Undernourishment prediction for Sub-Saharan Africa and Asia

Koen Dekeyser

14/07/2021

Contents

Contents

1 Executive summary

The world is not on track to achieve Sustainable Development Goal 2 (SDG2) to end hunger and improve sustainable agriculture by 2030. Global food insecurity has been rising since 2014 and COVID-19 could push a further 100 million people into the ranks of the food insecure. Redoubled efforts are needed to ensure our food systems are able to feed the world. The prevalence of undernourishment is an indicator that measures the proportion of the population that lacks enough dietary energy for a healthy, active life[1]. In 2015, the United Nations set the goal of zero hunger by 2030. Sadly, the opposite is likely to happen, with more undernourished people in 2030 compared to 2015. Most undernourished people reside in Asia, while the prevalence is highest in Sub-Saharan Africa. The goal of this report is to build a model that predicts the prevalence of undernourishment for Sub-Saharan Africa and Asia based on a selection of key economic, societal, and agricultural indicators.

All data comes from the World Bank Open Data repository. After pulling the data and initial wrangling, the dataset is split into a 90% training set (“edx”) and a 10% evaluation set (“evaluation). For training and regularisation, this training set is further divided in a 90% training set (”edx_train“) and a 10% evaluation set (”edx_test“). The purpose of this additional division is to only test with the”evaluation" set after the final model has been trained.

Besides this executive summary, this report has a ***Methods*** section which details how the data is pulled and wrangled, explores the data, and details the modelling approach. This is followed by a ***Results*** section, which pre-processes, builds, trains and regulates the models proposed in the Methods section. The Root Mean Square Error (RMSE) of the best performing model is calculated and evaluated with the final “evaluation” set. The final model is a ***finetuned Random forest model***, which is able to predict the prevalence of undernourishment with a rounded ***RMSE of 2.88***. The ***Conclusion*** section gives a brief summary of the report, its limitations and future work.

Important - to improve readability of this document, all code deemed non-essential, such as ggplot and kable code, has not been shown in this report (eval=T, echo=F) but can of course be found in the .rmd file. This approach has been confirmed with the teacher's assistant, and will provide a much clearer reading experience for you while focusing on the essentials.

2 Methods

2.1 Data pulling and wrangling

Seventeen key economic, societal, and agricultural indicators from the World Bank Open Data repository were chosen based on their potential to predict undernourishment. These indicators are shows in Table 1. The data come from the World Bank Data repository and is pulled in through the WDI R package. As the World Banks' servers are frequently not available, the data was saved in a csv file and uploaded to GitHub. The summary of the dataset shows that several indicators have large data gaps: for example, the indicators poverty, inequality and water use in agriculture has more than 2400 NA's (about 80%) and thus were dropped entirely from the dataset. Life expectancy was dropped because the causal linkage is quite clear: lower undernourishment would lead to longer lives, not the opposite. Lastly, only rows with undernourishment data present were kept.

```r
#URL is cut up to improve the aesthetics of the report
url1 <- "https://raw.githubusercontent.com/KoenDekeyser/EdX_DataScience/main/"
url2 <- "WDI_database.csv"
url <- paste0(url1,url2)

#initial wrangling to pivot it from tidy to long data
db_pull <- read.csv(url) %>%
  select(area, value, date, name) %>%
```

[1] FAO, IFAD, UNICEF, WFP, & WHO. (2019). The State of Food Security and Nutrition in the World 2019. Safeguarding against economic slowdowns and downturns. Rome: FAO.

```r
  pivot_wider(names_from = name, values_from = value,
              names_sep = "_") %>%
  filter(!is.na(Undernourishment))

#Replace the column headers with no-space names
names(db_pull) <-  str_replace_all(names(db_pull), c(" " = "_"))
summary(db_pull)
```

```
##      area                date      Population_growth Urban_population
##  Length:1368        Min.   :2001   Min.   :-1.551    Min.   : 13.95
##  Class :character   1st Qu.:2005   1st Qu.: 1.127    1st Qu.: 33.48
##  Mode  :character   Median :2010   Median : 1.880    Median : 49.47
##                     Mean   :2010   Mean   : 1.978    Mean   : 51.45
##                     3rd Qu.:2014   3rd Qu.: 2.743    3rd Qu.: 67.14
##                     Max.   :2018   Max.   :15.177    Max.   :100.00
##
##  Life_expectancy GDP_per_capita       GDP_growth      Agriculture_GDP
##  Min.   :40.37   Min.   :   90.53   Min.   :-36.658   Min.   : 0.0488
##  1st Qu.:59.44   1st Qu.:  787.47   1st Qu.:  2.765   1st Qu.: 5.4263
##  Median :67.78   Median : 1911.68   Median :  4.800   Median :12.8215
##  Mean   :66.25   Mean   : 6746.36   Mean   :  4.915   Mean   :16.1098
##  3rd Qu.:73.64   3rd Qu.: 5649.96   3rd Qu.:  6.920   3rd Qu.:24.7690
##  Max.   :84.93   Max.   :93022.88   Max.   : 53.382   Max.   :79.0424
##                  NA's   :19         NA's   :21        NA's   :48
##  Agricultural_jobs Food_production_index  Food_imports    Agricultural_land
##  Min.   : 0.17     Min.   : 33.19        Min.   : 1.076   Min.   : 1.954
##  1st Qu.:18.05     1st Qu.: 76.59        1st Qu.: 9.062   1st Qu.:23.215
##  Median :36.28     Median : 92.48        Median :13.897   Median :43.277
##  Mean   :35.64     Mean   : 88.85        Mean   :14.499   Mean   :42.392
##  3rd Qu.:49.37     3rd Qu.:100.94        3rd Qu.:17.829   3rd Qu.:60.401
##  Max.   :88.24     Max.   :156.23        Max.   :54.948   Max.   :83.762
##                                          NA's   :207      NA's   :28
##   Forest_land       Cereal_yield     Water_use_agriculture Undernourishment
##  Min.   : 0.0084   Min.   :   34.3   Min.   : 4.362        Min.   : 2.50
##  1st Qu.: 7.0853   1st Qu.: 1188.2   1st Qu.:55.238        1st Qu.: 5.50
##  Median :20.3936   Median : 1965.0   Median :73.301        Median :11.90
##  Mean   :25.9945   Mean   : 2788.5   Mean   :68.371        Mean   :14.69
##  3rd Qu.:41.1025   3rd Qu.: 3294.2   3rd Qu.:90.316        3rd Qu.:21.20
##  Max.   :91.9585   Max.   :36761.9   Max.   :98.431        Max.   :67.50
##  NA's   :46        NA's   :30        NA's   :1082
##     Poverty        Inequality
##  Min.   : 0.00   Min.   :25.30
##  1st Qu.: 0.50   1st Qu.:32.52
##  Median : 4.40   Median :37.00
##  Mean   :14.83   Mean   :37.70
##  3rd Qu.:21.98   3rd Qu.:41.30
##  Max.   :79.80   Max.   :64.80
##  NA's   :1022    NA's   :1022
```

There are still quite a few missing values for the other indicators. To estimate the values missing, an average is taken from each indicator grouped per area (equal to country) and this is used to replace the missing values. The logarithm of GDP per capita and cereal yield are taken as high performing countries can easily skew the data.

```r
#indicators deselected because of too many missing data
db <- db_pull %>%
  select(!c(Poverty, Inequality, Water_use_agriculture,
            Life_expectancy)) %>%
#Remaining NA's are filled up by taking the area mean of that indicator
  group_by(area) %>%
    mutate(Agricultural_land = ifelse(is.na(Agricultural_land),
                               mean(Agricultural_land, na.rm = T),
                               Agricultural_land),
         Forest_land  = ifelse(is.na(Forest_land),
                                mean(Forest_land, na.rm = T),
                               Forest_land),
         Cereal_yield  = ifelse(is.na(Cereal_yield),
                               mean(Cereal_yield, na.rm = T),
                               Cereal_yield),
         GDP_growth = ifelse(is.na(GDP_growth),
                                mean(GDP_growth, na.rm = T),
                                GDP_growth),
         Food_imports = ifelse(is.na(Food_imports),
                                mean(Food_imports, na.rm = T),
                                Food_imports),
         Population_growth = ifelse(is.na(Population_growth ),
                                 mean(Population_growth , na.rm = T),
                                 Population_growth),
         GDP_per_capita = ifelse(is.na(GDP_per_capita ),
                                  mean(GDP_per_capita , na.rm = T),
                                  GDP_per_capita),
         Agriculture_GDP = ifelse(is.na(Agriculture_GDP ),
                               mean(Agriculture_GDP , na.rm = T),
                               Agriculture_GDP),
         Agricultural_jobs = ifelse(is.na(Agricultural_jobs ),
                               mean(Agricultural_jobs , na.rm = T),
                               Agricultural_jobs),
         Food_production_index = ifelse(is.na(Food_production_index ),
                                    mean(Food_production_index , na.rm = T),
                                    Food_production_index),
         Urban_population = ifelse(is.na(Urban_population ),
                                    mean(Urban_population , na.rm = T),
                                   Urban_population)) %>%
  ungroup() %>%
#two indicators are converted to their logarithm
  mutate(GDP_per_capita_log = log(GDP_per_capita),
         Cereal_yield_log = log(Cereal_yield)) %>%
  select(!c(GDP_per_capita, Cereal_yield)) %>%
  na.omit()
rm(db_pull)
```

In the end, the dataset has 14 indicators and 1260 rows.

After this wrangling, the dataset is ready for analysis. The dataset is split in a 90% training set ("edx") and a 10% evaluation set ("evaluation"), in order to have a holdout dataset available to test the predictive strength of the model.

```r
# Validation set will be 10% of dataset
set.seed(1, sample.kind="Rounding") #if using R 3.5 or earlier, use `set.seed(1)
```

```
test_index <- createDataPartition(y = db$Undernourishment,
                                  times = 1, p = 0.1, list = FALSE)
edx <- db[-test_index,]
validation <- db[test_index,]
rm(test_index)
```

2.2 Data exploration

2.2.1 Introduction

There are 1260 rows of observations in the dataset. This has been split in a 90% training set with 1132 rows, and an validation set of 128 rows. This section focuses on describing the training dataset. There are 70 unique countries in the training set. Table 1 provides a description of all indicators.

Table 1: Description of the dataset variables

Indicator	Description
area	Country name
date	Year of the observation
Population_growth	% growth of the population
Urban_population	% of the population living in urban areas
GDP_growth	% growth of GDP
Agriculture_GDP	% of contribution of the agricultural sector to GDP
Agricultural_jobs	% of employment generated by the agricultural sector
Food_production_index	Index covering the production of edible and nutritious food crops
Food_imports	% of food imports over merchandise imports
Agricultural_land	% of land used for agriculture
Forest_land	% of land occupied by forests
Undernourishment	% of population that is undernourished
GDP_per_capita_log	Logarithm of GDP per capita, in current USD
Cereal_yield_log	Logarithm of kilograms per hectare of cereal crops

The structure of the dataset is shown in table 2, which depicts the first five rows of the training dataset and the first five columns. This shows how each row is organised according to an area and a year, together with the indicators.

Table 2: First five rows and five columns of the dataset

area	date	Population_growth	Urban_population	GDP_growth
Afghanistan	2018	2.384309	25.495	1.189228
Afghanistan	2017	2.547833	25.250	2.647003
Afghanistan	2016	2.778035	25.020	2.260314
Afghanistan	2014	3.355602	24.587	2.724543
Afghanistan	2013	3.494592	24.373	5.600745

Figure 1 presents the histograms of all indicators, showing that the number of observations per year is stable at around 60. Population growth and undernourishment have the most skewed distributions.

Figure 1. Histograms of all numeric variables

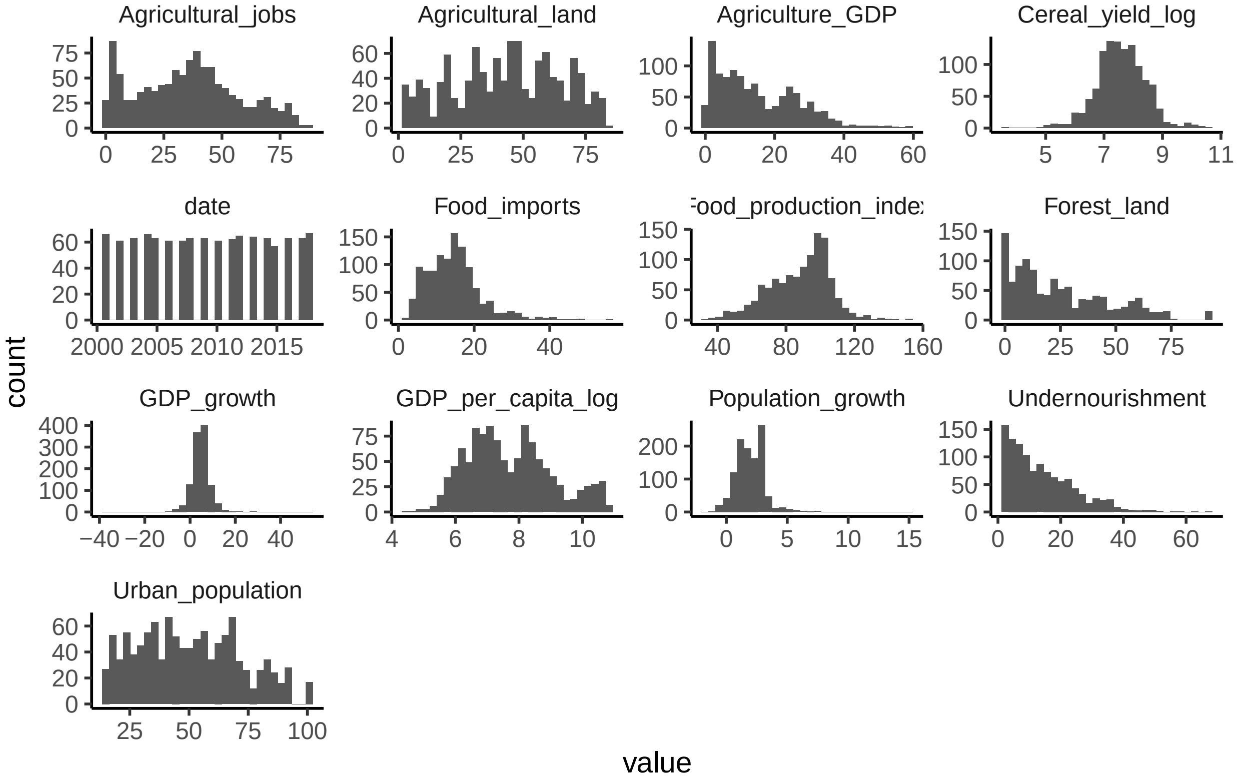


Figure 2 shows that undernourishment by country declined between 1990-2013, but then progress stagnated. Between 2016-2019, undernourishment increased.

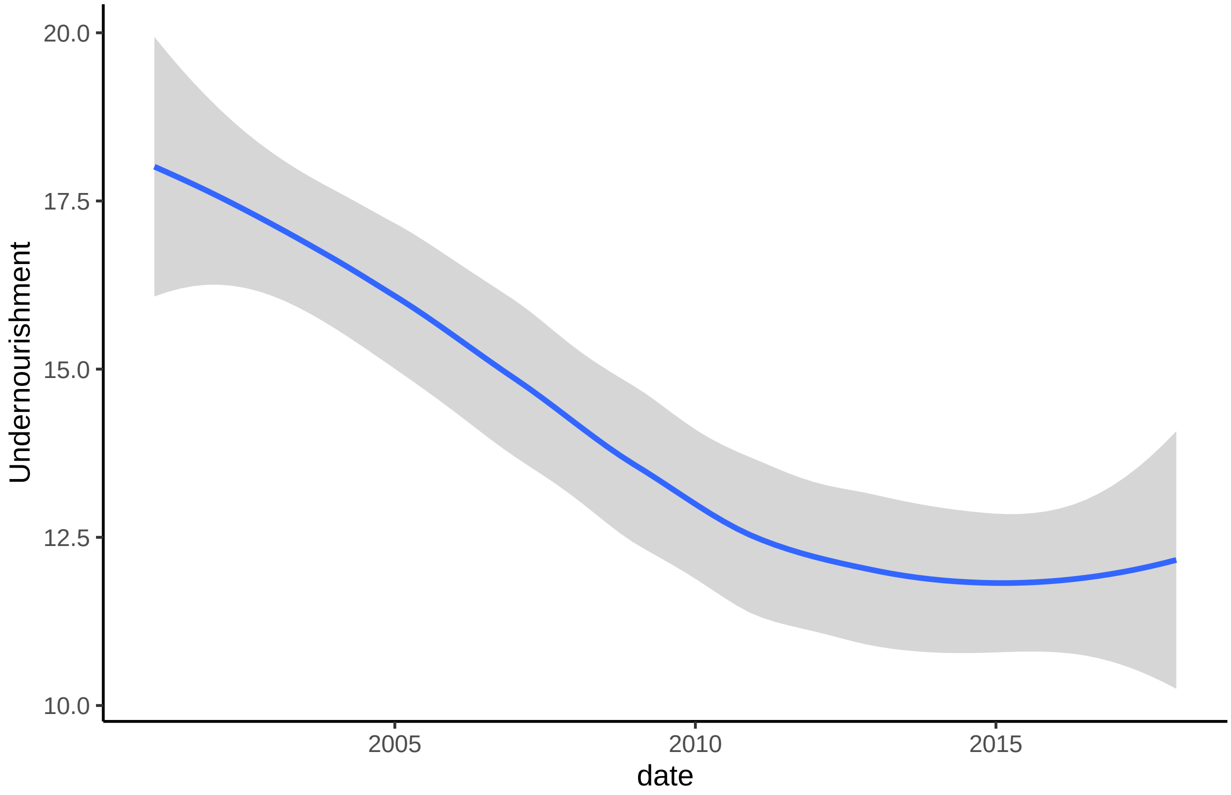
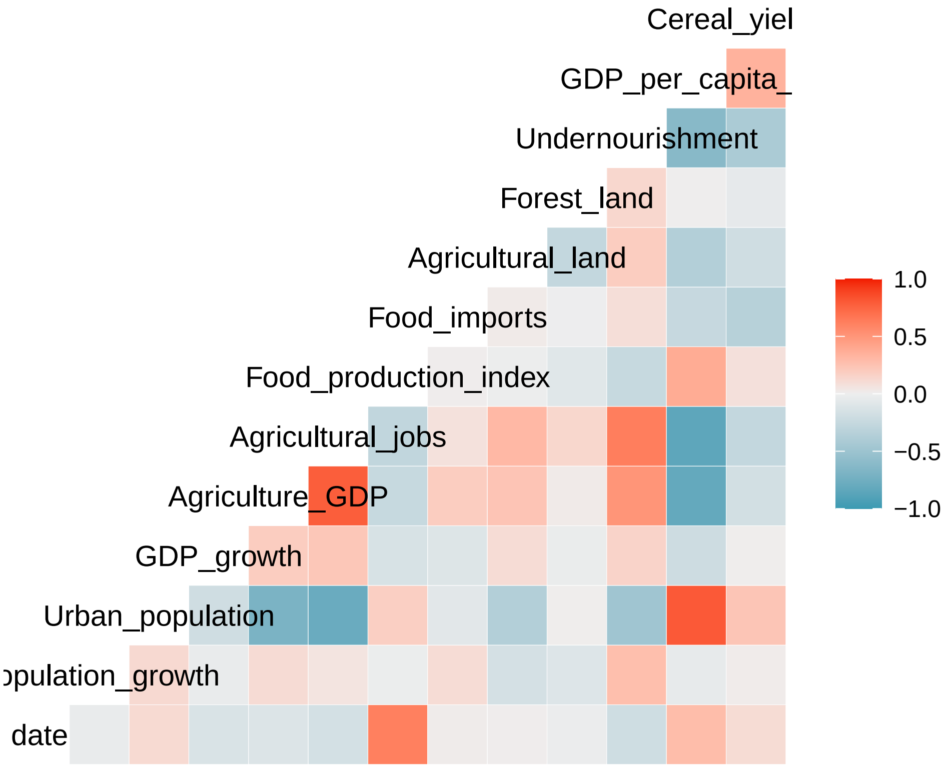


Figure 2. LOESS curve for undernourishment over years

Figure 3 shows the correlation matrix for all indicators. Undernourishment is positively correlated with agricultural jobs and agricultural GDP, while GDP per capita, GDP growth, and cereal yield are negatively correlated.

Figure 3. Correlations matrix of all indicators



2.3 Modelling approach

The exploratory analysis shows that undernourishment is correlated with a few indicators related to general wealth and agricultural economic performance. This will make it useful when building statistical models such as a general linear regression. But, intuitively, given that undernourishment declined over the years, it is surprising that time is not strongly correlated with undernourishment. More advanced machine learning models such as Random forests might have superior performance in predicting undernourishment in Asia and sub-Saharan Africa by better picking up these patterns in the data. The KNN model was used in creating this report, but because it is better for classification instead of regression, it was ultimately discarded. The performance of model prediction is evaluated through the root-mean-square error (RMSE), which measures the difference between values predicted by a model or an estimator and the values observed[2].

The RMSE function is:

```
#RMSE function
  RMSE <- function(true_undernourishment, predicted_undernourishment){
    sqrt(mean((true_undernourishment - predicted_undernourishment)^2))
  }
```

3 Results

3.1 Preparing additional data partitioning

The edx set will be further partitioned into a 90% training set (“edx_train”) and a 10% test set (“edx_test”), in order to lower (over-)training.

[2] https://en.wikipedia.org/wiki/Root-mean-square_deviation

```
#Additional evaluation dataset will be 10% of training dataset
set.seed(1, sample.kind="Rounding")#if using R 3.5 or earlier, use `set.seed(1)
test_index <- createDataPartition(y = edx$Undernourishment,
                                  times = 1, p = 0.1, list = FALSE)
edx_train <- edx[-test_index,]
edx_test <- edx[test_index,]

rm(test_index)
```

First, a generalized linear regression model employed on all continuous data.

```
#A general GLm model with all numeric indicators as predictors
glm_train <- train(Undernourishment ~ .,
                 method = "glm",
                 metric = "RMSE",
                 data = edx_train[,-1])
summary(glm_train)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -16.398   -4.609   -0.558    3.729   40.668
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)            69.487560 109.410964   0.635    0.526
## date                   -0.007564   0.055301  -0.137    0.891
## Population_growth       2.119282   0.180508  11.741  < 2e-16 ***
## Urban_population        0.006631   0.019585   0.339    0.735
## GDP_growth              0.072941   0.049754   1.466    0.143
## Agriculture_GDP        -0.029856   0.034856  -0.857    0.392
## Agricultural_jobs       0.118117   0.022771   5.187 2.58e-07 ***
## Food_production_index   0.003824   0.016447   0.233    0.816
## Food_imports           -0.185185   0.034725  -5.333 1.19e-07 ***
## Agricultural_land       0.003704   0.011831   0.313    0.754
## Forest_land             0.050838   0.011156   4.557 5.83e-06 ***
## GDP_per_capita_log     -2.898155   0.451127  -6.424 2.04e-10 ***
## Cereal_yield_log       -3.374318   0.292356 -11.542  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 47.83147)
##
##     Null deviance: 109462  on 1015  degrees of freedom
## Residual deviance:  47975  on 1003  degrees of freedom
## AIC: 6827.8
##
## Number of Fisher Scoring iterations: 2
```

```
glm_hat <- predict(glm_train, edx_test)
RMSE_glm <- RMSE(edx_test$Undernourishment, glm_hat)
```

The generalized linear model has six significant correlated variables and provides a RMSE of 8.3053835, which is way too high to make useful predictions on the undernourishment in sub-Saharan Africa and Asia. We then turn to training a Random forest model:

```
#A randomforest model primed for RMSE evaluation
set.seed(1, sample.kind="Rounding")#if using R 3.5 or earlier, use `set.seed(1)
rf_train <- train(Undernourishment ~ .,
                  method = "rf",
                  metric = "RMSE",
                  data = edx_train[,-1])
rf_hat <- predict(rf_train, edx_test)
RMSE_rf <- RMSE(edx_test$Undernourishment, rf_hat)
```

The Random forest model has a RMSE of 3.7122731, which is a large improvement (-4.5931104 better) compared to the generalized linear model.

Table 3: Performance of the models

Model	**RMSE**
Generalised linear model	8.305383
Random forest	3.712273

Table 3 shows that Random forest is the best performing model. Finetuning can enhance the predictive accuracy of a model by making small adjustments.

3.2 Finetuning

As Random forest models take long to compute, cross-validation was kept to 5 folds, with no repeats. The tuneGrid is a sequence of 4 to 10 whole numbers as the previous iteration of Random forest showed that the optimal tuning parameter would be within these bounds.

```
#finetuning
set.seed(1, sample.kind="Rounding")
#Five-fold cross validation with zero iterations
control <- trainControl(method="cv", number = 5)
#tuning parameters between 4 and 10
grid <- data.frame(mtry = seq(4,10,1))

train_rf<- train(Undernourishment ~ .,
                 method = "rf",
                 metric = "RMSE",
                 trControl = control,
                 tuneGrid = grid,
                 data = edx_train[,-1])
```

The best performing tuneGrid parameter is 5. All the parameters are now present to evaluate the final model on the holdout evaluation data ("evaluation"), by a Random forest model with a finetune parameter of 5.

3.3 Final model and validation with the validation data

```
#Final model
set.seed(1, sample.kind="Rounding")#if using R 3.5 or earlier, use `set.seed(1)
rf_fit <- randomForest(Undernourishment ~ .,
                 minNode  = train_rf$bestTune$mtry,
                data = edx_train[,-1])
rf_hat <- predict(rf_fit, validation)
RMSE_final <- RMSE(validation$Undernourishment, rf_hat)

#Show the importance
stack(importance(rf_fit)[order(importance(rf_fit), decreasing = T),])
```

```
##       values                   ind
## 1  25163.371    GDP_per_capita_log
## 2  16526.802     Agricultural_jobs
## 3  12545.336      Cereal_yield_log
## 4  11251.972     Population_growth
## 5  10267.140      Urban_population
## 6  10266.885       Agriculture_GDP
## 7   6746.954           Forest_land
## 8   5410.176     Agricultural_land
## 9   4260.391 Food_production_index
## 10  2855.433          Food_imports
## 11  1734.277            GDP_growth
## 12  1456.346                  date
```

The finetuned Random forest model has a RMSE of 2.8848627. The most important indicators for prediction are GDP per capita, agricultural jobs, cereal yield, and population growth.

4 Conclusion

The goal of this report was to build a model that predicts the prevalence of undernourishment for Sub-Saharan Africa and Asia based on a selection of key economic, societal, and agricultural indicators. After pulling and wrangling the data, the dataset was split into a 90% training and 10% evaluation set. The various elements of the dataset were then described - distribution, correlation, and structure. After presenting the modelling approach focust on the root-mean-square error evaluation, a generalized linear model and a Random forest was built. The best performing model - the Random forest model - was further finetuned through 5-fold cross-validation with no repeats.

The best performing model is a Random forest model which, after finetuning, delivers a ***RMSE of 2.8848627***. Another model calculated was a Generalized linear model with a RMSE of 8.3053835. Considering the performance of the generalized linear model, the final finetuned Random forest model is clearly superior. The machine learning model Random forest vastly outperform the regression model, with the final Random forest model being 2.8789528 times better performing than the regression model.

Table 4: Performance of all the models

Model	RMSE
Generalised linear model	8.305383
Random forest	3.712273
Tuned Random forest	2.884863

There are several clear shortcomings to this model. First, data availability: the data showed big gaps that were filed up by calculating the mean of the area. Second, predictive power: a RMSE of 2.8848627 is still too high to make meaningful predictions - and to base policy response to those numbers - in reality. Third, useability: given the vastness and difference between and within sub-Saharan Africa and Asia, a model that predicts the undernourishment for the whole two regions rather than on the country level is severely restricted in its usefulness. Undernourishment should be rather taken at country, or better, at the most local level possible to steer interventions. Perhaps the best value of this model is not necessarily its predictive power, but the selection of variables that are influential for prediction: the GDP per capita, agricultural jobs, and cereal yields. Policymakers should be sensitive to changes in these indicators for their importance to undernourishment, and this can already influence interventions for undernourishment, regardless of the specific RMSE performance of the model. For example, more sensitivity to the importance of agricultural jobs and cereal yield.

Future work - Predicting machine learning to a level that is useful to steer interventions requires more advanced models and data gathering. With the persistent and increasing challenge of undernourishment, data availability and prediction should be vastly improved. The World Bank Open Data repository is vast, but even for the more basic indicators - as selected for this report - large data gaps exist. Investments in national statistical agencies is perhaps as important, if not more, than the development of advanced machine learning in the field of development economics. Without good data, even mundaine statistical analysis becomes challenging.