, 50 GET PARANO FONT 'COMIC SANS MS', 13 STYLE 'S'
@ 19, 10 SAY "ENTER OUTPUT FILE NAME" FONT 'COMIC SANS MS', 13 STYLE 'S'
@ 19, 50 GET XX1 FONT 'COMIC SANS MS', 13 STYLE 'S'
READ

T1=ALLTRIM(XX1)*"DMAINSORT"
T2=ALLTRIM(XX1)*"DDRISLOW"

CREATE TABLE &T1 (Y N(8,2), N(9,5), P N(9,4), K N(9,5), CA N(9,5), MO N(9,5), NA N(9,5), S N(9,5), ZN N(9,5), CU N(9,5), FE N(9,5), Mn N(9,5), FNO N(9,5))
CREATE TABLE &T2 (N N(9,4), P N(9,4), K N(9,4), CA N(9,4), MO N(9,4), NA N(9,4), S N(9,4), ZN N(9,4), CU N(9,4), FE N(9,4), Mn N(9,4), TOT N(9,4), FNO N(9,4))
DECLARE DAT1 [SAMPLENO, PARANO+2], CSS1[SAMPLENO], CSS2[SAMPLENO], CSS3[SAMPLENO], CSS4[SAMPLENO], R2[SAMPLENO]
DECLARE TX1[SAMPLENO], TX2[SAMPLENO]
STORE 0 TO SS1, SS2, SS3, SS4, SS5, SS10, CAL1, I, LL1, XX2, AA3, AA4, AA5, AA6, AY, A6, ZZI, MAX1
STORE 0 TO MAX2, MAX3, MAX4, MAX5, LOWPOP, HIGHPOP, D, XXX, YYY, I, J, XXX, AAA, BBB, Z, FFF, GGG
STORE 0 TO TT, CU, SS, X, Z, G, H, X1L1, TOTD1, TOTD
SELECT 1
USE &T1 ALIAS X1 ZAP
APPEND FROM &F1 DELIMITED WITH TAB
COPY TO ARRAY DAT1
ASORT (DAT1, 1, -1,1)
CLEAR; ZAP
APPEND FROM ARRAY DAT1
CLOSE ALL
SELECT 2
USE &T2 ALIAS X14; ZAP

CV CALCULATION AND SELECTION OF CUT OFF FOR POPULATION

TT = 0; SS = 0
FOR i=1 TO SAMPLENO-1
    TT=TT+DAT1[i,1]; SS=SS+DAT1[i,1]; DAT1[i,1]=TT; CSS1[i]=SS
ENDFOR
TT=TT+DAT1[SAMPLENO,1]; SS=SS+DAT1[SAMPLENO,1]; DAT1[SAMPLENO,1]=TT; CSS1[SAMPLENO]=SS
TX1[SAMPLENO]=TT; CSS1[SAMPLENO]=SS
TSS=CSS1[SAMPLENO]-TX1[SAMPLENO]*TX1[SAMPLENO]*SAMPLENO
FOR i=2 TO SAMPLENO-1
TX2[i]=TX1[i]*[SAMPLENO]-TX1[i], CSS2[i]=CSS1[i]*[SAMPLENO]-CSS1[i]
CSS1[i]=CSS1[i]-TX1[i]*[TX1[i]]; CSS2[i]=CSS2[i]-TX2[i]*[TX2[i]]/[SAMPLENO]
R2[i]=TSS-(CSS1[i]+CSS2[i]); R2[i]=R2[i]/TSS100
ENDFOR

CUT=1
FOR I = 2 TO SAMPLENO-1
IF R2[I-1] > R2[I]
   CUT=CUT+1
ELSE
   EXIT
ENDIF
ENDFOR

GETTING PARAMETERS IN ORDER
DECLARE HIGH1[CUT, PARANO+1], LOW1[SAMPLENO-CUT, PARANO+1]
LOWPOP=SAMPLENO-CUT; HIGHPOP=CUT
FOR KKK= 1 TO PARANO+1
   FOR I = 1 TO HIGHPOP
      HIGH1[I,KKK]=DAT1[I], KKK+1
   ENDFOR
ENDFOR

KKK=0; X=1
FOR I= 1 TO PARANO+1
   FOR J= HIGHPOP + 1 TO SAMPLENO
      LOW1[X,]=DAT1[I,]+1; IF X=SAMPLENO-[HIGHPOP]
      X=1
   ELSE
      X=X+1
   ENDFOR
ENDFOR

INDICES CALCULATION - * CALCULATION OF INDICES A/B FOR HIGH POPULATION
INDNO=INT([PARANO*/PARANO-1])/2
DECLARE IND1[HIGHPOP, INDNO], IND2[HIGHPOP, INDNO]
l=0; XXX=1; K=0
FOR J= 1 TO PARANO
   FOR K=1(J+1) TO PARANO
      FOR I= 1 TO HIGHPOP
         IND1[I,XXX]=HIGH1[I,J]; IND1[I,INT(K)]; IND2[I,XXX]=HIGH1[I,INT(K)+1]+HIGH1[I,J]
      ENDFOR
      XXX+XXX+1
   ENDFOR
ENDFOR

CALCULATION OF MEAN, SD, CV, VARIANCE,
DECLARE SUMM1[INDNO], SSQ1[INDNO], VAR1[INDNO], CV1[INDNO], SSD1[INDNO], MEAN1[INDNO]
x=0; y=0
FOR I= 1 TO INDNO
   FOR J= 1 TO HIGHPOP
      SUM1=SUM1+IND1[I,J]; SS1=SS1+IND1[I,J]*IND1[I,J]
   ENDFOR
ENDFOR

MEAN1=SUM1/INDNO
VAR1=SSS1/INDNO
CV1=VAR1*
SD1=SQRT(VAR1)

MEAN1[INDNO]=MEAN1[INDNO]
ENDFOR
SUMM1[I]=SUM1; MEAN1[I]=SUMM1[I]/HIGHPOP; SSQ1[I]=SSS1
VAR1[I]=((HIGHPOP*SSQ1[I])-SUMM1[I])/(HIGHPOP*(HIGHPOP-1))
VAR1[I]=ABS(VAR1[I]); SSD1[I]=SQR(T(VAR1[I])); CV1[I]=((SQR(T(VAR1[I])))/MEAN1[I]); SUM1=0; SSS1=0
ENDFOR

SSS1=0; SUM1=0
DECLARE SUMM2[INDNO], SSQ2[INDNO], VAR2[INDNO], CV2[INDNO], SSD2[INDNO], MEAN2[INDNO]
FOR I= 1 TO INDNO
  FOR J= 1 TO HIGHPOP
    SUM1=SUM1+IND2[I,J]; SSS1=SSS1+(IND2[I,J]*IND2[I,J])
  ENDFORENDFOR
SUMM2[I]=SUM1; MEAN2[I]=(SUMM2[I]/HIGHPOP); SSQ2[I]=SSS1
VAR2[I]=((HIGHPOP*SSQ2[I]-SUMM2[I])/(HIGHPOP*(HIGHPOP-1))
VAR2[I]=ABS(VAR2[I]); SSD2[I]=SQR(T(VAR2[I])); CV2[I]=((SQR(T(VAR2[I])))/MEAN2[I])
SUM1=0; SSS1=0
ENDFOR

CALCULATION FOR LOW POPULATION SAMPLES

DECLARE IND3[LOWPOP, INDNO], IND4[LOWPOP, INDNO]
K1=0; X=1
FOR J= 1 TO PARANO
  FOR K=1+(J-1) TO PARANO
    FOR I= 1 TO LOWPOP
      IND3[I,X]=LOW1[I,J]/LOW1[I,K1]; IND4[I,X]=LOW1[I,K1]/LOW1[I,J]
    ENDFOR
    X=X+1
  ENDFOR
ENDFOR

SUM1=0; SSS1=0
DECLARE SUMM3[INDNO], SSQ3[INDNO], VAR3[INDNO], CV3[INDNO], SSD3[INDNO], MEAN3[INDNO], VARRATIO1[INDNO]
FOR I= 1 TO INDNO
  FOR J= 1 TO LOWPOP
    SUM1=SUM1+IND3[I,J]; SSS1=SSS1+(IND3[I,J]*IND3[I,J])
  ENDFORENDFOR
SUMM3[I]=SUM1; MEAN3[I]=SUMM3[I]/LOWPOP; SSQ3[I]=SSS1
VAR3[I]=((LOWPOP*SSQ3[I]-SUMM3[I])/(LOWPOP*(LOWPOP-1))); VAR3[I]=ABS(VAR3[I])
SSD3[I]=SQR(T(VAR3[I])); CV3[I]=((SQR(T(VAR3[I])))/MEAN3[I]); VARRATIO1[I]=VAR3[I]/VAR1[I]; SUM1=0; SSS1=0
ENDFOR

SSS1=0; SUM1=0
DECLARE SUMM4[INDNO], SSQ4[INDNO], VAR4[INDNO], CV4[INDNO], SSD4[INDNO], MEAN4[INDNO], VARRATIO2[INDNO]
FOR I= 1 TO INDNO
  FOR J= 1 TO LOWPOP
    SUM1=SUM1+IND4[I,J]; SSS1=SSS1+(IND4[I,J]*IND4[I,J])
  ENDFORENDFOR
SUMM4[I]=SUM1; MEAN4[I]=SUMM4[I]/LOWPOP; SSQ4[I]=SSS1
VAR4[I]=((HIGHPOP*SSQ4[I]-SUMM4[I])/(LOWPOP*(LOWPOP-1))
VAR4[I]=ABS(VAR4[I]); SSD4[I]=SQR(T(VAR4[I])); CV4[I]=((SQR(T(VAR4[I])))/MEAN4[I])
VARRATIO2[I]=VAR4[I]/VAR2[I]; SUM1=0; SSS1=0
ENDFOR

CALCULATION OF DEFICIENCY, OPTIMUM AND SUFFICIENCY CUTOFF LEVELS
DECLARE IND[1:HIGHPOP, INDNO], IND[2:LOWPOP, INDNO], INDIC[1:INDNO], PARAM[1:INDNO, 16], LOW[INDNO], DEF[INDNO], OPT[INDNO], HIG[INDNO]

FOR I = 1 TO INDNO
  PARAM[1,I]:=VAR[1,I]; PARAM[1,I]:=VAR[2,I]
  IF VAR[1,I] > VAR[2,I]
    FOR J = 1 TO HIGHPOP
      INDIC[1,J]:=IND1[J]
    ENDFOR
    FOR J = 1 TO LOWPOP
      IND[2,J]:=IND3[J]
    ENDFOR
    DEF[1]:=MEAN[1]-SSD[1]*1.33
  ELSE
    FOR J = 1 TO HIGHPOP
      INDIC[1,J]:=IND2[J]
    ENDFOR
    FOR J = 1 TO LOWPOP
      IND[2,J]:=IND4[J]
    ENDFOR
    DEF[2]:=MEAN[2]-SSD[2]*1.33
  ENDIF
ENDFOR

CALCULATION OF NUTRIENT INDEX FUNCTIONS

DECLARE INDICES[1:HIGHPOP, INDNO], INDICES[2:LOWPOP, INDNO]

FOR J = 1 TO INDNO
  FOR I = 1 TO HIGHPOP
    IF IND[1,J] >= PARAM[1,J, 1]
      INDICES[1,J]=((INDF[1,J]*ABS(PARAM[1,J,1]))-1)*(1000/PARAM[1,J,2])
    ELSE
      INDICES[1,J]=((1-(PARAM[1,J,1]*INDF[1,J])))*(1000/PARAM[1,J,2])
    ENDIF
  ENDFOR
ENDFOR

FOR J = 1 TO INDNO
  FOR I = 1 TO LOWPOP
    IF IND[2,J] >= PARAM[1,J, 1]
      INDICES[2,J]=((INDF[2,J]*PARAM[1,J,1])-1)*(1000/PARAM[1,J,2])
  ENDFOR
ENDFOR
ELSE
INDICESF2[I,J] = ((1-(PARAM1[I,J] * INDICESF2[I,J])) / (1000 * PARAM1[I,J]))
ENDIF
ENDFORENDFOR
DECLARE CLUE[PARANO, PARANO-1]
X=1
Z1=1
FOR I = 1 TO PARANO
  FOR J = Z1 TO PARANO-1
    IF INDICF1[I,J]#1
      CLUE[I,J]=1
    ELSE
      CLUE[I,J]=1
    ENDIF
    X=X+1
  ENDFORENDFOR
  Z1=Z1+1
ENDFOR
X=1
FOR I = 1 TO PARANO-1
  FOR J = I+1 TO PARANO
    IF INDICF1[I,J]#1
      CLUE[I,J]=1
    ELSE
      CLUE[I,J]=1
    ENDIF
    X=X+1
  ENDFORENDFOR
----------------------------------------------------------------------------------
CALCULATION OF DRIS INDEX AND NII VALUES
----------------------------------------------------------------------------------
DECLARE RAW1[PARANO, PARANO], TOTDRIS1[HIGHLPOP], DRIS1[HIGHLPOP, PARANO+2], TOTDRIS2[HIGHLPOP]
A1=0; Y=1; Z=1
FOR X1=1 TO HIGHLPOP
  X=1; Z1=1
  FOR I = 1 TO PARANO
    FOR J = Z1 TO PARANO-1
      RAW1[I,J]=INDICESF1[X1,X]*CLUE[I,J]
      X=X+1
    ENDFORENDFOR
  Z1=Z1+1
ENDFOR
X=1
FOR I = 1 TO PARANO-1
  FOR J = I+1 TO PARANO
    RAW1[I,J]=INDICESF1[X1,X]*CLUE[I,J]
    X=X+1
  ENDFORENDFOR
TOTDRIS1[X1]=0; TOTDRIS2[X1]=0
totd=0
FOR G=1 TO PARANO

14
DRIS1([X1, Y]=0
FOR H=1 TO PARANO-1
   DRIS1([X1, Y]=DRIS1([X1, Y]+RAW1[H, G])
ENDFOR
   DRIS1([X1, Y]=DRIS1([X1, Y]+PARANO); TODD1=ABS(DRIS1([X1, Y])); TODD=TODD+TODD1; Y=Y+1
   IF Y>PARANO; Y=1; ENDF
ENDFOR
   TOTDRIS1([X1]=TODD
ENDFOR
FOR I=1 TO HIGHPOP
   DRIS1[I, PARANO+1]=TOTDRIS1[I]; DRIS1[I, PARANO+2]=HIGH1[I, PARANO+1]
ENDFOR
DECLARE RAW2[PARANO, PARANO], TOTDRIS3[LOWPOP], DRIS2[LOWPOP, PARANO+2], TOTDRIS4[LOWPOP]
A1=0; Y=1; Z=1
FOR X1 = 1 TO LOWPOP
   X=1; Z1=1
FOR I=1 TO PARANO
   FOR J=Z1 TO PARANO-1
      RAW2[I, J]=INDICESF2[X1, X][CLUE][I, J]
      X=X+1
   ENDFOR
   Z1+Z1=1
ENDFOR
   X=1
FOR I=1 TO PARANO-1
   FOR J=I+1 TO PARANO
      RAW2[I, J]=INDICESF2[X1, X][CLUE][I, J]
      X=X+1
   ENDFOR
   ENDFOR
   TOTDRIS3[X1]=0; TOTDRIS4[X1]=0; TODD=0
FOR G=1 TO PARANO
   DRIS2[X1, Y]=0
   FOR H=1 TO PARANO-1
      DRIS2([X1, Y]=DRIS2([X1, Y]+RAW2[H, G])
   ENDFOR
      DRIS2([X1, Y]=DRIS2([X1, Y]+PARANO); TODD1=ABS(DRIS2([X1, Y])); TODD=TODD+TODD1; Y=Y+1
      IF Y>PARANO
         Y=1
      ENDF
   ENDFOR
   TOTDRIS3([X1]=TODD
ENDFOR
FOR I=1 TO LOWPOP
   DRIS2[I, PARANO+1]=TOTDRIS3[I]; DRIS2[I, PARANO+2]=LOW1[I, PARANO+1]
ENDFOR
TRANSFER OF DATA OUTPUT TO EXCEL & CLOSING ALL DATABASES AND MEMORY VARIABLES
SELE 2
APPEND FROM ARRAY DRIS2
CHAPTER III

MDRIS Model
(MODIFIED DIAGNOSIS AND RECOMMENDATION INTEGRATED SYSTEM)
i. Flow chart of the computer program - MDRIS

Database developed in Excel with uniform units of the data for input

- Data acquisition from database
- Set Environment
- Declare arrays and variables
- Create database tables

DATA INPUT

- Yield cutoff using CV (using Cate & Nelson method)
- Calculation of indices
- Creation high low population database
- Calculation of mean, SD, CV, variance
- Calculation of variance ratios and optimum ratios
- Calculation of deficiency cutoff, optimum level and sufficiency cutoff values
- Calculation of nutrient and yield Index functions
- Calculation of MDRIS index, yield index and NII values

DATA PROCESSING

- Transferring data of MDRIS index into EXCEL file
- Close all databases and memory variable

Microsoft EXCEL
ii. Computer program - MDRIS model

```
SET ENVIRONMENT
CLEAR; CLEAR ALL; CLOSE ALL
SET SCOREBOARD OFF; SET STATUS OFF; SET TALK OFF; SET CONFIRM OFF; SET SAFETY OFF
CLOSE ALL
CLEAR

F1 = SPACE(20); T1=SPACE(20); T16=SPACE(20); XX1=SPACE(40)
STORE 0 TO SAMPLENO, PARANO, INDNO

@ 2, 10 SAY "FILE NAME"
@ 2, 30 GET F1
@ 4, 10 SAY "ENTER TOTAL NUMBER OF SAMPLES"
@ 4, 70 GET SAMPLENO
@ 8, 10 SAY "ENTER NO OF PARAMETERS"
@ 8, 80 GET PARANO
@ 8, 10 SAY "ENTER OUTPUT FILE NAME"
@ 8, 80 GET XX1
READ

T1=ALLTRIM(XX1)+"MMAINSORT"
T16=ALLTRIM(XX1)+"MMDRISLOW"

CREATE TABLE &T1 (Y N(6,2), N N(3,4), P N(8,4), K N(8,8), CA N(8,5), MG N(8,5), Na N(8,5), S N(8,5), B N(8,5), ZN N(8,5), Cu N(8,5), Fe N(8,5), Mn N(8,5), FNO N(4))
CREATE TABLE &T16 (N N(3,4), P N(8,4), K N(8,8), CA N(8,5), MG N(8,5), Na N(8,5), S N(8,5), B N(8,5), ZN N(8,5), Cu N(8,5), Fe N(8,5), Mn N(8,5), A15 N(8,4), TOT N(8,4), FNO N(4))
DECLARE DAT1 [SAMPLENO, PARANO+=2, CSS[2][SAMPLENO], CSS[2][SAMPLENO], CSS[3][SAMPLENO], CSS[3][SAMPLENO], CSS[4][SAMPLENO], CSS[4][SAMPLENO]]
DECLARE TX1[SAMPLENO], TX2[SAMPLENO]
STORE 0 TO SS1, SS2, SS3, SS4, SS9, SS10, CAL1, I, LL1, XX2, AA3, AA4, XX5, AA5, AA6, AA7, AA9, ZZZ, MAX1
STORE 0 TO MAX2, MAX3, MAX4, MAX5, LOWPOP, HIGHPOP, D, XXX, YYY, I, J, KKK, AAA, BBB, Z, FFF, GGG
STORE 0 TO SS51, SUM1, MEAN1, VARR1, AAAA, BBBA, DRISTEMP1, DRISTEMP2, AA, BB
STORE 0 TO TT, CUT, SS, K, X, Z, G, H, X1, L1, TOTD1, TOTD, MMEAN1, MMMEAN2, TMDRIS, TMDRIS1, TMDRIS2, YYDRIS1,
YYDRIS2
SELECT 1
USE &T1 ALIAS X1; ZAP
APPEND FROM &F1 DELIMITED WITH TAB
COPY TO ARRAY DAT1; ASORT (DAT1, 1, -1, 1)
CLEAR; ZAP
CLOSE ALL
SELE 16
USE &T16 ALIAS X16; ZAP

CV CALCULATION AND SELECTION OF CUT OFF FOR POPULATION

TT=0; SS=0
FOR I=1 TO SAMPLENO-1
    TT=TT+DAT1[I,1]; SS=SS+(DAT1[I,1]*DAT1[I,1]); TX1[I]=TT; CSS[I]=SS
ENDFOR
TT=TT+DAT1[SAMPLENO,1]; SS=SS+DAT1[SAMPLENO,1]*DAT1[SAMPLENO,1]; TX1[SAMPLENO]=TT
CSS[SAMPLENO]=SS; TSS=CSS[SAMPLENO]; TX1[SAMPLENO]=TX1[SAMPLENO]; SAMPLENO
```
FOR I=2 TO SAMPLENO-1
    TX2[I]=TX1[SAMPLENO-I]; CSS2[I]=CSS1[SAMPLENO-I]; CSS1[I]=CSS1[I]-TX1[I]*TX1[I];
    CSS2[I]=CSS2[I]-TX2[I]*TX2[I](SAMPLENO-I); R2[I]=TSS-(CSS1[I]+CSS2[I]); R2[I]=R2[I]/TSS*100
ENDFOR

CUT=1
FOR I = 2 TO SAMPLENO-1
    IF R2[I] > R2[I]
        CUT=CUT+1
    ELSE
        EXIT
    ENDFOR
ENDFOR

GETTING PARAMETERS IN ORDER

DECLARE HIGH[1], CUT, PARANO+1, LOW[1][SAMPLENO-CUT, PARANO+1]
LOWPOP=SAMPLENO-CUT; HIGHPOP=CUT
FOR K=1 TO PARANO+1
    FOR I = 1 TO HIGHPOP
        HIGH[I][K]=DAT[I][, K]+1
    ENDFOR
ENDFOR

K=0; X=1
FOR I=1 TO PARANO+1
    FOR J=HIGHPOP+1 TO SAMPLENO
        LOW[I][J]=DAT[I][, J]+1
        IF X=SAMPLENO-(HIGHPOP)
            X=1
        ELSE
            X=X+1
        ENDFOR
    ENDFOR
ENDFOR

CALCULATION OF INDICES A/B FOR HIGH POPULATION

INDNO=INT((PARANO*(PARANO-1))/2)
DECLARE IND[1][HIGHPOP, INDNO], IND2[HIGHPOP, INDNO]
I=0; XXX+1; K=0
FOR J=1 TO PARANO
    FOR K1=(J+1) TO PARANO
        FOR I=1 TO HIGHPOP
            IND1[I][XXX]=HIGH1[I][J]; IND1[I][INT(K1)]; IND2[I][XXX]=HIGH1[I][INT(K1)][HIGH1[I][J]]
        ENDFOR
        XXX=XXX+1
    ENDFOR
ENDFOR

CALCULATION OF MEAN, SD, CV, VARIANCE

DECLARE SUMM[1][INDNO], SSQ[1][INDNO], VAR[1][INDNO], CV[1][INDNO], SSD[1][INDNO], MEAN[1][INDNO]
X=0; Y=0
FOR I=1 TO INDNO
    FOR J=1 TO HIGHPOP

SUM1=SUM1+IND1[J,J]; SS1=SS1+(IND1[J,J]^IND1[J,J])
ENDFOR
SUMM1[J]=SUM1; MEAN1[J]=SUMM1[J]/HIGHPOP; SSQ1[J]=SSS1
VAR1[J]=((HIGHPOP*SSQ1[J]-SUMM1[J]^SUMM1[J])/(HIGHPOP^HIGHPOP-1))
VAR1[J]=ABS(VAR1[J]); SSD1[J]=SQRT(VAR1[J]); CV1[J]=(SQRT(VAR1[J]))/MEAN1[J]; SUM1=0; SS1=0
ENDFOR

SSS1=0; SUM1=0
DECLARE SUMM2[INDNO], SSQ2[INDNO], VAR2[INDNO], CV2[INDNO], SSD2[INDNO], MEAN2[INDNO]
FOR J=1 TO INDNO
FOR J=1 TO HIGHPOP
SUM1=SUM1+IND2[J,J]; SS1=SS1+(IND2[J,J]^IND2[J,J])
ENDFOR
SUMM2[J]=SUM1; MEAN2[J]=SUMM2[J]/HIGHPOP; SSQ2[J]=SSS1
VAR2[J]=((HIGHPOP*SSQ2[J]-SUMM2[J]^SUMM2[J])/(HIGHPOP^HIGHPOP-1))
VAR2[J]=ABS(VAR2[J]); SSD2[J]=SQRT(VAR2[J]); CV2[J]=(SQRT(VAR2[J]))/MEAN2[J]; SUM1=0; SS1=0
ENDFOR

CALCULATION FOR LOW POPULATION SAMPLES

DECLARE IND3[LOWPOP, INDNO], IND4[LOWPOP, INDNO]
K1=0; X=1
FOR J=1 TO PARANO
FOR K1=(J+1) TO PARANO
FOR I=1 TO LOWPOP
IND3[I,J]=LOW1[I,J]; LOW1[K1] = LOW1[K1]; LOW8[I,J]
ENDFOR
X=X+1
ENDFOR
ENDFOR

SSM1=0; SS1=0
DECLARE SUMM3[INDNO], SSQ3[INDNO], VAR3[INDNO], CV3[INDNO], SSD3[INDNO], MEAN3[INDNO], VARRATIO1[INDNO]
FOR J=1 TO INDNO
FOR J=1 TO LOWPOP
SUM1=SUM1+IND3[J,J]; SS1=SS1+(IND3[J,J]^IND3[J,J])
ENDFOR
SUMM3[J]=SUM1; MEAN3[J]=SUMM3[J]/LOWPOP; SSQ3[J]=SSS1
VAR3[J]=((LOWPOP*SSQ3[J])-(SUMM3[J]^SUMM3[J])/(LOWPOP*(LOWPOP-1))
VAR3[J]=ABS(VAR3[J]); SSD3[J]=SQRT(VAR3[J]); CV3[J]=(SQRT(VAR3[J]))/MEAN3[J]
VARRATIO1[J]=VAR3[J]/VAR1[J]; SUM1=0; SS1=0
ENDFOR
SSS1=0; SUM1=0
DECLARE SUMM4[INDNO], SSQ4[INDNO], VAR4[INDNO], CV4[INDNO], SSD4[INDNO], MEAN4[INDNO], VARRATIO2[INDNO]
FOR I=1 TO INDNO
FOR J=1 TO LOWPOP
SUM1=SUM1+IND4[J,J]; SS1=SS1+(IND4[J,J]^IND4[J,J])
ENDFOR
SUMM4[J]=SUM1; MEAN4[J]=SUMM4[J]/LOWPOP; SSQ4[J]=SSS1
VARRATIO2[J]=VAR4[J]/VAR2[J]; SUM1=0; SS1=0
ENDFOR
DECLARE IND5[HIGHPOP, INDNO], IND5[LOWPOP, INDNO], INDIC[INDNO], PARAM1[INDNO, 18], LOW[INDNO], DEF[INDNO], OPT[INDNO], HIG[INDNO]
FOR I = 1 TO INDNO
    PARAM1[I, 15] = VARRATIO1[I]; PARAM1[I, 16] = VARRATIO2[I]
    IF VARRATIO1[I] > VARRATIO2[I]
        FOR J = 1 TO HIGHPOP
            INDIC1[I, J] = 1; INDIF1[J, I] = IND1[J, I]
        ENDFOR
        FOR J = 1 TO LOWPOP
            INDIF2[J, I] = IND3[J, I]
        ENDFOR
        PARAM1[I, 3] = SSD1[I]; PARAM1[I, 4] = VAR1[I]; PARAM1[I, 5] = MEAN2[I]; PARAM1[I, 6] = CV2[I]
        PARAM1[I, 7] = SSD2[I]; PARAM1[I, 8] = VAR2[I]; PARAM1[I, 9] = VARRATIO1[I]; PARAM1[I, 10] = INDIC1[I]
        PARAM1[I, 11] = DEFI[I]; PARAM1[I, 12] = LOWF[I]; PARAM1[I, 13] = OPT[I]; PARAM1[I, 14] = HIG[I]
    ELSE
        FOR J = 1 TO HIGHPOP
            INDIC1[I, J] = 2; INDIF1[J, I] = IND4[J, I]
        ENDFOR
        FOR J = 1 TO LOWPOP
            INDIF2[I, I] = IND4[I, I]
        ENDFOR
        HIG[I] = MEAN2[I] + (SSD2[I] ** (2 / 67)); PARAM1[I, 1] = MEAN2[I]; PARAM1[I, 2] = CV2[I]; PARAM1[I, 3] = SSD2[I]
        PARAM1[I, 4] = VAR2[I]; PARAM1[I, 5] = MEAN4[I]; PARAM1[I, 6] = CV4[I]; PARAM1[I, 7] = SSD4[I]
        PARAM1[I, 8] = VAR4[I]; PARAM1[I, 9] = VARRATIO2[I]; PARAM1[I, 10] = INDIC1[I]; PARAM1[I, 11] = DEFI[I]
        PARAM1[I, 12] = LOWF[I]; PARAM1[I, 13] = OPT[I]; PARAM1[I, 14] = HIG[I]
    ENDFOR
ENDFOR
DECLARE INDICESF1[HIGHPOP, INDNO], INDICESF2[LOWPOP, INDNO]
FOR J = 1 TO INDNO
    FOR I = 1 TO HIGHPOP
        IF INDIF1[I, J] >= PARAM1[I, 1]
            INDICESF1[I, J] = (((INDIF1[I, J] * ABS(PARAM1[I, 1]))) ** 1000) / PARAM1[I, 2]
        ELSE
            INDICESF1[I, J] = ((1 - PARAM1[I, 1] / INDIF1[I, J]) ** 1000) / PARAM1[I, 2]
        ENDFOR
    ENDFOR
ENDFOR
FOR J = 1 TO INDNO
    FOR I = 1 TO LOWPOP
        IF INDIF2[I, J] >= PARAM1[I, 1]
            INDICESF2[I, J] = (((INDIF2[I, J] * PARAM1[I, 1])) ** 1000) / PARAM1[I, 2]
        ELSE
            INDICESF2[I, J] = ((1 - PARAM1[I, 1] / INDIF2[I, J]) ** 1000) / PARAM1[I, 2]
        ENDFOR
    ENDFOR
ENDFOR
DECLARE CLUE[PARANO, PARANO-1]
X=1; Z1=1
FOR I = 1 TO PARANO
    FOR J = Z1 TO PARANO-1
        IF INDIC1[I, J] # 1
            CLUE[I, J] = 1
        ENDFOR
    ENDFOR
23
ELSE
  CLUE(J)=1
ENDIF
X=X+1
ENDFOR
Z1=Z1+1
ENDFOR
X=1
FOR I = 1 TO PARANO-1
  FOR J= I+1 TO PARANO
    IF INDIC(I)X#1
      CLUE(J)=1
    ELSE
      CLUE(J)=1
    ENDFOR
    X=X+1
  ENDFOR
ENDFOR
DECLARE RAW1[PARANO, PARANO], TOTDRIS1[HIGHPOP], DRIS1[HIGHPOP, PARANO+1], TOTDRIS2[HIGHPOP]
A1=0; Y=1; Z=1
FOR X1= 1 TO HIGHPOP
X=1; Z1=1
FOR I = 1 TO PARANO
  FOR J= Z1 TO PARANO-1
    RAW1[I,J]=INDICESF[I,X1,X]*CLUE(I,J)
    X=X+1
  ENDFOR
Z1=Z1+1
ENDFOR
X=1
FOR I = 1 TO PARANO-1
  FOR J= I+1 TO PARANO
    RAW1[I,J]=INDICESF[I,X1,X]*CLUE(I,J)
    X=X+1
  ENDFOR
ENDFOR
TOTDRIS1[X1]=0; TOTDRIS2[X1]=0; TOTD=0
FOR G= 1 TO PARANO
  DRIS1[X1,Y]=0
  FOR H= 1 TO PARANO-1
    DRIS1[X1,Y]=DRIS1[X1,Y]+RAW1[G,H]
  ENDFOR
  TOTD1=ABS(DRIS1[X1,Y]); TOTD=TOTD+TOTD1; Y=Y+1
  IF Y> PARANO
    Y=1
  ENDFOR
ENDFOR
TOTDRIS1[X1]=TOTD
ENDFOR
DECLARE RAW2[PARANO, PARANO], TOTDRIS3[LOWPOP], DRIS2[LOWPOP, PARANO+1], TOTDRIS4[LOWPOP]
A1=0; Y=1; Z=1
24
FOR X1 = 1 TO LOWPOP
X1 = 1; Z1 = 1
FOR I = 1 TO PARANO
 FOR J = Z1 TO PARANO - 1
     RAW2[J,J] = INDICESF2[X1, X2] * CLUE[I, J]
     X2 = X2 + 1
     ENDFOR
 Z1 = Z1 + 1
ENDFOR
X1 = 1
FOR I = 1 TO PARANO - 1
 FOR J = I + 1 TO PARANO
     RAW2[J,J] = INDICESF2[X1, X2] * CLUE[I, J]
     X2 = X2 + 1
     ENDFOR
ENDFOR
ENDFOR

TOTDRIS[X1] = 0; TOTDRIS[X1] = 0; TOTD = 0
FOR G = 1 TO PARANO
 DRIS2[X1, Y] = 0
 FOR H = 1 TO PARANO - 1
     DRIS2[X1, Y] = DRIS2[X1, Y] + RAW2[G, H]
     ENDFOR
 DRIS2[X1, Y] = DRIS2[X1, Y] / PARANO; Y = Y + 1
IF Y = PARANO
 Y = 1
ENDIF
ENDFOR
ENDFOR

MDRIS INDICES CALCULATION

DECLARE MIND1[HIGHPOP, PARANO], MMEAN1[PARANO], MIND2[LOWPOP, PARANO], MMEAN2[PARANO], MIND3[HIGHPOP, PARANO], MIND4[LOWPOP, PARANO], MDRIS2[LOWPOP, PARANO + 3], TOTMDRIS2[LOWPOP], YDRIS1[HIGHPOP], YDRIS2[LOWPOP], SUMMM2[PARANO], SSSQ2[PARANO], VARR1[PARANO], CVV2[PARANO], SSSD2[PARANO], MMEAN1[PARANO]
K = 1; MMEAN1 = 0; SSSS1 = 0; SUMM1 = 0
FOR I = 1 TO PARANO
 FOR J = 1 TO HIGHPOP
     MIND1[I,J] = HIGH[I,J] / DAT1[K, I]; MMEAN1 = MMEAN1 + MIND1[I,J]; SUMM1 = SUMM1 + MIND1[I,J]
     SSSS1 = SSSS1 + (MIND1[I,J] * MIND1[I,J]); K = K + 1
     ENDFOR
 SUMMM2[I] = SUMM1; MMEAN1[I] = MMEAN1[HIGHPOP] / SSSQ2[I]; SSSS1 = VARR1[I] = ABS(VARR1[I]); SSSD2[I] = SQRT(VARR1[I]); CVV2[I] = (SQRT(VARR1[I]) / MMEAN1[I])
 MMEAN1 = 0; SUMM1 = 0; SSSS1 = 0; K = 1
ENDFOR
K = HIGHPOP + 1; MMEAN2 = 0
FOR I = 1 TO PARANO
 FOR J = 1 TO LOWPOP
     MIND2[I,J] = LOW[I,J] / DAT1[K, I]; K = K + 1; MMEAN2 = MMEAN2 + MIND2[I,J]
     ENDFOR
 K = HIGHPOP + 1; MMEANN2 = MMEAN2[LOWPOP]; MMEAN2 = 0
ENDFOR
K = 1; TMDRIS = 0; TMDRIS1 = 0; YYDRIS1 = 0

25
FOR J= 1 TO HIGHPOP
    FOR I= 1 TO PARANO
        IF MIND1[I,J] > M MEAN 1[K]
            MIND3[I,J]=((MIND1[I,J]/ABS(M MEAN 1[K]))-1)*1000/CVV2[I])
            MDRIS1[I,J]=DRIS1[I,J]+MIND3[I,J]/(PARANO+1)
        ELSE
            MIND3[I,J]=((1-ABS(M MEAN 1[K])/MIND1[I,J]))*1000/CVV2[I])
            MDRIS1[I,J]=DRIS1[I,J]+MIND3[I,J]/(PARANO+1)
        ENDIF
        K=K+1
        TMDRIS1=ABS(MDRIS1[I,J]); TMDRIS=TMDRIS+TMDRIS1; YYDRIS1=YYDRIS1-MIND3[I,J]
    ENDFOR
    YDRIS1[I]=YYDRIS1; K=1; TOTMDRIS1[I]=TMDRIS; TMDRIS=0; YYDRIS1=0
ENDFOR
FOR I= 1 TO HIGHPOP
    MDRIS1[I, PARANO+1]=YYDRIS1[I]/(PARANO+1)
    MDRIS1[I, PARANO+2]=TOTMDRIS1[I]/(ABS(YDRIS1[I])/PARANO+1)
    MDRIS1[I, PARANO+3]=HIGH1[I, PARANO+1]
ENDFOR
ENDB

TMDRIS=0; TMDRIS=0; YYDRIS2=0; K=1
FOR J= 1 TO LOWPOP
    FOR I= 1 TO PARANO
        IF MIND2[J,J] > MM EAN 2[K]
            MIND4[I,J]=((MIND2[J,J]/MM EAN 2[K])-1)*1000/CVV2[I])
            MDRIS2[J,J]=DRIS2[J,J]+MIND4[I,J]/(PARANO+1)
        ELSE
            MIND4[I,J]=((1-ABS(M MEAN 1[K])/MIND2[J,J]))*1000/CVV2[I])
            MDRIS2[J,J]=DRIS2[J,J]+MIND4[I,J]/(PARANO+1)
        ENDIF
        K=K+1; YYDRIS2=YYDRIS2-MIND4[I,J]; TMDRIS2=ABS(MDRIS2[I,J]); TMDRIS=TMDRIS+TMDRIS2
    ENDFOR
    YDRIS2[J]=YYDRIS2; K=1; TOTMDRIS2[J]=TMDRIS; TMDRIS=0; YYDRIS2=0
ENDFOR
FOR I= 1 TO LOWPOP
    MDRIS2[I,PARANO+1]=YYDRIS2[I]/(PARANO+1);
    MDRIS2[I, PARANO+2]=TOTMDRIS2[I]/(ABS(YDRIS2[I])/PARANO+1)
    MDRIS2[I, PARANO+3]=LOW1[I, PARANO+1]
ENDFOR
SELE 16
APPEND FROM ARRAY MDRIS2
COPY TO 8T16 TYPE XLS
CLEAR ALL; CLEAR; CLOSE ALL
CHAPTER - IV

CND Model (COMPOSITIONAL NUTRIENT DIAGNOSIS)
i. Brief description of methodology – CND model

The data are processed similar way as that of DRIS for Compositional Nutritional Diagnosis (CND) approach (Parent and Dafir, 1992). In this model the full composition array for D nutrient compositions in plant tissues can be described by the following simplex S^D contained to 100 %

\[ S^D = \{ (N, P, K, ..., R) ; N > 0, P > 0, K > 0, ..., R > 0; N + P + K + ... + R = 100\% \} \]

Where 100 % is the dry matter content; N, P, K, are the nutrient concentrations and R is the filling value between 100 % and sum of the nutrients concentrations.

Nutrient concentration is corrected by geometric mean, G of all the D components including R.

\[ G = (N \times P \times K \times ... \times R)^{1/D} \]

The row centered log ratios are generated as follows:

\[ V_N = \ln \left( \frac{N}{G} \right), ... V_{mn} = \ln \left( \frac{M_n}{G} \right) \]

The \( V_N^* \) to \( V_{mn}^* \) and \( SD_N^* \) to \( SD_{mn}^* \) are the CND norms (indicated by asterisks) i.e. mean and standard deviation of each row centered log ratio in the high yielding sub-population. The standardized variables \( \frac{V_N - V_N^*}{SD_N^*} \) to \( \frac{V_{mn} - V_{mn}^*}{SD_{mn}^*} \) are the CND nutrient indices which are CND analogs of the DRIS nutrient indices

\[ I_N = \frac{V_N - V_N^*}{SD_N^*}, ... I_{mn} = \frac{V_{mn} - V_{mn}^*}{SD_{mn}^*} \]

Independent values for \( V_N \) to \( V_{2n} \) were introduced in the equation for diagnostic purpose.

These following approaches are computed using the package developed in VISUAL FOXPRO 98, the algorithm of the program is given below.
ii. Algorithm for CND nutrient model

SET ENVIRONMENT

INPUT VARIABLE DECLARATION

GET
- Input file(source file : EXCEL input FILE), Sample number,
  no of parameters/nutrients studied, Enter output file

CREATE database tables

DECLARE ARRAYS wholefile(sample number, total parimeters +1), class sum of square (sample no),
  class sum of square1 (sample no), class sum of square2 (sample no),
  class sum of square3 (sample no), class sum of square4 (sample no)....

DECLARE Memory variables

USE whole data file

APPEND FROM input file

COPY TO ARRAY whole file [sample number, total parameters +1]

SORT ARRAY -descending order of yield data  e.g. (ASORT (wholefile, 1, -1,1))

APPEND FROM ARRAY whole file database TO whole file array
  - Critical value calculation for yield cut off using Cate and Nelson's statistical class sum of
    square technique
  - Creation of high and low population nutrient data arrays based on the cut off derived from
    above method
  - Calculation of Geometric mean (G) = (N x P x K x .... x R)\(^{1/n}\) where R = filling value
  - CND norms (V\(_N\), V\(_P\), ...V\(_{MN}\) and SD\(_N\), SD\(_P\),...SD\(_{MN}\)) were calculated
  - CND indices (l\(_N\), l\(_P\), ...l\(_{MN}\)) were computed which are analogs to DRIS nutrient indices
  - Calculation, deficiency cut off, low level cut off, optimum level cut off, and high or sufficient
    level cutoff for both high and low population file
  - Transferring data from array to subsequent DRIS index EXCEL files

CLOSE ALL DATABASES

CLEAR ALL memory variables
iii. Flow chart of the computer program – CND model

- Database developed in Excel with uniform units of the data for input

- Data acquisition from database
- Set Environment
- Declare arrays and variables
- Create database tables

- Yield cutoff using CV (using Cate & Nelson method)
- Calculation of indices
- Creation high low population database
- Calculation of Geometric mean with R- filling factor
- Calculation of CND norms (\(V_n, V_p, \ldots V_m, SD_n, SD_p, SD_m\))
- Calculation of deficiency cutoff, optimum level and sufficiency cutoff values

- Transferring data of CND indices into EXCEL file
- Close all databases and memory variable

DATA PROCESSING

DATA OUTPUT

Microsoft EXCEL
iv. Computer program - CND model

CLEAR; CLEAR ALL; CLOSE ALL
SET SCOREBOARD OFF; SET STATUS OFF; SET TALK OFF; SET CONFIRM OFF; SET SAFETY OFF; CLOSE ALL; CLEAR

F1 = SPACE(20); T1=SPACE(20); T2=SPACE(20); T3=SPACE(20); T4=SPACE(20); T5=SPACE(20); T6=SPACE(20); T7=SPACE(20)
XX1=SPACE(40)
STORE 0 TO SAMPLENO, PARANO, INDO

@ 2, 10 SAY "FILE NAME"
@ 2, 30 GET F1

@ 4, 10 SAY "ENTER TOTAL NUMBER OF SAMPLES"
@ 4, 70 GET SAMPLENO

@ 5, 10 SAY "ENTER NO OF PARAMETERS"
@ 6, 60 GET PARANO
@ 8, 10 SAY "ENTER OUTPUT FILE NAME"
@ 8, 60 GET XX1
READ
T1=ALLTRIM(XX1)+"CWF"
T6=ALLTRIM(XX1)+"CLIN"
T7=ALLTRIM(XX1)+"CNORMS"

CREATE TABLE &T1 (Y N(9,2), N N(9,4), P N(9,4), K N(9,5), CA N(9,5), MG N(9,5), NA N(9,5), S N(9,5), B N(9,5), ZN N(9,5), CU N(9,5), FE N(9,5), MN N(9,5), FNO N(4),
RQ N(4,2))
CREATE TABLE &T6 (N N(10,7), P N(10,7), K N(10,7), CA N(10,7), MG N(9,7), NA N(9,7), S N(9,7), B N(9,7), ZN N(15,10), CU N(15,10), FE N(15,10), MN N(15,10))
CREATE TABLE &T7 (MEAN N(9,4), DEF N(9,4), LOW N(9,5), MED N(9,5), EXC N(9,5))

DECLARE DAT1[SAMPLENO, 17], CSS1[SAMPLENO], CSS2[SAMPLENO], CSS3[SAMPLENO], CSS4[SAMPLENO], R2[SAMPLENO]
DECLARE TX1[SAMPLENO], TX2[SAMPLENO]
STORE 0 TO LLLL, SS1, SS2, SS3, SS4, SS9, SS10, CAL1, i, LL1, XX2, AA3, AA4, XX5, AA5, AA6, AA7, AA9, ZZZ, MAX1
STORE 0 TO SD11, G11, SSS1, SUM1, VNY1, L112, KKK, LLLLL

SELECT 1
USE &T1 ALIAS X1; ZAP
APPEND FROM &F1 DELIMITED WITH TAB
COPY TO ARRAY DAT1
ASORT DAT1, 1, -1,1
CLEAR; ZAP
APPEND FROM ARRAY DAT1
CLOSE ALL

SELECT 6
USE &T6 ALIAS X6; ZAP

SELECT 7
USE &T7 ALIAS X7; ZAP

CALCULATION AND SELECTION OF CUT OFF FOR POPULATION

TT=0; SS=0
FOR i=1 TO SAMPLENO-1
   TT=TT+DAT1[i,1]; SS=SS+(DAT1[i,1]*DAT1[i,1]); TX1[i]=TT; CSS1[i]=SS
ENDFOR
TT=TT+DAT[1][SAMPLENO,1]; SS=SS+DAT[1][SAMPLENO,1]+DAT[1][SAMPLENO,1]; TX[1][SAMPLENO]=TT
CSS[1][SAMPLENO]=SS; TSS=CSS[1][SAMPLENO]+TX[1][SAMPLENO]; TX[1][SAMPLENO]=TX[1][SAMPLENO];

FOR I=2 TO SAMPLENO-1
    TX[2][I]=TX[1][SAMPLENO]-TX[1][I]; CSS[2][I]=CSS[1][SAMPLENO]-CSS[1][I]+TX[1][I]*TX[1][I];
    R2[I]=R2[I]; TSS=TSS*100
ENDFOR

CUT=1

FOR I = 2 TO SAMPLENO-1
    IF R2[I+1] > R2[I]
        CUT=CUT+1
    ELSE
        EXIT
    ENDIF
ENDFOR

DECLARE HIGH1[CUT, PARANO], LOW1[SAMPLENO-CUT, PARANO]
LOWPOP=SAMPLENO-CUT, HIGHPOP=CUT
FOR KKK=1 TO PARANO
    FOR I = 1 TO HIGHPOP
        HIGH1[I][KKK]=DAT[I][KKK]
    ENDFOR
ENDFOR

KKK=0; X=1
FOR I = 1 TO PARANO
    FOR J = HIGHPOP+1 TO SAMPLENO
        LOW1[J][I]=DAT[J][I]
        IF X=SAMPLENO-(HIGHPOP)
            X=1
        ELSE
            X=X+1
        ENDIF
    ENDFOR
ENDFOR

DECLARE SD1[HIGHPOP], SD2[LOWPOP], G1[HIGHPOP], G2[LOWPOP]
SD11=0; G11=1
FOR J = 1 TO HIGHPOP
    FOR I = 1 TO PARANO
        SD11=SD11+HIGH1[I][J]; G11=G11*HIGH1[I][J]
    ENDFOR
    SD11=100-SD11; LLLL=1/(PARANO+1); G1[J]=((SD1[J]*G11)*LLLL); SD11=0; G11=1
ENDFOR

G11=1; SD11=0
FOR J = 1 TO LOWPOP
    FOR I = 1 TO PARANO
        SD11=SD11+LOW1[I][J]; G11=G11*LOW1[I][J]
    ENDFOR
    SD2[J]=100-SD11; LLLL=ABS(1/(PARANO+1)); G2[J]=((SD2[J]*G11)*LLLL); G11=1; SD11=0
ENDFOR

33
DECLARE SUMM1[PARANO], SSQ1[PARANO], VAR1[PARANO], CV1[PARANO], SSD1[PARANO], MEAN1[PARANO], VN[HIGHPOP, PARANO], VNZ[LOWPOP, PARANO], N1[HIGHPOP, PARANO], N2[LOWPOP, PARANO], MN[PARANO], VNNORM[PARANO, 6], PARAM1[PARANO, 5]

X=1; Y=0; SUM1=0

FOR J= 1 TO PARANO
  FOR J= 1 TO HIGHPOP
    L1=HIGH1[J]/G1[J]; VN[J]=LOG(L1); SUM1=SUM1+VN[J]; SSS1=SSS1+(VN[J]*VN[J])
    ENDFOR
  
  SUMM1[J]=SUM1; MEAN1[J]=SUMM1[J]/HIGHPOP; SSQ1[J]=SSS1
  VAR1[J]=(HIGHPOP*SSQ1[J]-SUMM1[J]*SUMM1[J])/(HIGHPOP*(HIGHPOP-1))
  SUM1=0; SSS1=0
  FOR J= 1 TO LOWPOP
    L2=LOW1[J]/G2[J]; VNZ[J]=LOG(L2); SUM1=SUM1+VNZ[J]; SSS1=SSS1+(VNZ[J]*VNZ[J])
    ENDFOR
  X=X+1; SUM1=0; SSS1=0
  ENDFOR

FOR I= 1 TO PARANO
  KKK=0
  FOR J= 1 TO HIGHPOP
    N1[J]=((VN[J]-MEAN1[J])/SSD1[J]); KKK = KKK + G1[J]
    ENDFOR
  KKK=KKK / HIGHPOP
  VNNORM[I]=(MEAN1[I])^2*(2.72*VNNORM[I])^2*KKK; PARAM1[I, 1]=(2.72*VNNORM[I])^2*KKK; PARAM1[I, 2]=(2.72*VNNORM[I])^2*(2.67*(SSD1[I]))*KKK
  PARAM1[I, 3]=(2.72*VNNORM[I])^2*(1.33*(SSD1[I]))*KKK; PARAM1[I, 4]=(2.72*VNNORM[I])^2*(1.33*(SSD1[I]))*KKK
  PARAM1[I, 5]=(2.72*VNNORM[I])^2*(2.67*(SSD1[I]))*KKK
  ENDFOR

FOR I= 1 TO PARANO
  FOR J= 1 TO LOWPOP
    N2[J]=((VNZ[J]-MEAN1[J])/SSD1[J])
    ENDFOR
  ENDFOR

SELE 6
APPEND FROM ARRAY N2
COPY TO &T6 TYPE XLS
SELE 7
APPEND FROM ARRAY PARAM1
COPY TO &T7 TYPE XLS
USE; CLOSE ALL; CLEAR; RETURN
CHAPTER V

OUTPUT DISPLAY OF THE MODEL ON PROGRAM EXECUTION
Nutrient database is to be created in Excel with common units as given in the figure.

<table>
<thead>
<tr>
<th>Yield</th>
<th>N%</th>
<th>P%</th>
<th>K</th>
<th>Ca</th>
<th>Mg</th>
<th>Na</th>
<th>S percent</th>
<th>boron</th>
<th>Zn</th>
<th>Cu</th>
<th>Fe</th>
<th>Mn</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>31.75</td>
<td>2.284</td>
<td>0.79</td>
<td>2.66</td>
<td>1.73</td>
<td>0.3</td>
<td>0.347</td>
<td>0.699</td>
<td>0.0194</td>
<td>0.0065</td>
<td>0.0063</td>
<td>0.0524</td>
</tr>
<tr>
<td>4</td>
<td>31.20</td>
<td>2.308</td>
<td>0.754</td>
<td>2.014</td>
<td>2.18</td>
<td>0.27</td>
<td>0.355</td>
<td>0.61</td>
<td>0.0074</td>
<td>0.0068</td>
<td>0.0038</td>
<td>0.0782</td>
</tr>
<tr>
<td>5</td>
<td>29.75</td>
<td>2.372</td>
<td>0.514</td>
<td>3.112</td>
<td>2.4</td>
<td>0.28</td>
<td>0.357</td>
<td>0.796</td>
<td>0.0148</td>
<td>0.0079</td>
<td>0.0026</td>
<td>0.0857</td>
</tr>
<tr>
<td>6</td>
<td>28.00</td>
<td>2.138</td>
<td>1.29</td>
<td>3.265</td>
<td>1.61</td>
<td>0.28</td>
<td>0.339</td>
<td>0.865</td>
<td>0.0087</td>
<td>0.0038</td>
<td>0.0885</td>
<td>0.0046</td>
</tr>
<tr>
<td>7</td>
<td>26.86</td>
<td>2.347</td>
<td>0.41</td>
<td>3.252</td>
<td>2.13</td>
<td>0.18</td>
<td>0.345</td>
<td>0.41</td>
<td>0.0131</td>
<td>0.0046</td>
<td>0.0826</td>
<td>0.0026</td>
</tr>
<tr>
<td>8</td>
<td>27.08</td>
<td>2.142</td>
<td>0.479</td>
<td>3.218</td>
<td>2.34</td>
<td>0.26</td>
<td>0.311</td>
<td>0.737</td>
<td>0.0162</td>
<td>0.0023</td>
<td>0.0704</td>
<td>0.0020</td>
</tr>
<tr>
<td>9</td>
<td>27.83</td>
<td>2.207</td>
<td>0.498</td>
<td>3.189</td>
<td>2.1</td>
<td>0.23</td>
<td>0.340</td>
<td>0.316</td>
<td>0.0094</td>
<td>0.0046</td>
<td>0.0571</td>
<td>0.0030</td>
</tr>
<tr>
<td>10</td>
<td>26.78</td>
<td>2.287</td>
<td>0.399</td>
<td>3.173</td>
<td>1.89</td>
<td>0.39</td>
<td>0.345</td>
<td>0.279</td>
<td>0.0084</td>
<td>0.0083</td>
<td>0.0032</td>
<td>0.0648</td>
</tr>
<tr>
<td>11</td>
<td>26.07</td>
<td>2.390</td>
<td>0.401</td>
<td>2.873</td>
<td>2.02</td>
<td>0.4</td>
<td>0.350</td>
<td>0.809</td>
<td>0.0187</td>
<td>0.0000</td>
<td>0.0068</td>
<td>0.0763</td>
</tr>
<tr>
<td>12</td>
<td>25.83</td>
<td>2.174</td>
<td>0.62</td>
<td>2.93</td>
<td>2.3</td>
<td>0.42</td>
<td>0.285</td>
<td>0.289</td>
<td>0.0112</td>
<td>0.0053</td>
<td>0.0034</td>
<td>0.0977</td>
</tr>
<tr>
<td>13</td>
<td>25.66</td>
<td>2.113</td>
<td>0.366</td>
<td>2.894</td>
<td>1.98</td>
<td>0.38</td>
<td>0.328</td>
<td>0.647</td>
<td>0.0002</td>
<td>0.0051</td>
<td>0.0089</td>
<td>0.0030</td>
</tr>
<tr>
<td>14</td>
<td>24.85</td>
<td>2.051</td>
<td>0.708</td>
<td>2.503</td>
<td>1.68</td>
<td>0.38</td>
<td>0.322</td>
<td>0.51</td>
<td>0.0031</td>
<td>0.0009</td>
<td>0.0077</td>
<td>0.0022</td>
</tr>
<tr>
<td>15</td>
<td>23.38</td>
<td>2.178</td>
<td>0.62</td>
<td>2.611</td>
<td>2.13</td>
<td>0.28</td>
<td>0.335</td>
<td>0.364</td>
<td>0.0155</td>
<td>0.0041</td>
<td>0.0031</td>
<td>0.0836</td>
</tr>
<tr>
<td>16</td>
<td>24.32</td>
<td>1.938</td>
<td>0.727</td>
<td>2.604</td>
<td>1.91</td>
<td>0.30</td>
<td>0.309</td>
<td>0.497</td>
<td>0.0111</td>
<td>0.0046</td>
<td>0.0049</td>
<td>0.0713</td>
</tr>
<tr>
<td>17</td>
<td>23.72</td>
<td>2.243</td>
<td>1.468</td>
<td>2.688</td>
<td>2.05</td>
<td>0.17</td>
<td>0.342</td>
<td>0.474</td>
<td>0.0156</td>
<td>0.0058</td>
<td>0.0062</td>
<td>0.0874</td>
</tr>
<tr>
<td>18</td>
<td>23.50</td>
<td>1.851</td>
<td>0.367</td>
<td>2.672</td>
<td>1.71</td>
<td>0.34</td>
<td>0.279</td>
<td>0.365</td>
<td>0.0152</td>
<td>0.0039</td>
<td>0.0022</td>
<td>0.0730</td>
</tr>
<tr>
<td>19</td>
<td>22.73</td>
<td>1.857</td>
<td>0.723</td>
<td>2.660</td>
<td>2.32</td>
<td>0.5</td>
<td>0.302</td>
<td>0.868</td>
<td>0.0096</td>
<td>0.0047</td>
<td>0.0041</td>
<td>0.0708</td>
</tr>
<tr>
<td>20</td>
<td>22.49</td>
<td>1.859</td>
<td>0.547</td>
<td>2.688</td>
<td>1.91</td>
<td>0.31</td>
<td>0.260</td>
<td>0.343</td>
<td>0.0162</td>
<td>0.0018</td>
<td>0.0005</td>
<td>0.0538</td>
</tr>
<tr>
<td>21</td>
<td>22.28</td>
<td>1.789</td>
<td>0.87</td>
<td>2.663</td>
<td>1.87</td>
<td>0.22</td>
<td>0.291</td>
<td>0.644</td>
<td>0.0118</td>
<td>0.0004</td>
<td>0.0070</td>
<td>0.0038</td>
</tr>
<tr>
<td>22</td>
<td>22.28</td>
<td>2.352</td>
<td>0.701</td>
<td>2.694</td>
<td>1.47</td>
<td>0.46</td>
<td>0.355</td>
<td>0.897</td>
<td>0.0085</td>
<td>0.0065</td>
<td>0.0047</td>
<td>0.0882</td>
</tr>
<tr>
<td>23</td>
<td>21.05</td>
<td>1.719</td>
<td>0.708</td>
<td>2.816</td>
<td>1.30</td>
<td>0.24</td>
<td>0.283</td>
<td>0.440</td>
<td>0.0003</td>
<td>0.0081</td>
<td>0.0050</td>
<td>0.0416</td>
</tr>
<tr>
<td>24</td>
<td>21.41</td>
<td>1.788</td>
<td>0.897</td>
<td>2.775</td>
<td>1.76</td>
<td>0.33</td>
<td>0.291</td>
<td>0.381</td>
<td>0.0214</td>
<td>0.0047</td>
<td>0.0046</td>
<td>0.0564</td>
</tr>
<tr>
<td>25</td>
<td>21.09</td>
<td>1.845</td>
<td>0.795</td>
<td>2.423</td>
<td>1.81</td>
<td>0.27</td>
<td>0.275</td>
<td>0.813</td>
<td>0.0149</td>
<td>0.0082</td>
<td>0.0039</td>
<td>0.0527</td>
</tr>
<tr>
<td>26</td>
<td>20.46</td>
<td>1.810</td>
<td>0.564</td>
<td>2.521</td>
<td>1.57</td>
<td>0.37</td>
<td>0.293</td>
<td>0.458</td>
<td>0.0234</td>
<td>0.0023</td>
<td>0.0050</td>
<td>0.0540</td>
</tr>
<tr>
<td>27</td>
<td>19.81</td>
<td>1.654</td>
<td>0.488</td>
<td>2.332</td>
<td>1.43</td>
<td>0.4</td>
<td>0.276</td>
<td>0.868</td>
<td>0.0112</td>
<td>0.0078</td>
<td>0.0042</td>
<td>0.0476</td>
</tr>
<tr>
<td>28</td>
<td>19.78</td>
<td>1.659</td>
<td>0.393</td>
<td>2.578</td>
<td>1.72</td>
<td>0.35</td>
<td>0.278</td>
<td>0.353</td>
<td>0.0196</td>
<td>0.0082</td>
<td>0.0044</td>
<td>0.0429</td>
</tr>
<tr>
<td>29</td>
<td>19.36</td>
<td>1.638</td>
<td>0.227</td>
<td>2.412</td>
<td>1.65</td>
<td>0.28</td>
<td>0.290</td>
<td>0.35</td>
<td>0.0190</td>
<td>0.0054</td>
<td>0.0040</td>
<td>0.0417</td>
</tr>
<tr>
<td>30</td>
<td>17.87</td>
<td>1.056</td>
<td>0.877</td>
<td>2.398</td>
<td>1.7</td>
<td>0.27</td>
<td>0.292</td>
<td>0.35</td>
<td>0.0214</td>
<td>0.0035</td>
<td>0.0056</td>
<td>0.0540</td>
</tr>
</tbody>
</table>
End of the creation of the database the top row with label should be deleted leaving only the datas in the file.
Check that the database holds the yield in the front (column A) and the survey lot no. in the last column.

To get the overall nutrient indices for the survey area, the mean of the survey data should be appended to the last row in the database as last record (it should be added to the no of samples)
Save the file as Text (Tab delimited)
The data file will be a text file with the order of data as given below (but without the header row):

<table>
<thead>
<tr>
<th></th>
<th>N</th>
<th>P</th>
<th>K</th>
<th>Ca</th>
<th>Mg</th>
<th>Na</th>
<th>S</th>
<th>B</th>
<th>Zn</th>
<th>Cu</th>
<th>Fe</th>
</tr>
</thead>
<tbody>
<tr>
<td>22.49</td>
<td>1.69</td>
<td>0.55</td>
<td>2.88</td>
<td>1.91</td>
<td>0.30</td>
<td>0.28</td>
<td>0.182</td>
<td>0.0182</td>
<td>0.0182</td>
<td>0.0182</td>
<td>0.0182</td>
</tr>
<tr>
<td>12.79</td>
<td>1.37</td>
<td>0.53</td>
<td>2.04</td>
<td>1.39</td>
<td>0.24</td>
<td>0.24</td>
<td>0.1111</td>
<td>0.0111</td>
<td>0.0111</td>
<td>0.0111</td>
<td>0.0111</td>
</tr>
<tr>
<td>24.32</td>
<td>1.94</td>
<td>0.73</td>
<td>2.58</td>
<td>1.91</td>
<td>0.36</td>
<td>0.36</td>
<td>0.0194</td>
<td>0.0194</td>
<td>0.0194</td>
<td>0.0194</td>
<td>0.0194</td>
</tr>
<tr>
<td>31.75</td>
<td>2.28</td>
<td>0.79</td>
<td>2.66</td>
<td>1.73</td>
<td>0.30</td>
<td>0.30</td>
<td>0.0140</td>
<td>0.0140</td>
<td>0.0140</td>
<td>0.0140</td>
<td>0.0140</td>
</tr>
<tr>
<td>26.76</td>
<td>2.27</td>
<td>0.79</td>
<td>3.17</td>
<td>1.89</td>
<td>0.44</td>
<td>0.44</td>
<td>0.0085</td>
<td>0.0085</td>
<td>0.0085</td>
<td>0.0085</td>
<td>0.0085</td>
</tr>
<tr>
<td>22.26</td>
<td>2.35</td>
<td>0.79</td>
<td>2.88</td>
<td>1.47</td>
<td>0.34</td>
<td>0.34</td>
<td>0.0084</td>
<td>0.0084</td>
<td>0.0084</td>
<td>0.0084</td>
<td>0.0084</td>
</tr>
</tbody>
</table>

Note: Yield values are given in each column.
<table>
<thead>
<tr>
<th>Field no</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.55</td>
<td>2.88</td>
<td>1.91</td>
<td>0.30</td>
<td>0.28</td>
<td>0.84</td>
<td>0.0182</td>
<td>0.0108</td>
<td>0.0055</td>
<td>0.0638</td>
<td>0.0014</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0.53</td>
<td>2.84</td>
<td>1.39</td>
<td>0.24</td>
<td>0.24</td>
<td>0.75</td>
<td>0.0158</td>
<td>0.0096</td>
<td>0.0024</td>
<td>0.0362</td>
<td>0.0022</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>0.73</td>
<td>2.58</td>
<td>1.91</td>
<td>0.36</td>
<td>0.40</td>
<td>0.40</td>
<td>0.0111</td>
<td>0.0046</td>
<td>0.0049</td>
<td>0.0713</td>
<td>0.0051</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>0.79</td>
<td>2.66</td>
<td>1.73</td>
<td>0.30</td>
<td>0.36</td>
<td>0.40</td>
<td>0.0194</td>
<td>0.0065</td>
<td>0.0053</td>
<td>0.0524</td>
<td>0.0067</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>0.40</td>
<td>3.17</td>
<td>1.89</td>
<td>0.40</td>
<td>0.34</td>
<td>0.28</td>
<td>0.0084</td>
<td>0.0083</td>
<td>0.0032</td>
<td>0.0646</td>
<td>0.0058</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>0.70</td>
<td>2.80</td>
<td>1.47</td>
<td>0.48</td>
<td>0.35</td>
<td>0.70</td>
<td>0.0085</td>
<td>0.0065</td>
<td>0.0047</td>
<td>0.0952</td>
<td>0.0059</td>
<td>6</td>
</tr>
<tr>
<td>7</td>
<td>0.70</td>
<td>2.80</td>
<td>1.47</td>
<td>0.48</td>
<td>0.35</td>
<td>0.70</td>
<td>0.0085</td>
<td>0.0065</td>
<td>0.0047</td>
<td>0.0952</td>
<td>0.0059</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>0.77</td>
<td>2.82</td>
<td>1.29</td>
<td>0.20</td>
<td>0.23</td>
<td>0.55</td>
<td>0.0140</td>
<td>0.0058</td>
<td>0.0020</td>
<td>0.0254</td>
<td>0.0058</td>
<td>8</td>
</tr>
<tr>
<td>9</td>
<td>1.46</td>
<td>2.65</td>
<td>2.05</td>
<td>0.17</td>
<td>0.34</td>
<td>0.45</td>
<td>0.0093</td>
<td>0.0081</td>
<td>0.0050</td>
<td>0.0416</td>
<td>0.0065</td>
<td>9</td>
</tr>
<tr>
<td>10</td>
<td>0.70</td>
<td>2.60</td>
<td>1.70</td>
<td>0.40</td>
<td>0.25</td>
<td>0.63</td>
<td>0.0156</td>
<td>0.0056</td>
<td>0.0052</td>
<td>0.0670</td>
<td>0.0129</td>
<td>10</td>
</tr>
<tr>
<td>11</td>
<td>1.06</td>
<td>2.0</td>
<td>1.22</td>
<td>0.39</td>
<td>0.26</td>
<td>0.71</td>
<td>0.0141</td>
<td>0.0086</td>
<td>0.0077</td>
<td>0.0708</td>
<td>0.0129</td>
<td>11</td>
</tr>
<tr>
<td>12</td>
<td>1.01</td>
<td>2.04</td>
<td>1.41</td>
<td>0.24</td>
<td>0.24</td>
<td>0.69</td>
<td>0.0142</td>
<td>0.0052</td>
<td>0.0033</td>
<td>0.0708</td>
<td>0.0072</td>
<td>12</td>
</tr>
<tr>
<td>13</td>
<td>0.47</td>
<td>2.08</td>
<td>1.74</td>
<td>0.38</td>
<td>0.23</td>
<td>0.67</td>
<td>0.0193</td>
<td>0.0057</td>
<td>0.0022</td>
<td>0.0237</td>
<td>0.0022</td>
<td>13</td>
</tr>
<tr>
<td>14</td>
<td>0.10</td>
<td>2.10</td>
<td>1.8</td>
<td>0.38</td>
<td>0.25</td>
<td>0.71</td>
<td>0.0140</td>
<td>0.0058</td>
<td>0.0020</td>
<td>0.0254</td>
<td>0.0058</td>
<td>14</td>
</tr>
<tr>
<td>15</td>
<td>0.46</td>
<td>2.16</td>
<td>1.06</td>
<td>0.22</td>
<td>0.26</td>
<td>0.69</td>
<td>0.0152</td>
<td>0.0039</td>
<td>0.0022</td>
<td>0.0730</td>
<td>0.0034</td>
<td>15</td>
</tr>
<tr>
<td>16</td>
<td>0.73</td>
<td>2.25</td>
<td>1.42</td>
<td>0.32</td>
<td>0.27</td>
<td>0.81</td>
<td>0.0149</td>
<td>0.0082</td>
<td>0.0039</td>
<td>0.0952</td>
<td>0.0051</td>
<td>16</td>
</tr>
<tr>
<td>17</td>
<td>0.72</td>
<td>2.42</td>
<td>1.61</td>
<td>0.27</td>
<td>0.27</td>
<td>0.81</td>
<td>0.0149</td>
<td>0.0082</td>
<td>0.0039</td>
<td>0.0952</td>
<td>0.0051</td>
<td>17</td>
</tr>
<tr>
<td>18</td>
<td>0.72</td>
<td>2.67</td>
<td>2.32</td>
<td>0.50</td>
<td>0.30</td>
<td>0.67</td>
<td>0.0099</td>
<td>0.0047</td>
<td>0.0041</td>
<td>0.0708</td>
<td>0.0072</td>
<td>18</td>
</tr>
<tr>
<td>19</td>
<td>0.57</td>
<td>2.57</td>
<td>1.71</td>
<td>0.34</td>
<td>0.28</td>
<td>0.37</td>
<td>0.0133</td>
<td>0.0059</td>
<td>0.0051</td>
<td>0.0708</td>
<td>0.0072</td>
<td>19</td>
</tr>
<tr>
<td>20</td>
<td>0.71</td>
<td>2.34</td>
<td>1.04</td>
<td>0.22</td>
<td>0.28</td>
<td>0.65</td>
<td>0.0155</td>
<td>0.0059</td>
<td>0.0051</td>
<td>0.0708</td>
<td>0.0072</td>
<td>20</td>
</tr>
<tr>
<td>21</td>
<td>0.36</td>
<td>2.58</td>
<td>1.72</td>
<td>0.39</td>
<td>0.28</td>
<td>0.32</td>
<td>0.0133</td>
<td>0.0059</td>
<td>0.0051</td>
<td>0.0708</td>
<td>0.0072</td>
<td>21</td>
</tr>
<tr>
<td>22</td>
<td>0.45</td>
<td>3.19</td>
<td>2.1</td>
<td>0.23</td>
<td>0.35</td>
<td>0.32</td>
<td>0.0133</td>
<td>0.0059</td>
<td>0.0051</td>
<td>0.0708</td>
<td>0.0072</td>
<td>22</td>
</tr>
<tr>
<td>23</td>
<td>0.87</td>
<td>2.85</td>
<td>1.87</td>
<td>0.22</td>
<td>0.29</td>
<td>0.54</td>
<td>0.0116</td>
<td>0.0042</td>
<td>0.0049</td>
<td>0.0762</td>
<td>0.0036</td>
<td>23</td>
</tr>
<tr>
<td>24</td>
<td>0.62</td>
<td>2.93</td>
<td>2.3</td>
<td>0.42</td>
<td>0.29</td>
<td>0.27</td>
<td>0.0112</td>
<td>0.0055</td>
<td>0.0034</td>
<td>0.0977</td>
<td>0.0039</td>
<td>24</td>
</tr>
<tr>
<td>25</td>
<td>0.61</td>
<td>2.19</td>
<td>0.99</td>
<td>0.31</td>
<td>0.27</td>
<td>0.66</td>
<td>0.0232</td>
<td>0.0055</td>
<td>0.0024</td>
<td>0.0377</td>
<td>0.0079</td>
<td>25</td>
</tr>
<tr>
<td>26</td>
<td>0.52</td>
<td>3.11</td>
<td>2.4</td>
<td>0.28</td>
<td>0.36</td>
<td>0.80</td>
<td>0.0146</td>
<td>0.0079</td>
<td>0.0026</td>
<td>0.0857</td>
<td>0.0022</td>
<td>26</td>
</tr>
<tr>
<td>27</td>
<td>0.62</td>
<td>2.61</td>
<td>2.13</td>
<td>0.28</td>
<td>0.33</td>
<td>0.35</td>
<td>0.0155</td>
<td>0.0041</td>
<td>0.0031</td>
<td>0.0635</td>
<td>0.0027</td>
<td>27</td>
</tr>
<tr>
<td>28</td>
<td>0.46</td>
<td>2.84</td>
<td>2.02</td>
<td>0.40</td>
<td>0.36</td>
<td>0.81</td>
<td>0.0187</td>
<td>0.0060</td>
<td>0.0066</td>
<td>0.0763</td>
<td>0.0027</td>
<td>28</td>
</tr>
<tr>
<td>29</td>
<td>0.69</td>
<td>2.26</td>
<td>1.17</td>
<td>0.26</td>
<td>0.54</td>
<td>0.41</td>
<td>0.0131</td>
<td>0.0086</td>
<td>0.0037</td>
<td>0.0825</td>
<td>0.0028</td>
<td>29</td>
</tr>
<tr>
<td>30</td>
<td>0.74</td>
<td>3.26</td>
<td>2.13</td>
<td>0.18</td>
<td>0.35</td>
<td>0.41</td>
<td>0.0131</td>
<td>0.0086</td>
<td>0.0037</td>
<td>0.0825</td>
<td>0.0028</td>
<td>30</td>
</tr>
<tr>
<td>31</td>
<td>0.55</td>
<td>2.14</td>
<td>1.38</td>
<td>0.37</td>
<td>0.24</td>
<td>0.55</td>
<td>0.0182</td>
<td>0.0083</td>
<td>0.0019</td>
<td>0.0454</td>
<td>0.0017</td>
<td>31</td>
</tr>
<tr>
<td>32</td>
<td>0.71</td>
<td>2.50</td>
<td>1.96</td>
<td>0.36</td>
<td>0.32</td>
<td>0.51</td>
<td>0.0081</td>
<td>0.0098</td>
<td>0.0036</td>
<td>0.0775</td>
<td>0.0022</td>
<td>32</td>
</tr>
</tbody>
</table>
COUNT INCLUDES THE AVERAGE SURVEY DATA

* ONLY THE NUTRIENT PARAMETERS TO BE COUNTED AND ENTERED
* THE YIELD AND FIELD NO WILL BE ACCOUNTED AUTOMATICALLY
<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Type</th>
<th>Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIAGNOSE</td>
<td>16KB</td>
<td>Microsoft Visual FoxPro</td>
<td>17/11/98 14:15</td>
</tr>
<tr>
<td>diagnoser.VCT</td>
<td>1KB</td>
<td>Microsoft Visual FoxPro</td>
<td>17/11/98 14:13</td>
</tr>
<tr>
<td>diagnoser.vcx</td>
<td>2KB</td>
<td>Microsoft Visual FoxPro</td>
<td>17/11/98 14:15</td>
</tr>
<tr>
<td>DIAGNOSE1</td>
<td>168KB</td>
<td>Application</td>
<td>17/11/98 14:10</td>
</tr>
<tr>
<td>disp.FXP</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>disp.prg</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Foxuser.dbf</td>
<td>20KB</td>
<td>Microsoft Visual FoxPro</td>
<td>17/11/98 18:29</td>
</tr>
<tr>
<td>Foxuser.fxt</td>
<td>19KB</td>
<td>Microsoft Visual FoxPro</td>
<td>17/11/98 18:28</td>
</tr>
<tr>
<td>grapescvmp.DBF</td>
<td>1KB</td>
<td>Microsoft Visual FoxPro</td>
<td>17/11/98 11:05</td>
</tr>
<tr>
<td>grapescvmp</td>
<td>2KB</td>
<td>Microsoft Visual FoxPro</td>
<td>17/11/98 11:05</td>
</tr>
<tr>
<td>grapescvmp.DBF</td>
<td>3KB</td>
<td>Microsoft Visual FoxPro</td>
<td>08/01/00 07:13</td>
</tr>
<tr>
<td>grapescvmp</td>
<td>4KB</td>
<td>Microsoft Excel Workbooks</td>
<td>08/01/00 07:13</td>
</tr>
<tr>
<td>grapescvmp.DBF</td>
<td>7KB</td>
<td>Microsoft Visual FoxPro</td>
<td>08/01/00 07:13</td>
</tr>
<tr>
<td>Dovapi.l</td>
<td></td>
<td>LIB File</td>
<td>20/05/98 00:00</td>
</tr>
<tr>
<td>Dovapi.l</td>
<td></td>
<td>Text Document</td>
<td>17/11/98 18:13</td>
</tr>
<tr>
<td>Dovapi.l</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dovapi.l</td>
<td>2KB</td>
<td>Text Document</td>
<td>16/11/98 23:53</td>
</tr>
<tr>
<td>Dovapi.l</td>
<td>48KB</td>
<td>LIB File</td>
<td>20/05/98 00:00</td>
</tr>
</tbody>
</table>

OUTPUT FILE AFTER ANALYSIS IN EXCEL (XLS) FORMAT
CLICK ON IT TO OPEN IN EXCEL AND VIEW THE INDICES FOR THE
WHOLE SURVEY POPULATION AND LOW YIELDING POPULATION
CHAPTER - VI

REFERENCES


