Predicting Food Insecurity

Narhan, J.S., Nylander, E., Ramarao, T.

3 May 2016

Abstract

The etiology as to why individuals go hungry has its roots in complex social, political and economic interactions that govern food production, purchase capability and entitlement to food. Many in the humanitarian industry recognize that understanding and redressing food insecurity entails understanding multi- dimensional variants. The limited availability of predictive tools, are one reason complicating the study of the problem which still exists in several countries such as Niger and Burundi. A set of Generalized Linear Models will be built to analyze the variables affecting the the Global Hunger Index score and the best model will be chosen to predict the food insecurity status.

Keywords: Food Security, GHI, Poisson, Multinomial, VIF

Introduction and Literature Review

The World Food Program has estimated that approximately 800 million people are undernourished in the world today. That represents almost one in nine people who do not consume enough food to lead a healthy and active life (United Nations Secretariat, 2016). Corroborating the findings that hunger remains a serious problem in today's world, a recent study (International Food Policy Research Institute (IFPRI), Welthungerhilfe (WHH) & Concern Worldwide, 2015), reported that 52 countries (see Figure 1) in 2015, saw the issue of food insecurity as a major problem, based on aggregate measures reported as the Global Hunger Index (GHI) score.

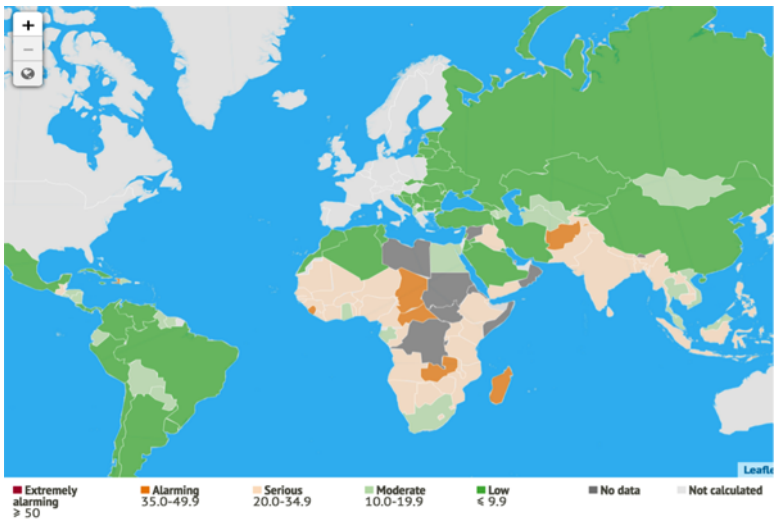


Fig 1: Countries with 2015 GHI Scores

Definitions of what constitutes food security are numerous. Following World War II, the definition of food security (FS) refereed to the incidence of famine and the resulting loss of life through starvation. Clearly this definition faltered as the dichotomy between increasing food production (via improved agricultural practices)

was being observed along side the continuing persistence of hunger and malnutrition. In 1996, the World Food Summit (WFS) presented four key dimensions around FS that are broadly accepted by many working in this field. These dimensions include:

- Availability to food
- Access to food
- Stability (production, food prices)
- Utilization practices surrounding food

This project aims to predict food insecurity based on indicators thought to be important determinants of FS. The work is similar to that of (Mbukwa, 2013) who proposed a methodology to predict FS in Tanzania. In that study, the author applied a logistic regression analysis against household level data to identify certain attributes that appeared predictive of food insecurity. These data included age and education levels of the heads of households. Unlike that study however, the current work differs in that Generalized Linear Models are to be used over the logistic regression analysis in the Tanzania study. Moreover, the current project is restricted to macro level predictors rather than the detailed household level information that Mbukwa was able to use.

In a similar vein, Gubert et al (Gubert, Benício, Da Silva, Da Costa Rosa, et al., 2010) made use of dis-aggregated data down to the municipal level within Brazil. The authors created multivariate logistic regression models to successfully predict FS within Brazilian municipalities. The advantage of both Mbukwa and Gubert et al's approaches reside in an ability to predict FS based on micro level, country-specific data. Access to such data can be difficult. Moreover, having models that enable prediction of food insecurity that are more generally applicable to multiple countries would certainly be beneficial in predicting levels of food insecurity. As such, this analysis looks to leverage broader, macro-level predictor variables, aligned to the aforementioned WFS 1996 dimensions of food insecurity.

Methodology

Data Acquisition

The data sets were mined from different agencies: The FAO (FAO Statistics, 2016) and the the Harvard University's dataverse platform (International Food Policy Research Institute (IFPRI), Welthungerhilfe (WHH) & Concern Worldwide, 2015). Data was acquired from these website during April 2016. FAO data was available in Microsoft Excel format, while data from the Global Hunger Index (GHI) was in PDF format and was scraped to put into digital format.

Data Exploration and Preperation

As noted in the introduction, there are numerous factors affecting the multidimensional nature of hunger. The indicators available from the FAO data set included undernourishment, wasting, stunting, child mortality, number of people undernourished, total population, security, availability of food, access to food, stability (production, food prices) and utilization practices surrounding food.

The GHI data set, presented an aggregated score of levels of FS based on the following scale.

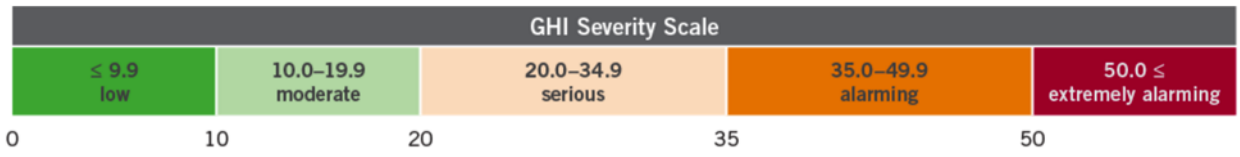


Fig 2: Severity scale of Global Health Index

Within the GHI data, each record has a response variable, RAWSCORE, that indicates the score of that country, values ranging from 0 to 50. If a country has low score from 0-10, that country was assigned a "low"" global hunger score. A score of above 40 warranted an "alarming" ranking. The analysis in the current study attempts to identify which macro-level attributes from the FAO data set, contributed more to the probability that a country will fall into the alarming category. A generalized linear model will be applied along with the maximum quasi likelihood model for this report.

It should be noted that the combined data set of FAO and GHI information was small considering the difficulty in collecting this data. The data consisted of 131 observations with 30 predictor variables and one response variable. The best subset of predictors was selected and data was analyzed for multi-collinearity and goodness of fit among other tests. A description of the data set is given in Table A1 of the Appendix.

Some initial work was performed to extract the selected predictors from the FAO data set and to combine these data with the GHI data set. The consolidated data set was used to predict the GHI scores. Data from 2008 would be used in a training context. Our intent was to apply the selected regression model against 2009 data as part of our efforts to assess predictive capability.

Data was loaded in a manner in which character data were treated as strings. The data sets were cleaned and converted to numeric format by replacing characters like "<5.0" to 5.0. Spaces were replaced by NA values.

Overall values of all the missing values in the test data are summarized in the following table.

```
sapply(hunger,function (x) sum(is.na(x)))
                Index                Region              RawScore                  GHI08           energysupply
                    2                     0                    11                     11                     14
      foodproduction               tubers               protein         proteinanimal      PavedvsTotalRoads
                   11                    14                    14                     14                     82
         RoadDensity          RailDensity       purchasingpower         foodpricelevel      undernourishment
                   68                    61                     8                     31                      0
         fooddeficit         foodinadequacy         cerealimport          arableland          foodimports
                   29                     0                     7                      4                      3
   politicalstability  foodpricevolatility foodproductionvolatility  foodsupplyvariability      watersources
                    3                    32                     4                     18                      5
          sanitation             wasting1              stunted1             underweight1          anemialpreg
                    4                   107                   107                    107                      4
      anemiachildren        undernourished               wasting              stunting              underfive
                    4                     0                     0                      0                      0
               score
                    0
```

Table 1: Extent of missing data

These missing values were replaced by the mean value of respective columns. Three columns, namely, wasting, stunted and underweight, had NA values for 75% of its data or higher. Consequently these potential predictors were removed.

Descriptive statistics for the data were generated (see Table 2). Columns such as Purchasing-power and food-production appeared to have outliers as their maximum value were substantially larger than their median values.

	means	medians	sds	maxim	minim	missing
RawScore	14.2292	11.7	9.568444	42.7	4.9	0
energysupply	109.531	115	29.03971	153	0	0
foodproduction	241.8053	216	156.7826	1081	0	0
tubers	50.88496	51	16.94832	81	0	0
protein	67.9115	65	22.72018	125	0	0
proteinanimal	24.9646	25	15.19747	76	0	0
PavedvsTotalRoads	17.51062	0	29.33597	100	0	0
RoadDensity	15.7115	0	32.07154	201.3	0	0

	means	medians	sds	maxim	minim	missing
RailDensity	0.7132743	0.2	1.177505	4.8	0	0
purchasingpower	8398.356	6733.7	7731.01	43567.3	577.6	0
foodpricelevel	4.505929	4.68	2.933368	10.53	0	0
undernourishment	15.42991	12.3	11.53396	53.1	5	0
fooddeficit	100.0708	71	102.5073	524	0	0
foodinadequacy	24.52823	24.53	12.30414	59.9	5.8	0
cerealimport	25.84159	26.3	47.22871	100	-163.8	0
arableland	24.78053	10.9	29.35328	100	0	0
foodimports	40.17699	17	105.0215	861	0	0
politicalstability	-0.4943363	-0.36	0.8201055	1.19	-2.57	0
foodpricevolatility	10.67611	10.2	9.976376	78.4	0	0
foodproductionvolatility	10.61593	7.2	10.57285	70.9	0	0
foodsupplyvariability	35.0708	29	25.56774	154	0	0
watersources	79.30354	86.6	20.33665	99.6	0	0
sanitation	58.97699	62.7	30.75312	100	0	0
anemiachildren	44.79912	40.2	19.26334	87.2	0	0
undernourished	16.54071	16.54	12.28178	49.4	0	0
wasting	7.405929	7.41	4.905743	26	0	0
stunting	26.10885	26.11	12.20621	49.6	0	0
underfive	6.253097	6	4.89775	20.2	0	0
score	2.477876	3	1.018554	4	1	0

Table 2: Key descriptive statistics

The Raw Score was treated as the response variable. This was binned to be categorized under 5 groups: Low, Moderate, Serious, Alarming, and Extremely Alarming.

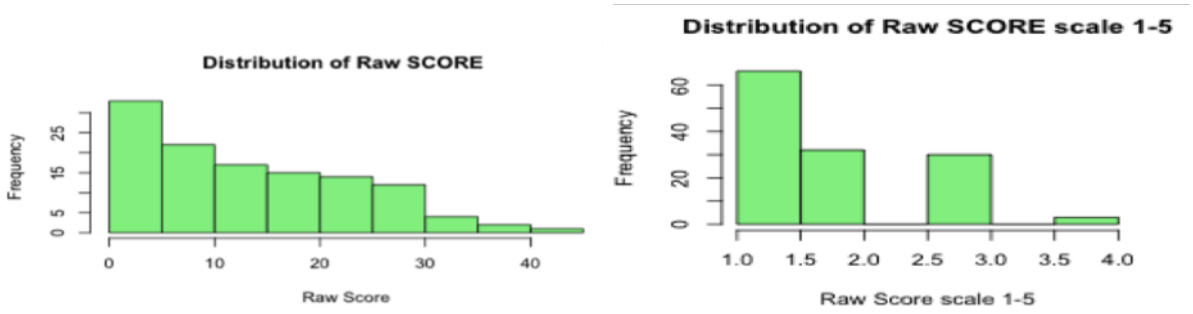


Fig 3a: Raw scores categorized and plotted. Fig 3b: Serious and Alarming = 1 else 0

It can be seen that most of the scores fell into the first category, which is in the lower scale of Global Health Index (see Figure 2).

The distribution for the response variable (RawScore) in shown in Figure 3a. The data was positively skewed indicating that there were fewer countries with serious and alarming global health index.

A dummy Boolean variable called RawScoreBool was created to indicate whether a country is in an alarming state for GHI. There were 33 out of 131 countries in Serious and the Alarming Severity scale (RawScoreBool = 1) and the data will be analyzed in detail to understand the factors affecting their high GHI scores.

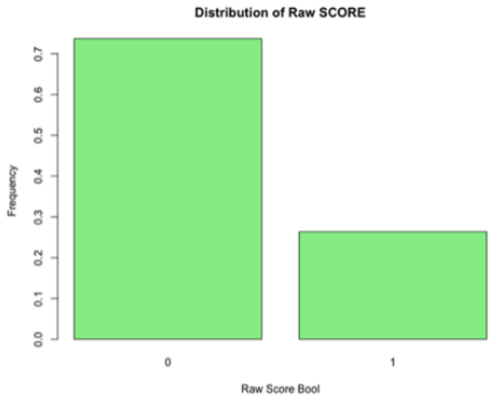


Fig 4: Frequency of countries in alarming FS state or otherwise

```
> correlations
                          RawScore
RawScore                      1.00
energysupply                 -0.65
foodproduction               -0.68
tubers                        0.51
protein                      -0.74
proteinanimal                -0.80
PavedvsTotalRoads            -0.25
RoadDensity                  -0.35
RailDensity                  -0.36
purchasingpower              -0.85
foodpricelevel                0.35
undernourishment              0.77
fooddeficit                   0.59
foodinadequacy                0.58
cerealimport                 -0.02
arableland                   -0.29
foodimports                   0.40
politicalstability           -0.36
foodpricevolatility           0.00
foodproductionvolatility     -0.45
foodsupplyvariability        -0.06
watersources                 -0.70
sanitation                   -0.71
anemiachildren                0.74
undernourished                0.50
wasting                       0.59
stunting                      0.65
underfive                     0.72
score                         0.58
```

Table 3: Correlation of potential predictors to response

A correlation table was created to view the correlation between the predictors and the target variables. The predictor names are shortened for display purposes. The results are shown in Table 3.

Note that predictors were selected from these candidate variables if their absolute correlation value was greater than or equal to 0.5. The selected predictors were therefore streamlined to: energysupply, foodproduction, tubers, protein, proteinanimal, purchasingpower, undernourishment, fooddeficit, watersources, sanitation , anemiachildren, undernourished, wasting, stunting and underfive.

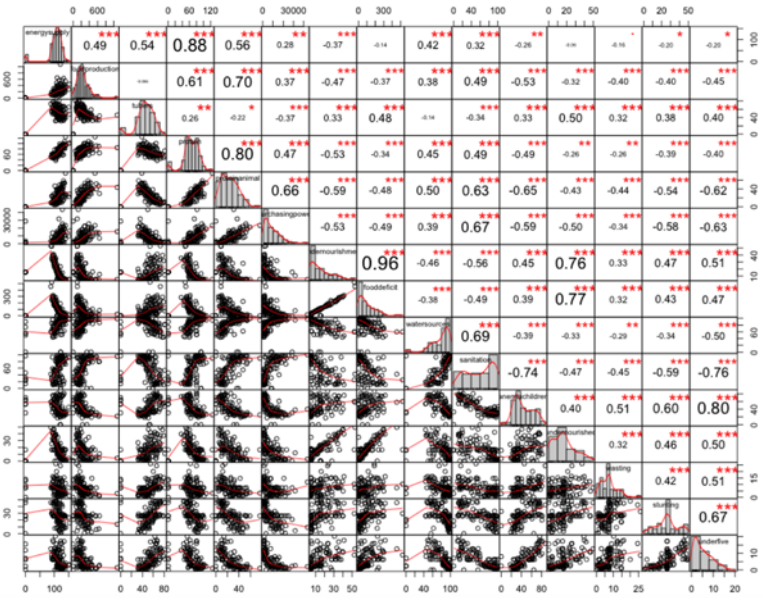


Fig 5: Correlations between predictors

To understand the possibility of co-linear relationships between predictors, an analysis was performed on the relationships between the different variables. These are summarized in Fig 5.

High correlation was noted among a number of variables including between undernourishment and fooddeficit (0.90), between undernourishment and undernourished (0.76), between fooddeficit and undernourished (0.81), between sanitation and watersources (0.69), and between protein and proteinanimal (0.80).

Model Building and Selection

A number of models were built based on the discrete score of the response variable and the observed distribution of raw scores. The Poisson family of GLMs were used in the creation of the models. None of the predictors showed normal distribution (see Fig 13 in the Appendix), so the Generalized Linear Model was a good option. In the glm R function, the quasi Poisson option was used for the link.

Results

Model 1

The first model developed applied a Poisson GLM using all possible predictors. Results are shown below.

```
glm(formula = RawScore ~ ., family = poisson, data = hungerpear)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.55757  -0.20055  -0.00532   0.18220   0.53588

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)        3.439e-01  9.724e-01   0.354    0.724
energysupply      -4.213e-03  9.657e-03  -0.436    0.663
foodproduction    -1.581e-04  8.360e-04  -0.189    0.850
tubers             3.952e-03  9.780e-03   0.404    0.686
protein           -2.038e-04  1.149e-02  -0.018    0.986
proteinanimal     -1.149e-03  1.435e-02  -0.080    0.936
purchasingpower   -2.037e-05  1.833e-05  -1.111    0.266
undernourishment   2.079e-02  4.760e-02   0.437    0.662
fooddeficit       -1.168e-03  4.467e-03  -0.261    0.794
watersources       2.007e-03  5.696e-03   0.352    0.725
sanitation         8.548e-05  5.121e-03   0.017    0.987
anemiachildren     3.791e-03  7.725e-03   0.491    0.624
undernourished    -3.787e-03  1.067e-02  -0.355    0.723
wasting            1.430e-02  1.692e-02   0.845    0.398
stunting           2.569e-03  8.309e-03   0.309    0.757
underfive          6.573e-03  2.731e-02   0.241    0.810

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 47.325  on 112  degrees of freedom
Residual deviance:  7.186  on  97  degrees of freedom
AIC: 316.16

Number of Fisher Scoring iterations: 4
```

Fig 6: Results from Poisson Model 1

Variables	VIF
RawScore	6.190303
energysupply	16.780785
foodproduction	2.295931
tubers	6.137209
protein	14.696334
proteinanimal	8.616424
purchasingpower	3.094845
undernourishment	58.861366
fooddeficit	44.456481
watersources	2.696109
sanitation	4.970387
anemiachildren	4.273243
undernourished	3.370434
wasting	1.862603
stunting	2.272121
underfive	4.450594

Fig 7: VIF values

None of the predictors in this model were found to be significant. Concern over multicolinearlity resulted in a variance inflation factor analysis. The VIF are shown below.

Clearly a number of variables breach the threshold of 10. These large values were removed to create Model 2.

Model 2

Model two eliminated a number of variables, given the VIF concerns mentioned above. Moreover, a quasipoisson option for the family parameter was used to create the model. If the variance of the error distribution in the data is greater than that expected under the Poisson distribution, then the quasipoisson option should be used. Results from this model are shown below.

```
Call:
glm(formula = RawScore ~ ., family = quasipoisson(link = log),
    data = hungervif)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.60065  -0.24150  -0.00362   0.20847   0.74216

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     8.519e-01  1.754e-01   4.856 4.38e-06 ***
foodproduction -4.128e-04  2.570e-04  -1.606 0.111427
tubers         -3.603e-03  1.372e-03  -2.625 0.010016 *
proteinanimal  -7.987e-03  3.051e-03  -2.618 0.010202 *
purchasingpower -2.201e-05  5.529e-06  -3.981 0.000129 ***
watersources   -1.360e-04  1.607e-03  -0.085 0.932705
sanitation     -7.371e-04  1.516e-03  -0.486 0.627862
anemiachildren  2.445e-03  2.245e-03   1.089 0.278621
undernourished  4.186e-03  2.138e-03   1.958 0.053022 .
wasting         1.355e-02  4.822e-03   2.810 0.005957 **
stunting        3.807e-03  2.475e-03   1.538 0.127056
underfive       2.482e-03  8.060e-03   0.308 0.758754
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for quasipoisson family taken to be 0.09270752)

    Null deviance: 47.3250  on 112  degrees of freedom
Residual deviance:  9.2705  on 101  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 4
```

Fig 8: Results from Poisson Model 2

Predictors: watersources, anemiachildren,sanitation, stunting, foodproduction and underfive, were observed to have p-values above 0.5. Consequently, one of each of the highly correlated values among predictors can be removed to make the remaining predictors significant.

Model 3

The correlation matrix in Fig 5 showed that watersources and sanitation are highly correlated, as such watersources would be removed. The same logic was used with underfive and foodproduction. The resulting model shows predictors that are significant. The results from this model are summarized below.

```
Call:
glm(formula = RawScore ~ ., family = quasipoisson(link = log),
    data = hungervif)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.60065  -0.24150  -0.00362   0.20847   0.74216

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)     8.519e-01  1.754e-01   4.856 4.38e-06 ***
foodproduction -4.128e-04  2.570e-04  -1.606 0.111427
tubers         -3.603e-03  1.372e-03  -2.625 0.010016 *
proteinanimal  -7.987e-03  3.051e-03  -2.618 0.010202 *
purchasingpower -2.201e-05 5.529e-06  -3.981 0.000129 ***
watersources   -1.360e-04  1.607e-03  -0.085 0.932705
sanitation     -7.371e-04  1.516e-03  -0.486 0.627862
anemiachildren  2.445e-03  2.245e-03   1.089 0.278621
undernourished  4.186e-03  2.138e-03   1.958 0.053022 .
wasting         1.355e-02  4.822e-03   2.810 0.005957 **
stunting        3.807e-03  2.475e-03   1.538 0.127056
underfive       2.482e-03  8.060e-03   0.308 0.758754
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 0.09270752)

    Null deviance: 47.3250  on 112  degrees of freedom
Residual deviance:  9.2705  on 101  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 4
```

Fig 8: Results from Poisson Model 2

```
glm(formula = RawScore ~ tubers + proteinanimal + undernourished +
    wasting + stunting, family = quasipoisson(link = log), data = hungervif)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.1839  -0.2380  -0.0419   0.1615   1.0732

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    0.678319   0.111237   6.098 1.74e-08 ***
tubers        -0.002963   0.001500  -1.975 0.050804 .
proteinanimal -0.018228   0.002251  -8.098 9.61e-13 ***
undernourished 0.006432   0.002302   2.795 0.006160 **
wasting        0.017984   0.005171   3.478 0.000731 ***
stunting       0.008485   0.002515   3.374 0.001033 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 0.1196839)

    Null deviance: 47.325  on 112  degrees of freedom
Residual deviance: 12.446  on 107  degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 4
```

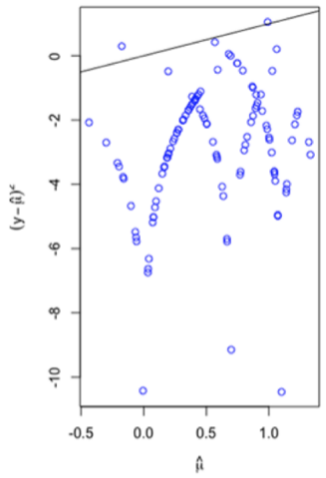
Fig 9: Results from Poisson Model 3

The sandwich package in R was used to calculate the robust standard errors and other values like p-value and the confidence Intervals

The residual deviance of Model 3 was used to perform a goodness of fit for the overall model. This measurement is the difference between the deviance of this model and the maximum deviance of the ideal model where predicted as well as the calculated values are identical. Getting a low residual difference indicates that the goodness of fit will not be significant. The goodness of fit chi-squared test was found to be statistically significant indicating that the model did not fit well. The reason could be the possibility of omitted variables, or the linearity assumption may not be good or potentially due to over-dispersion.

Since the mean is equal to the variance in Poisson distribution, $(y - \widehat{\mu})^2$ is plotted against the mean as its difficult to estimate the variance for a given mean. The line in Figure 11 represents variance = mean. The variance is not proportional to mean, so there is over-dispersion.

It should be noted that AIC stepwise method was also applied and Model 3 was still a better model than the AIC model. The summary of the final model showed that the exponentiation of coefficients yields multiplicative terms that can be used to calculate the estimated RawScore for a one unit increase in the predictor, holding all other variables constant.



The chosen model is represented by the formula below:

$$Y = 0.678 + \beta_1(-0.02) + \beta_2(-0.018) + \beta_3(0.0064) + \beta_4(0.017) + \beta_5(0.0084)$$

where: $\beta_1 = tubers$, $\beta_2 = proteinanimal$, $\beta_3 = undernourished$, $\beta_4 = wasting$, $\beta_5 = stunting$

Fig 11: Assessing over-dispersion

That is:

- Tubers: a 1 unit increase in tubers yields a 0.2% (0.002) decrease in the expected raw score for hunger.
- Proteinanimal: a 1 unit increase in proteinanimal yields a 1.8% (0.018) decrease in the expected raw score for hunger.
- undernourished: a 1 unit increase in undernourished yields a .6% (0.0064) increase in the expected raw score for hunger.
- Wasting: a 1 unit increase in wasting yields a 1% (0.017) increase in the expected raw score for hunger.
- Stunting: a 1 unit increase in stunting yields a 0.8% (0.0084) increase in the expected raw score for hunger.

Note that the direction of these coefficients makes logical sense.

Model using Box-Cox transformation

Many statistical tests are based on the assumption of normality in data. In reality, most data sets are not approximately normal like most of the predictors in the hunger data set. The log likelihood function for selecting the lambda power transform is dependent on the R^2 of the model, so without a model, transformation cannot be applied. Applying the box-cox transformation yielded a value of -0.1010101 for lambda. Since the value of lambda was within the range of [-2 to 2], the final chosen model was considered to be good. The Box-Cox transformation showed that no further improvement or transformation can be made to the model to improve its fit (see the 2nd and the 3rd plots in the figure below).

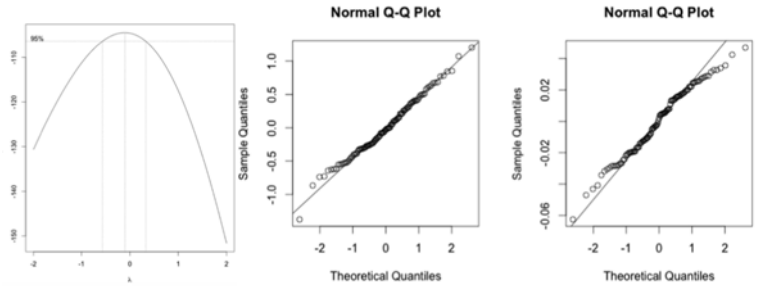


Fig 12: Box-cox transformations

Model using leverage points:

Leverage is defined as a measure of how much each data point influences the regression.

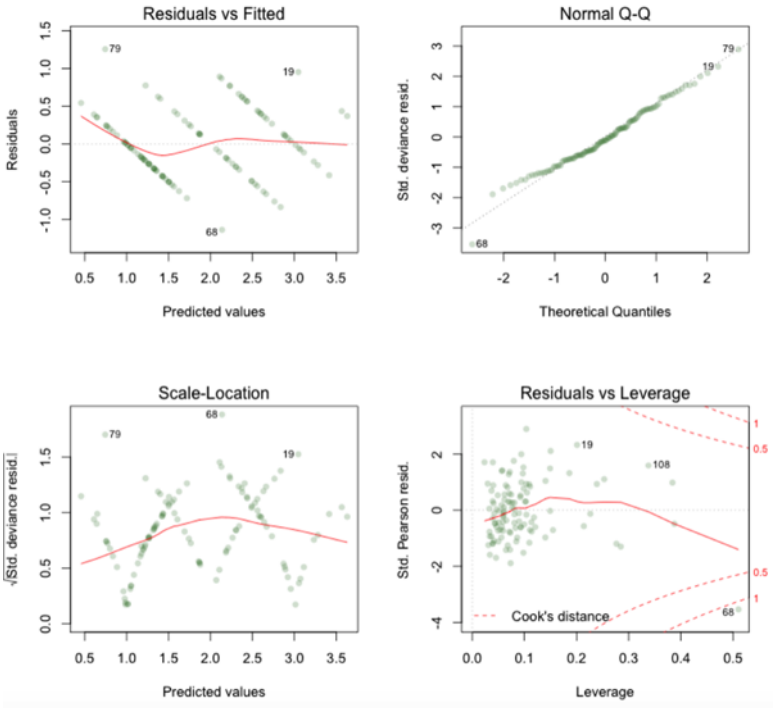


Fig 13: Cooks distance

The pattern of constant variation in the first plot in Fig 13 indicates that the choice of variance function was reasonable. Cook's distance measures the changes of regression if a data point was removed. This distance increases by leverage and also by large residuals. The red line in this plot should stay close to the grey dotted line at 0 and the points should stay within 0.5 cook's distance. The residuals versus leverage graph as shown in the plot shows labelled points indicating an undue influence on the regression by these points. For example, the data points 19,68 and 108 were investigated and a low value of underfive existed in the first two observations. However there was nothing unusual about the third observation.

Appication to the test set

A test data set comprised of values from 2009 was also prepared as per the training data set. The final selected model was applied to this test set.

RawScore	tubers	proteinanimal	undernourished	wasting	stunting	pred1.fit
1	77	14	26.80	5.61	40.90	2
1	41	60	11.41	7.20	17.80	1
1	57	24	2.90	4.10	11.70	1
3	60	10	14.20	5.61	20.94	2
1	36	45	0.20	1.60	7.70	1
1	43	33	11.41	4.20	20.80	1
1	63	39	1.70	3.10	18.00	1
1	0	26	11.41	5.61	20.94	2
3	80	12	16.40	14.30	36.10	2
1	38	64	0.80	2.20	3.70	1

When tested on the evaluation data set for the alarming or otherwise classification, the misclassification error rate was found to be 0.03100775.

The probabilities for the evaluation set is calculated to predict whether a country is going to be in an alarming range of GHI score index. The area under the ROC curve gave 0.7433862, which indicates an accurate test.

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 125   1
         1   3   0

               Accuracy : 0.969
                 95% CI : (0.9225, 0.9915)
    No Information Rate : 0.9922
    P-Value [Acc > NIR] : 0.9965

                  Kappa : -0.0118
 Mcnemar's Test P-Value : 0.6171

            Sensitivity : 0.000000
            Specificity : 0.976562
         Pos Pred Value : 0.000000
         Neg Pred Value : 0.992063
             Prevalence : 0.007752
         Detection Rate : 0.000000
   Detection Prevalence : 0.023256
      Balanced Accuracy : 0.488281

       'Positive' Class : 1
```

Fig 15: Assessment of classification performance

Negative binomial distribution[Fig.10a] was also used for over dispersed count data, that is, when the conditional variance exceeds the conditional mean. Since it has the same mean structure, it is considered a generalization of the Poisson regression, but with an extra parameter to model the over-dispersion. The model is as shown below, but more work is needed to be done.

```
Call:
glm(formula = RawScore ~ ., family = negative.binomial(1), data = datan1)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-0.67039  -0.14878  -0.02304   0.10391   0.72055

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)    0.634968   0.122111   5.200 9.65e-07 ***
tubers        -0.003073   0.001715  -1.792 0.075956 .
proteinanimal -0.017097   0.002223  -7.690 7.60e-12 ***
undernourished  0.006525   0.002546   2.563 0.011763 *
wasting         0.020067   0.005846   3.433 0.000851 ***
stunting        0.008679   0.002688   3.229 0.001649 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for Negative Binomial(1) family taken to be 0.04537824)

    Null deviance: 16.7402  on 112  degrees of freedom
Residual deviance:  4.6093  on 107  degrees of freedom
AIC: 418.06

Number of Fisher Scoring iterations: 5
```

Fig 10a: Results from Binomial Model

```
Fixed effects: RawScore ~ .
                    Value Std.Error DF   t-value p-value
(Intercept)     1.7415370 0.3249347 66  5.359652  0.0000
tubers          0.0142546 0.0051252 39  2.781254  0.0083
foodproduction -0.0026638 0.0004689 66 -5.680861  0.0000
 Correlation:
               (Intr) tubers
tubers         -0.908
foodproduction -0.553  0.237

Standardized Within-Group Residuals:
       Min          Q1         Med          Q3         Max
-2.92321353 -0.63484133 -0.07816256  0.60264418  2.09844111

Number of Observations: 113
Number of Groups:
                    tubers foodproduction %in% tubers
                        41                        108
```

Fig 10b: Results from Mixed Effects model

The data was also fit using the mixed effects model. It showed foodproduction is about 3 times the tubers value. The model seemed to know what we were trying to solve. The coefficients also showed random intercepts of

tubers and foodproduction. Every tuber for every foodproduction value gave a different coefficient. More work is needed to interpret these results.

Multinomial logistic regression is used to model nominal outcome variables, where the linear combination of the predictor variables is used model the log odds of the outcomes. This regression requires a large sample size because the Multinomial regression uses a maximum likelihood estimation method. However, this method was chosen for the sake of completeness. The summary and the plot are as shown below. The prediction was pretty accurate, especially for the tubers.

```
multinom(formula = wastingCat2 ~ RawScore + tubers, data = hungerMult)

Coefficients:
  (Intercept)   RawScore2 RawScore3 RawScore4       tubers
0    6.659500 -1.40697889 -3.611001 -38.90326 -0.091348414
6    1.804570  0.07715706 -1.663789 -29.07787 -0.045757084
7    4.454501 -0.67101712 -1.465131 -34.60864 -0.064374920
8   -1.273805  0.71968442 -1.038537 -22.07420  0.003517453
9  -12.063743 11.42665763 11.387976  11.37071 -0.021539883

Std. Errors:
  (Intercept) RawScore2 RawScore3    RawScore4     tubers
0    1.743678 0.8896764 1.1412938 6.584807e-12 0.03340679
6    2.399244 1.2574475 1.5201596 6.948195e-11 0.04635988
7    1.697984 0.9142333 0.9280302 3.006472e-12 0.03204219
8    2.580131 1.4058412 1.5677453 3.589764e-09 0.04611251
9    1.935455 1.9598364 2.1307235 2.120363e+00 0.06053191

Residual Deviance: 268.392
AIC: 318.392
```

Fig 10c: Multinomial Regression Model

```
# weights:  36 (25 variable)
initial  value 202.468820
iter  10 value 142.805975
iter  20 value 134.442121
iter  30 value 134.224113
iter  40 value 134.197156
iter  50 value 134.195995
final  value 134.195985
converged
```

Fig 10d: Negative log likelihood

The model shows iteration history and the final negative log-likelihood value of 134.19. This value multiplied by 2 is then seen in the model summary as the Residual Deviance (268.382). These results can be compared to the other models. Results: A one-unit increase in the variable tubers is associated with the decrease in the log odds of being in Wasting bin value of 6 vs. 2 in the amount of .0913 and so on.

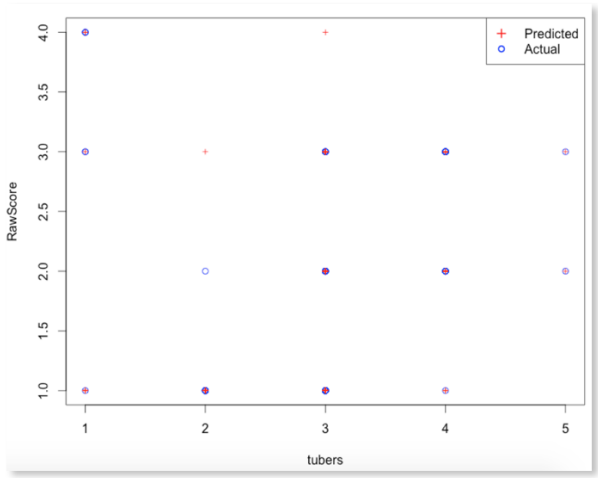


Fig 10 e: Predicted vs. Actual

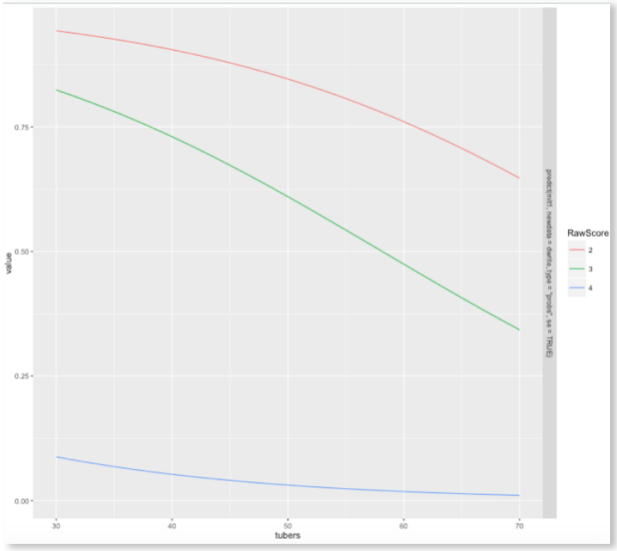


Fig 10 f: Categorical data for Tubers & RawScore

It can be seen in the right side plot that the predicted probabilities across the values for tubers for each level of the score is faceted by program type.As the hunger score of a country goes higher, the probability of tubers' availability dips down sharply. More research is needed especially about the economic condition of the countries, before this model can be compared to other models.

Summary and Future Work

Poisson regression was used to fit the GHI score to understand what variables effects the GHI score the most. Six models were created in Poisson using Pearson coefficients, Variance inflation factor, AIC, log and quasi Poisson.The quasi Poisson model performed better even though the Residual deviance was way below the degrees of freedom. This model used log as its link function. Other models like Binomial and Mixed effects' models were also explored. Mixed effects' model showed promise in the sense that the tubers were 1/3rd of the food production and the p-values were significant for tubers/foodproduction. More research needs to be done to interpret and compare the results from these two models with the chosen Poisson model.

References

FAO Statistics. Food security indicators. Last updated February 2016. 2016. URL: http://www.fao.org/economic/ess/ess-fs/ess-fadata/en/#.VygIsaN97UI.

Gubert, M., M. Benício, J. Da Silva, T. Da Costa Rosa, et al. "Use of a predictive model for food insecurity estimates in Brazil." In: Archivos latinoamericanos de nutrición 60.2 (2010), pp. 119-225.

International Food Policy Research Institute (IFPRI), Welthungerhilfe (WHH) & Concern Worldwide. Global Hunger Index. 2015. URL: http://dx.doi.org/10.7910/DVN/JL16EW.

Mbukwa, J. "A model for predicting food security status among households in developing countries." In: International Journal of Development and Sustainability 2.2 (2013), pp. 544-555.

United Nations Secretariat. Hunger Statistics. 2016. URL: http://www.wfp.org/hunger/stats.

Appendix

Index	Columns	Definitions
0	Countries	Countries considered for this dataset
1	Raw-Score	GHI Score of the countries
2	GHI_08	
3	Average dietary energy supply adequacy	Measures adequacy of the national food supply in terms of calories
4	Average value of food production	The total value of Annual Food Production (as estimated by FAO) expressed in International Dollars per caput.
5	Share of Dietary Energy Supply Derived from Cereals, Roots and Tubers	Energy supply provided by cereals, roots and tubers divided by total Dietary Energy Supply (DES) (in kcal/caput/day)
6	Average protein supply	National average protein supply (expressed in grams per caput per day)
7	Average supply of protein of animal origin	National average protein supply (expressed in grams per caput per day
8	Percent of paved roads over total roads	An index of the quality of roads
9	Road density (per 100 square km of land area)	Total length of roads in km per 100 sq. km of land area
10	Rail lines density (per 100 square km of land area)	Total length of rail lines routes in km per 100 sq. km of land area
11	Gross domestic product per capita (in purchasing power	The purchasing power
12	Domestic food price level index	It allows comparison of the relative price of food across countries and over time
13	Prevalence of undernourishment	Proportion of population estimated to be at risk of caloric ina
14	Depth of the food deficit	Average food consumption of the undernourished, multiplied by the number of undernourished, and divided by the total population
15	Prevalence of food inadequacy	Proportion of population at risk of not covering the food requirements associated with normal physical activity
16	Cereal import dependecy ratio	Proxy to measure the cereal self sufficiency of a country and the potential impact of shocks in the international trade market
17	Percent of arable land equipped for irrigation	Share of land irrigated over total land area
18	Value of food imports in total merchandise exports	% of imports of food over total exports of merchandise. Indicator of exposure of the country to changes in international trade conditions
19	Political stability and absence of violence/terrorism	Political stability and absence of violence measures perceptions of the likelihood that the government will be destabilized or overthrown by unconstitutional or violent means, including politically-motivated violence and terrorism
20	Domestic food price volatility index	Variability of the Domestic Food Price Index across countries
21	Per capita food production variability	Variability of the net food production value
22	Per capita food supply variability	Variability of the total food supply
23	Percentage of population with access to improved water sources	percentage of the population with access to an adequate amount of water from an improved source
24	Percentage of population with access to sanitation facilities	facilities that can effectively prevent human, animal, and insect contact with excreta
25	Percentage of children under 5 years of age affected by wasting	Proportion below 2 standard deviations from the median weight-for-height of the reference population
26	Percentage of children under 5 years of age who are stunted	Proportion below 2 standard deviations from the median height-for-age of the reference population
27	Percentage of children under 5 years of age who are underweight	Proportion below 2 standard deviations from the median weight-for-age of the reference population
28	Prevalence of anaemia among pregnant women	percdentage of anemic adults
29	Prevalence of anaemia among children under 5 years of	percdentage of anemic children
30	undernourished	percentage of undernourished who are children
31	wasting	children significantly below standard height to weight ratios
32	stunting	children significantly below standard height for their age
33	underfive	children under 5
34	score	GHI trend score ??

Table A1: Description of data

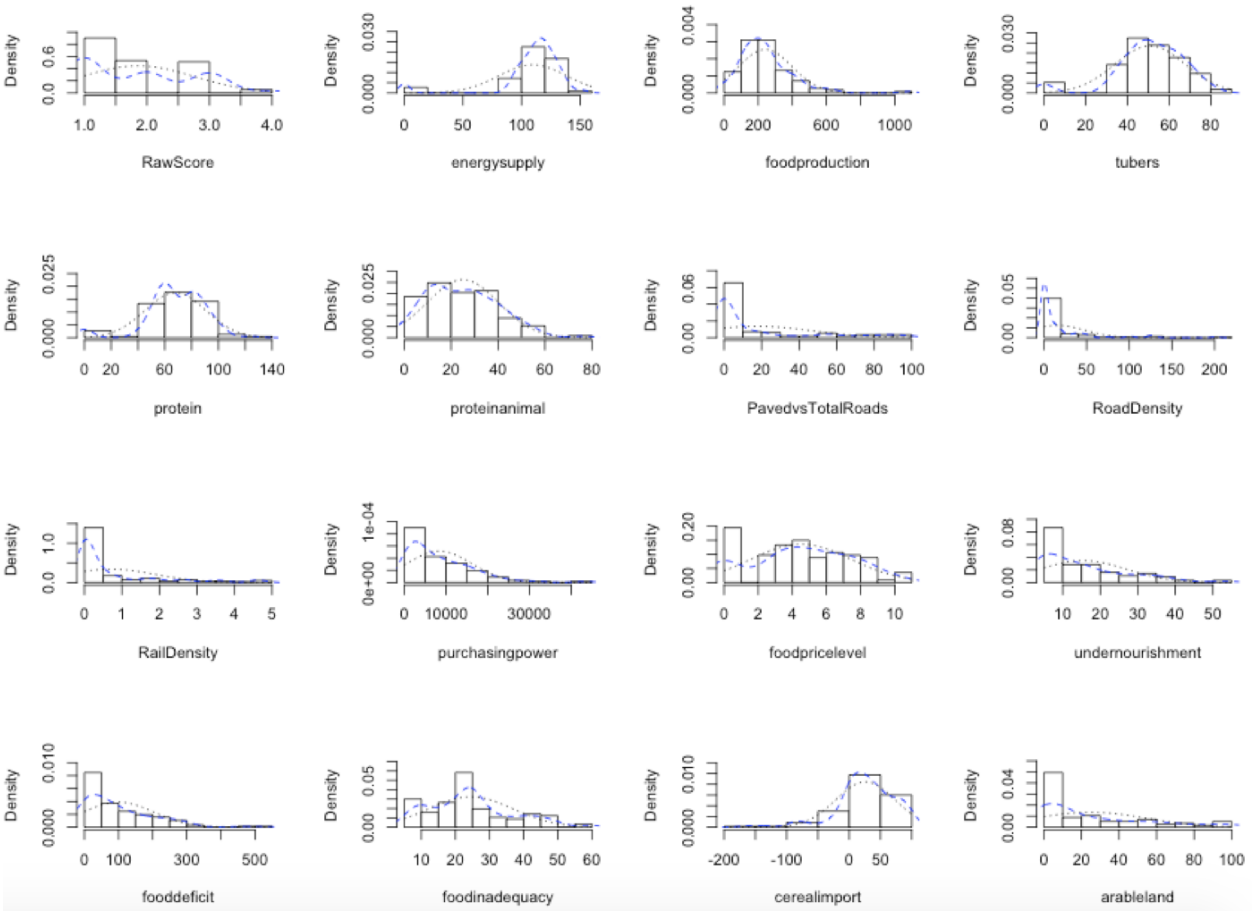


Fig 13a:Histograms of the first 15 predictors

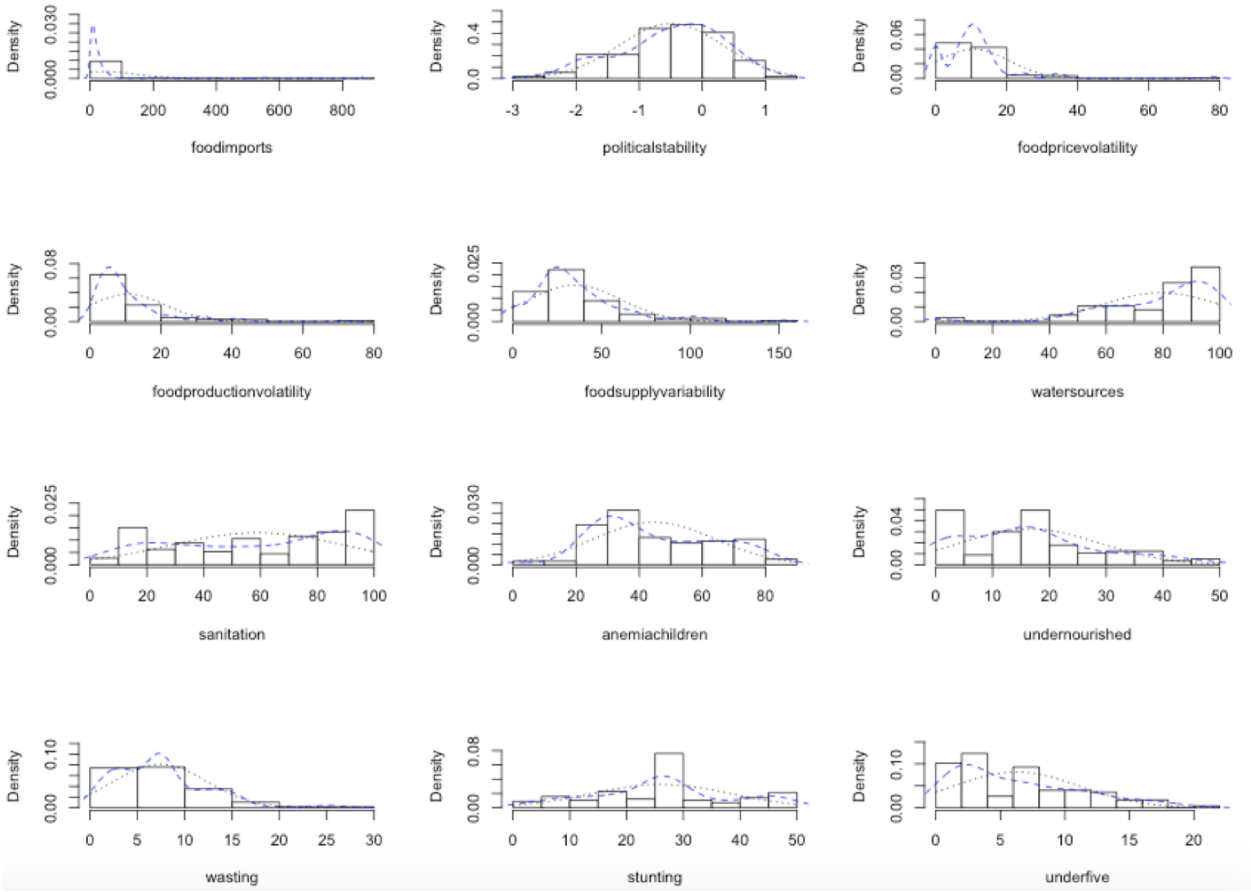


Fig 13b:Histograms of the last 11 predictors

Code

```
# Program to extract PDF data on Global Hunger Index 2008/9

# Converted PDF to txt file using linux utility pdftotext
# Minor bash manipulation to strip certain characters
# files: tr -d "* \t" < infile.txt > outfile.txt

# Reset R -----------------------------------------------------------------

rm(list = ls(all = TRUE))

# Libraries ---------------------------------------------------------------

library(countrycode)


# Useful Variables --------------------------------------------------------

countries <- countrycode_data$country.name
```

```r
# Functions ------------------------------------------------------------------

# Function to return rank of a given raw score into Global Hunger Index ranks:
# 0 = Low
# 1 = Moderate
# 2 = Serious
# 3 = Alarming
# 4 = Extremely Alarming
getGHI <- function(score) {

  if (is.na(score)) {
    return (NA)
  }

  if (score <= 4.9) {
    return ('Low')
  } else {
    if (score <= 9.9) {
      return ('Moderate')
    } else {
      if (score <= 19.9) {
        return ('Serious')
      } else {
        if (score < 29.9) {
          return ('Alarming')
        } else {
          return('Extremly_Alarming')
        }
      }
    }
  }
}

# Load Data ------------------------------------------------------------------

uri08 = '../data/GHI_2008.txt'
GHI_2008 <- read.delim(file = uri08, sep = '\n', stringsAsFactors = FALSE)

uri09 <- '../data/GHI_2009.txt'
GHI_2009 <- read.delim(file = uri09, sep = '\n', stringsAsFactors = FALSE)

# Extract 2008 Data on Global Hunger Index -----------------------------------

GHI_S08 <- rbind(data.frame(country = GHI_2008[5:69,]),
                 data.frame(country = GHI_2008[606:670,]))

l1 <- 535  + length(GHI_2008[5:69,]) - 1
l2 <- 1138 + length(GHI_2008[606:670,]) - 1

GHI_S08 <- cbind(GHI_S08, rbind(data.frame(raw_score_08 = GHI_2008[535:l1,]),
                                data.frame(raw_score_08 = GHI_2008[1138:l2,])))
```

```r
# Clean 2008 Data ------------------------------------------------------------

GHI_S08$country  <- as.character(GHI_S08$country)
GHI_S08$raw_score_08 <- as.character(GHI_S08$raw_score_08)

head(GHI_S08); tail(GHI_S08)

GHI_S08[(GHI_S08$raw_score_08 == '<5'),"raw_score_08"] <- "4.9"
GHI_S08[(GHI_S08$raw_score_08 == '-'),"raw_score_08"]  <- NA

GHI_S08$raw_score_08 <- as.numeric(GHI_S08$raw_score_08)

# Clean up spelling of country names
sum(!GHI_S08$country %in% countries)  # 33 to resolve

GHI_S08[!(GHI_S08$country %in% countries), ]

replace <- countries[c(30, 32, 39, 46, 54, 55, 57, 58, 68, 70, 71, 85, 114, 132,
                       133, 143, 159, 128, 186, 199, 211, 214, 216, 219, 223,
                       227, 234, 237, 243, 253, 260, 261, 268)]

GHI_S08[!(GHI_S08$country %in% countries), 'country'] <- replace

nrow( GHI_S08[!(GHI_S08$country %in% countries), ] )  ## all good on names


# Rank 2008 Global Hunger Index ----------------------------------------------

GHI_S08$score_08 <- sapply(GHI_S08$raw_score_08, function(x) getGHI(x))

head(GHI_S08)


# 2008 Details ---------------------------------------------------------------

# Number of Countries in 2008
nrow(GHI_S08)

# Summary
summary(GHI_S08)

# Countries with GHI scores
length(na.omit(GHI_S08$score_08))

# Countries rank as extremely alarming
subset(GHI_S08, subset = score_08 > 4)


# Extract 2009 Data on Global Hunger Index -----------------------------------

GHI_S09 <- rbind(data.frame(country = GHI_2009[2:51, ]),
                 data.frame(country = GHI_2009[479:528, ]),
                 data.frame(country = GHI_2009[952:980, ]))
```

```r
nrow(GHI_S09)

l1 <- 420 + length(GHI_2009[2:51, ]) - 1
l2 <- 893 + length(GHI_2009[479:528, ]) - 1
l3 <- 984 + length(GHI_2009[952:980, ]) - 1

GHI_S09 <- cbind(GHI_S09, rbind(data.frame(raw_score_09 = GHI_2009[420:l1,]),
                                data.frame(raw_score_09 = GHI_2009[893:l2,]),
                                data.frame(raw_score_09 = GHI_2009[984:l3, ])))


# Clean 2009 Data ----------------------------------------------------------

GHI_S09$country  <- as.character(GHI_S09$country)
GHI_S09$raw_score_09 <- as.character(GHI_S09$raw_score_09)

head(GHI_S09); tail(GHI_S09)

GHI_S09[(GHI_S09$raw_score_09 == '<5'),"raw_score_09"] <- "4.9"
GHI_S09[(GHI_S09$raw_score_09 == '-'),"raw_score_09"]  <- NA

GHI_S09$raw_score_09 <- as.numeric(GHI_S09$raw_score_09)

# Clean up spelling of country names
sum(!GHI_S09$country %in% countries)  # 32 to resolve

GHI_S09[!(GHI_S09$country %in% countries), ]

replace <- countries[c(30, 32, 39, 46, 54, 55, 57, 58, 68, 70, 71, 85, 114, 132,
                       133, 143, 159, 128, 186, 199, 211, 214, 216, 219, 223,
                       227, 234, 237, 243, 260, 261, 268)]

GHI_S09[!(GHI_S09$country %in% countries), 'country'] <- replace

nrow( GHI_S09[!(GHI_S09$country %in% countries), ] )  ## all good on names

# Missing countries between the two years:
GHI_S08[!GHI_S08$country %in% GHI_S09$country, ]  # UAE data missing in '09
                                                  # but values in '08 either

# Drop UAE from 2008 data frame
GHI_S08 <- subset(GHI_S08, subset = country != countries[253])


# Rank 2009 Global Hunger Index --------------------------------------------

GHI_S09$score_09 <- sapply(GHI_S09$raw_score_09, function(x) getGHI(x))

head(GHI_S09)

summary(GHI_S09)
```

```r
# Combine 2008 and 2009 GHI data ----------------------------------------

df <- cbind(GHI_S08, GHI_S09)
df <- df[, c(1,2,3,5,6)]
names(df) <- c('Countries', 'Raw_Score_08', 'GHI_08', 'Raw_Score_09', 'GHI_09')


# What do we have? ------------------------------------------------------

nrow(df)                          # 129 countries
length(na.omit( df$GHI_08 ))      # 120 countries with scores in 2008
length(na.omit( df$GHI_09 ))      # 121 countries with scores in 2009

summary(df)
# Write combined 2008 and 2009 data to disk -----------------------------

write.csv(x = df, file = '../data/GHI_Combined.csv')
```

```r
# Combining the three hunger data sets
#
# Food Security data required manipulation by hand, each variable was a seperate sheet in an
# Excel file so we needed to hand copy out the column and create a single file for the year.

# Load Libraries
library(dplyr)
library(countrycode)

# Load Data
ghi <- read.csv("D:/repos/FoodSecurity/data/GHI_Combined.csv")
fs2008 <- read.csv("D:/repos/FoodSecurity/data/Food_Security_Indicators_2008.csv")
trends2008 <- read.csv("D:/repos/FoodSecurity/data/trendsOfHunger2005.csv")
fs2009 <- read.csv("D:/repos/FoodSecurity/data/Food_Security_Indicators_2009.csv")
trends2009 <- read.csv("D:/repos/FoodSecurity/data/trendsOfHunger2005.csv")

# Fixing Column Names
names(ghi)
names(fs2008)
names(trends)
fs2008 <- rename(fs2008, Countries = Region)
trends2008 <- rename(trends2008, Countries = Country)
fs2009 <- rename(fs2009, Countries = Region)
trends2009 <- rename(trends2009, Countries = Country)

# Joining the 2008 data sets based on the Countries column
hunger2008 <- ghi %>%
  left_join(fs2008, by = "Countries") %>%
  left_join(trends2008, by = "Countries")

sum(!ghi$Countries %in% fs2008$Countries)
sum(!ghi$Countries %in% trends2008$Countries)
ghi[!(ghi$Countries %in% fs2008$Countries), 2]
ghi[!(ghi$Countries %in% trends2008$Countries), 2]
# We elected to fix this smaller subset of countries by hand.
```

```r
# Writing the data frame to .csv
write.csv(hunger2008,
          file = "D:/repos/FoodSecurity/data/FoodSecurity_2008_train.csv")


# Joining the 2009 data sets based on the Countries column
hunger2009 <- ghi %>%
  left_join(fs2009, by = "Countries") %>%
  left_join(trends2009, by = "Countries")

sum(!ghi$Countries %in% fs2009$Countries)
sum(!ghi$Countries %in% trends2009$Countries)
ghi[!(ghi$Countries %in% fs2009$Countries), 2]
ghi[!(ghi$Countries %in% trends2009$Countries), 2]
# Once again we fixed this subset of problems by hand

# Writing the data frame to .csv
write.csv(hunger2008,
          file = "D:/repos/FoodSecurity/data/FoodSecurity_2009_test.csv")
```

```r
## Correlation matrix with p-values. See http://goo.gl/nahmV for documentation of this function
cor.prob <- function (X, dfr = nrow(X) - 2) {
  R <- cor(X, use="pairwise.complete.obs")
  above <- row(R) < col(R)
  r2 <- R[above]^2
  Fstat <- r2 * dfr/(1 - r2)
  R[above] <- 1 - pf(Fstat, 1, dfr)
  R[row(R) == col(R)] <- NA
  R
}

## Use this to dump the cor.prob output to a 4 column matrix
## with row/column indices, correlation, and p-value.
## See StackOverflow question: http://goo.gl/fCUcQ
flattenSquareMatrix <- function(m) {
  if( (class(m) != "matrix") | (nrow(m) != ncol(m))) stop("Must be a square matrix.")
  if(!identical(rownames(m), colnames(m))) stop("Row and column names must be equal.")
  ut <- upper.tri(m)
  data.frame(i = rownames(m)[row(m)[ut]],
             j = rownames(m)[col(m)[ut]],
             cor=t(m)[ut],
             p=m[ut])
}

hunger <- read.csv("./FoodSecurity/data/FoodSecurity_2008_train.csv",header=TRUE, sep=",",stringsAsFacto
head(hunger)

colnames(hunger) <- c("Index","Region","RawScore","GHI08","energysupply","foodproduction","tubers","pro

head(hunger)
max(hunger$RawScore)

nrow(hunger)
```

```r
#131
# remove observations with high purchasing power like Bahrain
hunger <- hunger[ which(hunger$purchasingpower < 50000), ]
nrow(hunger)
#129
hunger <- hunger[ which(hunger$RawScore > 1), ]
nrow(hunger)

hungerbackup <- hunger

#take a look at the rows and columns
dim(hunger)
#113  31

mean(hunger$score)
var(hunger$score)

max(hungerbackup$RawScore)
mean(hungerbackup$RawScore)
var(hungerbackup$RawScore)


#Missing data
sapply(hunger,function (x) sum(is.na(x)))

# From the output table, there are more than 75% NA values for wasting1, stunted1 and underweight1, so
# RoadDensity

hunger<- hunger[,c(1:2,3,5:16,17:26,31:36)]
colnames(hunger)


# remove less than sign
hunger$undernourishment[hunger$undernourishment=="<5.0"] <-5.0

# remove NAs
hunger[is.na(hunger)]<-0

#Missing data
sapply(hunger,function (x) sum(is.na(x)))

head(hunger)
str(hunger)


hunger$score<- sub("^$", "NA", hunger$score)
hunger$score <- as.numeric(hunger$score)
unique(hunger$score)

hunger$undernourished<- sub("^$", "NA", hunger$undernourished)
hunger$undernourished <- as.numeric(hunger$undernourished)
unique(hunger$undernourished)
```

```r
hunger$wasting<- sub("^$", "NA", hunger$wasting)
hunger$wasting <- as.numeric(hunger$wasting)
unique(hunger$wasting)

hunger$stunting<- sub("^$", "NA", hunger$stunting)
hunger$stunting <- as.numeric(hunger$stunting)
unique(hunger$stunting)

hunger$underfive<- sub("^$", "NA", hunger$underfive)
hunger$underfive <- as.numeric(hunger$underfive)
unique(hunger$underfive)

hunger$foodinadequacy<- sub("^$", "NA", hunger$foodinadequacy)
hunger$foodinadequacy <- as.numeric(hunger$foodinadequacy)
unique(hunger$foodinadequacy)

hunger$undernourishment<- sub("^$", "NA", hunger$undernourishment)
hunger$undernourishment <- as.numeric(hunger$undernourishment)
unique(hunger$undernourishment)

hunger$foodinadequacy<- sub("^$", "NA", hunger$foodinadequacy)
hunger$foodinadequacy <- as.numeric(hunger$foodinadequacy)
unique(hunger$foodinadequacy)

# replace NA produced above with the mean value
hunger$score[is.na(hunger$score)]<-round(mean(hunger$score, na.rm=TRUE),digits=2)
hunger$undernourished[is.na(hunger$undernourished)]<-round(mean(hunger$undernourished, na.rm=TRUE),digi
hunger$wasting[is.na(hunger$wasting)]<-round(mean(hunger$wasting, na.rm=TRUE),digits=2)
hunger$stunting[is.na(hunger$stunting)]<-round(mean(hunger$stunting, na.rm=TRUE),digits=2)
hunger$underfive[is.na(hunger$underfive)]<-round(mean(hunger$underfive, na.rm=TRUE),digits=2)
hunger$undernourishment[is.na(hunger$undernourishment)]<-round(mean(hunger$undernourishment, na.rm=TRUE
hunger$foodinadequacy[is.na(hunger$foodinadequacy)]<-round(mean(hunger$foodinadequacy, na.rm=TRUE),digi


str(hunger)
t <- as.data.frame( t(sapply(hunger, function(cl) list(means=mean(cl), medians = median(cl), sds=sd(cl)
                                                       missing=sum(is.na(cl)))) ))

kable( t[-c(1,2), ])

## bin the scorea to 5 values
max(hunger$score)
hunger$score[hunger$score < 10] <- 1
hunger$score[hunger$score >= 50] <- 5
hunger$score[hunger$score >= 35] <- 4
hunger$score[hunger$score >= 20] <- 3
hunger$score[hunger$score >= 10] <- 2
unique(hunger$score)

mean(hunger$RawScore)
var(hunger$RawScore)

max(hunger$RawScore)
```

```r
hunger$RawScore[hunger$RawScore < 10] <- 1
hunger$RawScore[hunger$RawScore >= 50] <- 5
hunger$RawScore[hunger$RawScore >= 35] <- 4
hunger$RawScore[hunger$RawScore >= 20] <- 3
hunger$RawScore[hunger$RawScore >= 10] <- 2
unique(hunger$RawScore)

#hungerbackup

max(hunger$RawScore)

hunger$RawScoreBool[hunger$RawScore == 3] <- 1
hunger$RawScoreBool[hunger$RawScore < 3] <- 0

unique(hunger$RawScoreBool)

head(hunger)

# normality test
hist(hunger$RawScore,col="lightgreen", main="Distribution of Raw SCORE scale 1-5",xlab="Raw Score scale

# plot counts of 0 and 1 for RawScoreBool
barplot(prop.table(table(hunger$RawScoreBool)),col="lightgreen", main="Distribution of Raw SCORE",xlab=

table(hunger$RawScoreBool)
head(hunger)
colnames(hunger)
require(psych)
multi.hist(hunger[,3:16],dcol=c("blue","black"),main="")
multi.hist(hunger[,17:30],dcol=c("blue","black"),main="")


head(hunger[,3:30])
colnames(hunger)

#full model
m0 <- glm(RawScore ~., family=poisson,data=hunger[3:31])
summary(m0)


correlations <- round(cor(hunger[,3:31], use="pairwise", method="spearman"),2)
correlations

correlations[,1]
hungerpear <- hunger[,c("RawScore","energysupply", "foodproduction", "tubers","protein","proteinanimal"

# colorful plot the data
require(PerformanceAnalytics)
#chart.Correlation(hungerpear)


#pearson model
m1 <- glm(RawScore ~., family=poisson(link=log),data=hungerpear)
```

```r
summary(m1)

kable(summary(m1)$coefficients)

summary(m1)$coefficients[,1]


# All pvalues very high making all the predictors insignificant.
#check for multicollinearity
library(usdm)
vif(hungerpear)

# so remove values above 10. ( removing protein, undernourishment,fooddeficit)
hungervif <- hungerpear[,c("RawScore","foodproduction","tubers","proteinanimal","purchasingpower","wate

#vif model(removing multicollinearity)
m2 <- glm(RawScore ~., family=poisson(link=log),data=hungervif)
summary(m2)

# TRY quazi instead
# quasi - sanitation watersources foodinadequacy and proteinanimal are < 0.5
#If the variance of the error distribution in the data is greater than that
#expected under the poisson distributions ("overdispersion")

m3 <- glm(RawScore ~., family=quasipoisson(link=log),data=hungervif)
summary(m3)


m4 <- update(m3, . ~ .  -purchasingpower)
summary(m4)

m5 <- update(m4, . ~ .  -watersources-underfive-foodproduction)
summary(m5)


m6 <- update(m5, . ~ .  -anemiachildren-sanitation)
summary(m6)

m6$coefficients
## test model differences with chi square test
#anova(m3, m4, test="Chisq")

#anova(m4,test="Chisq")
#drop1(m4,test="Chisq")


require(sandwich)
require(msm)

cov.m6 <- vcovHC(m6, type="HC0")
std.err <- sqrt(diag(cov.m6))
r.est <- cbind(Estimate= coef(m6), "Robust SE" = std.err,
"Pr(>|z|)" = 2 * pnorm(abs(coef(m6)/std.err), lower.tail=FALSE),
```

```
LL = coef(m6) - 1.96 * std.err,
UL = coef(m6) + 1.96 * std.err)

r.est

with(m6, cbind(res.deviance = deviance, df = df.residual,
  p = pchisq(deviance, df.residual, lower.tail=FALSE)))

# STEP AIC and BIC
#All subset selection using stepwise procedure and AIC value
step(m2, direction="both", k=2)
lm.best.aic=m2
summary(lm.best.aic)
summary(m2)
anova(m4,lm.best.aic)
drop1(m2, test="Chisq")

anova(lm.best.aic,m6)

#######
s <- deltamethod(list(~ exp(x1), ~ exp(x2), ~ exp(x3), ~ exp(x4),
                      ~ exp(x5),~ exp(x6),~ exp(x7)), coef(m6), cov.m6)

## exponentiate old estimates dropping the p values
rexp.est <- exp(r.est[, -3])
## replace SEs with estimates for exponentiated coefficients
rse <- rexp.est[, "Robust SE"]
rse2 <- data.frame(as.list(rse))
str(rse2)
# delete the starsbool
rse2 <- rse2[,-11]
rse2

rse2 <- s

rexp.est

colnames(p)
#plot variance vs. mean
y=5
par(mfrow=c(1,2))
plot(log(fitted(m6)),log((hungervif$RawScore - fitted(m6))^2),
                         xlab=expression(hat(mu)),
                         ylab=expression((y-hat(mu))^2),col="blue")
abline(0,1)

#conf intervals
(est <- cbind(Estimate = coef(m5), confint(m5)))
exp(est)

require(aod)
m5$coefficients
wald.test(b=coef(m5), Sigma=vcov(m5), Terms=4:8)
```

```
require(MASS)
bc <-boxcox(m5)
transL <- bc$x[which.max(bc$y)]
transL
#-0.1010101
#re-run with transformation
mnew <- glm(RawScore^transL ~ ., data=hungervif)

# QQ-plot
op <- par(pty = "s", mfrow = c(1, 2))
qqnorm(m5$residuals); qqline(m5$residuals)
qqnorm(mnew$residuals); qqline(mnew$residuals)
par(op)

#### Cook's distance
y=5
hvalues <- influence(m4)$hat
stanresDeviance <- residuals(m4)/sqrt(1-hvalues)
plot(hvalues,stanresDeviance,ylab="Standardized Deviance Residuals",xlab="Leverage Values",ylim=c(-3,3)
abline(v=2*7/length(y),lty=2)
identify(hvalues,stanresDeviance,cex=0.75)

par(mfrow=c(2,2))
#cook's distance
plot(m4,col=rgb(0,100,0,50,maxColorValue=255), pch=16)

# leverage points - find rows
hungervif[19,]
hungervif[68,]
hungervif[108,]

hungertest <- read.csv("./FoodSecurity/data/FoodSecurity_2009_test.csv",header=TRUE, sep=",",stringsAsFa
head(hungertest)

colnames(hungertest) <- c("Index","Region","RawScore","GHI08","energysupply","foodproduction","tubers",

head(hungertest)

hungertestSelect <- hungertest[,c("RawScore","tubers","proteinanimal","undernourished","wasting","stunti


str(hungertestSelect)

# remove NAs
hungertestSelect[is.na(hungertestSelect)]<-0

head(hungertestSelect)
hungertestSelect$undernourished <- as.numeric(hungertestSelect$undernourished)
hungertestSelect$wasting <- as.numeric(hungertestSelect$wasting)
hungertestSelect$stunting <- as.numeric(hungertestSelect$stunting)

#Missing data
sapply(hungertestSelect,function (x) sum(is.na(x)))
```

```r
#head(hunger)
#unique(hunger$RawScore)

hungertestSelect$wasting[is.na(hungertestSelect$wasting)]<-round(mean(hungertestSelect$wasting, na.rm=TR
hungertestSelect$tubers[is.na(hungertestSelect$tubers)]<-round(mean(hungertestSelect$tubers, na.rm=TRUE)
hungertestSelect$proteinanimal[is.na(hungertestSelect$proteinanimal)]<-round(mean(hungertestSelect$prote
hungertestSelect$undernourished[is.na(hungertestSelect$undernourished)]<-round(mean(hungertestSelect$und
hungertestSelect$stunting[is.na(hungertestSelect$stunting)]<-round(mean(hungertestSelect$stunting, na.rm


unique(hungertestSelect$RawScore)

#!!!!!!
pred1 <- predict(m6, hungertestSelect, type="response", se.fit=TRUE)
df <- data.frame(pred1$fit)
head(df)
predVal <- round(df,0)
head(predVal)
hungertestSelect <- cbind(hungertestSelect,predVal)
kable( head(hungertestSelect, n=10) )
#hungertestSelect$pred1 <- pred1$fit
unique(hungertestSelect$pred1.fit)
nrow(hungertestSelect)
unique(hungertestSelect$RawScore)


unique(hungertestSelect$RawScoreMod)
str(hungertestSelect)
hungertestSelect$RawScore <- as.numeric(round(hungertestSelect$RawScore,0))
hungertestSelect$RawScore[hungertestSelect$RawScore < 10] <- 1
hungertestSelect$RawScore[hungertestSelect$RawScore >= 40] <- 5
hungertestSelect$RawScore[hungertestSelect$RawScore >= 35] <- 4
hungertestSelect$RawScore[hungertestSelect$RawScore >= 20] <- 3
hungertestSelect$RawScore[hungertestSelect$RawScore >= 10] <- 2
unique(hungertestSelect$RawScore)


hungertestSelect$pred1.fit[hungertestSelect$pred1.fit<= 3]<- 0
hungertestSelect$pred1.fit[hungertestSelect$pred1.fit>3]<- 1

# Rawscore 3 or more belong to alarming score
hungertestSelect$RawScore[hungertestSelect$RawScore<= 3]<- 0
hungertestSelect$RawScore[hungertestSelect$RawScore>3]<- 1


#confusion matrix - training set
table(hungertestSelect$RawScore,hungertestSelect$pred1.fit)
#      0   1
#  0 125   1
#  1   3   0

#calculate the misclassification error rate
len <- length(hungertestSelect$pred1.fit)
```

```
s<-sum(hungertestSelect$RawScore != hungertestSelect$pred1.fit)/len
s
# 0.031007775

confint(m6)
confMat <- data.frame(matrix(c(125,1,3,0),nrow=2,ncol=2))
confMat
colnames(confMat) <- c(0,1)
head(class.output)
hungertestSelect$myProbabilities <- predict(m6,hungertestSelect,type="response")


##########  Professor's code from hw3 solutions ###
### instead of the libraries so that I can keep a copy of his code for my records ##
# Write a function that returns the confusion matrix

head(hungertestSelect)
class.output <- hungertestSelect
accuracy(class.output, 1, 7) # accuracy 0.9689922
error.rate(class.output, 1, 7) # classification error rate  0.031007775
sum(accuracy(class.output, 1, 7),
 error.rate(class.output, 1, 7)) # accuracy + error = 1
precision(class.output,1, 7) # precision 0
sensitivity(class.output,1, 7) # sensitivity 0
specificity(class.output,1, 7) # specificity 0.9920635
f.one(class.output,1, 7)# F1 score

require(caret)
confusionMatrix(data = class.output$RawScore,
                reference = class.output$pred1.fit, positive = '1')
roc.curve(class.output$myProbabilities,
 class.output$RawScore) # ROC Curve and AUC  0.7433862

head(hungertestSelect)
x1<- rep(0,20)
x2 <- rep(1,20)
x<- c(x1,x2)
x

s<-seq(from=1,to=129,by=1)
data<-rbind(s,hungertestSelect$pred1.fit)
colnames(data)<-c(s,x2)
data

data <- t(data)
plot(data,axes=FALSE,
xlab="row number of observations in the Evaluation set)", ylab="logit outcome")
axis(side=2, at=c(0:1))
axis(side=1, at=seq(1,129,by=1))
data

colnames(hungervif)
colnames(hungertestSelect)
```

```
plot(hunger$wasting, hunger$RawScore,col="blue",xlab="Tubers", ylab="RawScore")
points(hungertestSelect$wasting,hungertestSelect$pred1.fit,col="red",pch=3,cex=.6)

require(MASS)

datan1 <- hungervif[,c("RawScore","tubers","proteinanimal","undernourished","wasting","stunting")]
n1 = glm(RawScore ~ ., family = negative.binomial(1), data=datan1)
summary(n1)

n2 <- update(n1, . ~ .  -purchasingpower-proteinanimal-sanitation-underfive)
p1 <- glm(RawScore ~.,  family = "poisson", data = datan1)
X2 <- 2 * (logLik(p1) - logLik(n2))
X2
pchisq(X2, df = 2, lower.tail=FALSE)
(est <- cbind(Estimate = coef(n2), confint(n2)))
exp(est)

## LME Fit the model using mixed effects model
library(nlme)
str(hungervif)
lm1<- lm(RawScore ~.,data=hungervif)
summary(lm1)
lmmodel <- lme(RawScore ~.,data=hungervif,random=~1|tubers/foodproduction)
# shows randon intercepts of tubers and foodproduction - every tubers and every foodproduction gives a
summary(lmmodel)
#foodproduction is about 3 times of tubers production
coef(lmmodel)
coef(lm1)
```

```
# Multinomial regression model
library(nnet)
mlt1<- multinom(RawScore ~.,data=hungervif)
summary(mlt1)

mlt2 <- update(mlt1, . ~ .  -foodproduction-purchasingpower-watersources-sanitation-anemiachildren-unde
summary(mlt2)

colnames(hungervif)
p1 <- predict(mlt2,hungervif,"probs")

# Function to predict multinomial logit choice model outcomes
# model = nnet class multinomial model
# newdata = data frame containing new values to predict
predictMNL <- function(model, newdata) {

  # Only works for neural network models
  if (is.element("nnet",class(model))) {
    # Calculate the individual and cumulative probabilities
    probs <- predict(model,newdata,"probs")
    cum.probs <- t(apply(probs,1,cumsum))

    # Draw random values
    vals <- runif(nrow(newdata))
```

```
    # Join cumulative probabilities and random draws
    tmp <- cbind(cum.probs,vals)

    # For each row, get choice index.
    k <- ncol(probs)
    ids <- 1 + apply(tmp,1,function(x) length(which(x[1:k] < x[k+1])))

    # Return the values
    return(ids)
  }
}

y1 <- predictMNL(mlt2,hungervif)
df1 <- cbind(hungervif,y=y1)
head(df1)

y2 <- predictMNL(mlt2,hungertestSelect)
df2 <- cbind(hungertestSelect,y=y2)
head(df2)

plot(df1$tubers, df1$y,col="blue",xlab="tubers", ylab="RawScore")
points(df2$tubers, df2$y,col="red",pch=3,cex=.6)
legend( x="topright",
        legend=c("Predicted","Actual"),
        col=c("red","blue"), lwd=2, lty=c(0,0), # no line type
        pch=c(3,21) ) # cross and circle


#Multinomial Regression
head(hungervif$wasting)
max(hungervif$wasting)
hungervif$wastingCat <- hungervif$wasting
hungervif$wastingCat <- round(hungervif$wastingCat,0)
hungervif$wastingCat[hungervif$wasting < 5] <- 0
hungervif$wastingCat[hungervif$wasting >= 15] <- 3
hungervif$wastingCat[hungervif$wasting >= 10] <- 2
unique(hungervif$wastingCat)
colnames(hungervif)
hungerMult <- hungervif[,c("RawScore","tubers","wastingCat")]
colnames(hungerMult)
head(hungerMult)
str(hungerMult)
hungerMult$wastingCat <- as.factor(hungerMult$wastingCat)

# Multinomial regression model
# continuous variable is the tubers
library(nnet)
with(hungerMult,table(RawScore,wastingCat))
with(hungerMult,do.call(rbind,tapply(tubers,wastingCat,function(x) c(M=mean(x), SD = sd(x)))))

hungerMult$RawScore <- as.factor(hungerMult$RawScore)

hungerMult$wastingCat2 <- relevel(hungerMult$wastingCat,ref="2")
```

```r
head(hungerMult)
mlt1<- multinom(wastingCat2 ~RawScore+tubers,data=hungerMult)
summary(mlt1)


z <- summary(mlt1)$coefficients/summary(mlt1)$standard.errors
z

#2-tailed z test
p <- (1 - pnorm(abs(z), 0, 1))*2
p

dwrite <- data.frame(RawScore = rep(c("2", "3", "4"), each = 41),
 tubers = rep(c(30:70), 3))
head(dwrite)
str(dwrite)

#dwrite <- hungervif[,c("RawScore","tubers","RawScoreOrig")]
## store the predicted probabilities for each value of ses and write
pp.write <- cbind(dwrite, predict(mlt1, newdata = dwrite, type = "probs", se = TRUE))

head(pp.write)
## calculate the mean probabilities within each level of ses
by(pp.write[,3:5], pp.write$RawScore, colMeans)


library("reshape")
lpp <- melt(pp.write, id.vars = c("RawScore", "tubers"), value.name = "probability")
head(lpp)
#str(lpp)

## plot predicted probabilities across tubers values for
## each level of ses facetted by program type
ggplot(lpp, aes(x = tubers, y = value, colour = RawScore )) +
  geom_line() +
  facet_grid(variable ~ ., scales="free")

###

colnames(hungervif)
p1 <- predict(mlt2,hungervif,"probs")



# Function to predict multinomial logit choice model outcomes
# model = nnet class multinomial model
# newdata = data frame containing new values to predict
predictMNL <- function(model, newdata) {

  # Only works for neural network models
  if (is.element("nnet",class(model))) {
    # Calculate the individual and cumulative probabilities
    probs <- predict(model,newdata,"probs")
```

```
    cum.probs <- t(apply(probs,1,cumsum))

    # Draw random values
    vals <- runif(nrow(newdata))

    # Join cumulative probabilities and random draws
    tmp <- cbind(cum.probs,vals)

    # For each row, get choice index.
    k <- ncol(probs)
    ids <- 1 + apply(tmp,1,function(x) length(which(x[1:k] < x[k+1])))

    # Return the values
    return(ids)
  }
}

y1 <- predictMNL(mlt2,hungervif)
df1 <- cbind(hungervif,y=y1)
head(df1)

y2 <- predictMNL(mlt2,hungertestSelect)
df2 <- cbind(hungertestSelect,y=y2)
head(df2)

plot(df1$tubers, df1$y,col="blue",xlab="tubers", ylab="RawScore")
points(df2$tubers, df2$y,col="red",pch=3,cex=.6)
legend( x="topright",
        legend=c("Predicted","Actual"),
        col=c("red","blue"), lwd=2, lty=c(0,0), # no line type
        pch=c(3,21) ) # cross and circle
```