HOUSEHOLD FOOD INSECURITY INDICATOR

This chapter describes the prevalence of households with moderate or severe food insecurity using the FIES, a Feed the Future phase two indicator. This chapter has two sections; the first section describes the guidelines to construct the indicator, and the second section outlines the step-by-step procedures to calculate the indicator.

14.1 Guidelines to construct the FIES indicator

This section provides the guidelines to construct the Feed the Future food insecurity indicator: the prevalence of moderate or severe food insecurity in the population, based on the Food Insecurity Experience Scale.

The prevalence of moderate or severe insecurity is calculated from respondent's raw score—that is, the number of affirmative responses given to the eight FIES questions. It is simply an integer with a value between 0 and 8, and hence there will always be up to 9 distinct values of respondent parameters, one for each possible raw score (0–8). Although the raw scores and associated respondent parameters depend only on the number of affirmative responses, certain response patterns are expected to conform to the Rasch model's assumptions—that is, when arranged in the order of increasing severity, responses start with "yes" and are followed by "no" (without alternating). This guide provides steps for statistical validation that must precede the calculation of prevalence estimates.

The Food and Agriculture Organization of the United Nations (FAO) produces two FIES indicators for global monitoring: moderate and severe food insecurity and severe food insecurity. Only the moderate and severe food insecurity indicator has been selected as a monitoring indicator for Sustainable Development Goal 2 (Target 2.1). This guide provides prevalence estimate calculations for four levels of food insecurity: little to no food insecurity, moderate food insecurity, severe food insecurity, and moderate or severe food insecurity. For further guidance, the analyst should review the guidance available at http://www.fao.org/in-action/voices-of-the-hungry/using-fies/en/.

14.2 Step-by-step procedures to calculate the FIES indicator

This section describes the detailed step-by-step procedures to calculate the prevalence of moderate or severe food insecurity in the population, based on the Food Insecurity Experience Scale.

Definitions

Numerator	Population with a probability of exceeding the food insecurity severity level
Denominator	The surveyed population
Unit of measure	Percentage
Level of data	Household
Sampling weight	Household
Disaggregation levels	Gendered household type* Wealth quintile Poverty status Shock exposure severity

Treatment of missing data	All missing and refused responses are excluded from calculations of the indicator
Survey variables used	`v301, v302, v303, v304, v305, v306, v307, v308, hhea, wgt_hh, strata`
Analytic variables used	`genhhtype_dj, edulevel_hh_dj, awiquint, poor190, shock_sev`
Analytic variables created	`worried, healthy, fewfood, skipped, ateless, runout, hungry, whlday, mem, , strata, wt (temporary dataset)`

*Standard Feed the Future disaggregate

Commented [EM1]: For discussion with RFS: This is an area for investigation. Regarding the missing data in the disaggregation variables, to avoid losing data, I used to impute the missing data. The current standard code does not impute data. Should we impute or not is an open question.

Commented [S(2]: *ahtype* already exists

Commented [PJ3R2]: Noted. Removed.

Calculations

The Voices of the Hungry project provides a free analytical tool for FIES analysis using R software.

Following the FAO technical resources, this guide describes the steps required to compute the prevalence of moderate or severe food insecurity based on the FIES scale.

Step 1. Download R, install the required packages, and set a working directory and reading data.

Step 1a. Go to https://cran.r-project.org/ to download R. After it is downloaded, R can be used to program directly in the console or through a user-friendly compiler, RStudio, which needs to be downloaded separately.

Step 1b. Go to https://www.rstudio.com/products/rstudio/download/ to download RStudio, which is an integrated development environment for R. It includes a console, a syntax-highlighting editor that supports direct code execution, and tools for plotting, tracking history, debugging, and managing the workspace.

Step 1c. Clear the R environment.

```
rm(list = ls(all = TRUE)) # to clear the R environment
```

Install the required packages the first time you use the code. Run the following code to install the "`RM.weights`," "`survey`," "`foreign`," and "`dplyr`" packages:

```
install.packages("RM.weights") # for weighted Rasch modeling and
extensions using CML
install.packages("survey") # for analysis of complex survey samples.
install.packages("haven") # to read data stored in Stata,[1] SPSS, or
other software
install.packages("tidyverse") # for data manipulation
```

After the packages are installed, upload the package in the working library by running the following code:

```
library(RM.weights)
library(survey)
library(haven)
library(tidyverse)
```

Note that you only need to install the packages above one time. However, each time before you use the code, you need to upload the packages using the *library* function.

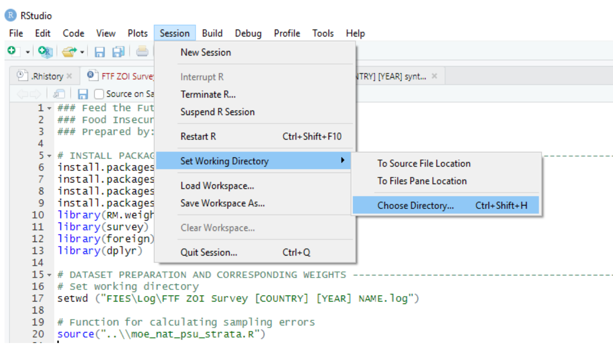
[1] Note that data in Stata version 13.0 or above should be saved in Stata versions 11.0 or 12.0 for compatibility before reading.

Step 1d. Set and specify a working directory in which the data are stored. This will serve as the root directory.

```
setwd("FIES\\FTF ZOI Survey [COUNTRY] [YEAR] NAME")
```

The working directory and folder name should be changed to reflect the directory path and name. Alternatively, go to Rstudio console to set the working directory. Click on Session, and then Set Working Directory and Choose Directory (Figure 1).

Figure 1: Setting up a Working Directory in R



Step 1e. Read the source code of the *moe* function to calculate the confidence interval and design effect. The source code "`moe_nat_psu_strata_v2.R`" should be added to the working directory, and the `moe` function should be called into the code as:

```
source("moe_nat_psu_strata_v2.R")
```

The `moe` function facilitates the calculations of the confidence interval and design effect using the *survey* package of R, which accounts for the complex survey design, such as clustering and unequal probabilities of selection, by specifying the PSU, strata, and sampling weights.

Step 1f. Read the data file. Using the haven package to read the dataset will lead to 'double' type variables with the STATA labels preserved.

```
ftfzoi_midline = read_dta("FTF ZOI Survey [COUNTRY] [YEAR] household
data analytic.dta")
```

Step 2. Prepare the data by preparing the data frame, analytic variables, recoding variables, and background characteristics.

Step 2a. To create a data frame with only the FIES variables, first rename the variables to the corresponding item using the *rename* function. Then, use the *select* function to keep the relevant variables.

```
ftfzoi_midline <-
  ftfzoi_midline %>%
  rename(
```

```
        "WORRIED" = "v301",
        "HEALTHY" = "v302",
        "FEWFOOD" = "v303",
        "SKIPPED" = "v304",
        "ATELESS" = "v305",
        "RUNOUT"  = "v306",
        "HUNGRY"  = "v307",
        "WHLDAY"  = "v308")

FIES_midline <- ftfzoi_midline %>%
        select(WORRIED:WHLDAY)
```

Step 2c. Recode all variables. Keep the 'YES' answers coded as '1,' and recode the 'NO' answers as '0.' A response 'REFUSED' or '3' will be excluded from the analysis.

```
FIES_midline <- FIES_midline %>% mutate_all(
  function(x) case_when(
    x == 1 ~ 1,
    x == 2 ~ 0
  )
)
```

Step 2d. Define variables of sampling weight and the number of de jure household members (i.e., the household size). The estimates represent the entire ZOI population and not only the individuals or the households interviewed in the survey, so the sampling weight must be applied to account for the complex sample design, and the household size should be accounted for to represent the entire ZOI population.

Commented [S(4]: These variables *wt* and *mem* are not listed in the list of variables created

```
ftfzoi_midline <- ftfzoi_midline %>%
  mutate(wt = wgt_hh,
         mem = hhsize_dj)
```

Step 2e. Store the background variables and disaggregates that will be used later for analysis. Remember to edit the variable names, values, and labels as needed.

```
wt <- wt

ftfzoi_midline <- ftfzoi_midline %>%
  mutate(fcluster = hhea,
         hhnum = hhnum,
         strata = a03c,
         region = factor(region, levels=c(1,2,3),
labels=c("region1","region2","region3")),
         urbanrural = factor(ahtype, levels=c(1,2),
labels=c("urban","rural")),
         genhhtype_dj = factor(genhhtype_dj,
                levels=c(1,2,3,4), labels=c("Male and Female
adults","Female adults only","Male adults only", "Children only"))) %>%
```

select(fcluster:genhhtype_dj, WORRIED:WHLDAY) **Step 3.** Fit the weighted Rasch model for the full sample, using household weights only.

The Rasch model (1) assesses the suitability of the items (i.e., the eight FIES variables) for scale construction, (2) computes parameter estimates and assessment statistics for each item, (3) generates a scale from the items, and (4) assesses the location of an individual or household along the continuum of the scale that uniquely reflects the food insecurity situation of that household. The model is critical for the next step, in which item parameters are equated to the FAO global scale.

```
FIES_midline_rr <- RM.w(FIES_midline, weights = ftfzoi_midline$wt,
write.file = TRUE, country = "FTF ZOI SURVEY COUNTRY")
rs = rowSums(FIES_midline)
```

The `RM.w` function automatically exports the Rasch output to the CSV file saved in the working directory. These results can be used to test the quality of data collected. The four main results obtained are as follows:

- **Infit (and corresponding standard errors):** Assesses the data for items that did not perform well in a particular ZOI population. If the infit statistic is 1.0, all items discriminate equally—that is, items are strongly or consistently related to a food insecurity condition—which is one of the main assumptions of the Rasch model.
- **Outfit (and corresponding standard errors):** Similar to infit, but sensitive to cases with unusual response patterns, even among a few respondents
- **Residual correlation matrix:** Looks at items that may be slightly redundant (i.e., that represent the same or closely related conditions caused by food insecurity)
- **Rasch reliability:** The proportion of total variance in the ZOI population that is accounted for by the measurement model. In other words, it provides information on the discriminatory power to the scale. The standard Rasch reliability statistic can be influenced by the total number of cases across the raw scores because it is weighted by the number of cases in each score. The flat Rasch reliability statistic assumes that each score has an equal number of cases, and thus it provides a comparable measure of model fit.

In addition, the Rasch model output also provides information on the raw scores, distribution of valid responses, missing data by item, and a detailed output that shows the observed response proportion and the predicted response proportion for each raw score and items. While assessing quality of the FIES data, an analyst needs to consider the following:

- An adequate fit to the Rasch model is indicated by infit and outfit statistics of 0.7-1.3 for each item. Items with infit values of more than 1.3 should not be used for scoring.
  If the item infit is greater than 1.3 for any single item, a useful first step is examining person outfit statistics to remove respondents with outlier patterns, or where responses do not follow the hierarchical logic of the item severity. To accompany this, an item outfit of >2 is considered high and may be useful for identifying specifically where the unusual patterns are located.
  Examining this with information on enumerators is also suggested to determine if these patterns emerge based on who collected the data.
  When rows are removed, the Rasch model should be re-run. If the item infit still remains above 1.3 after removing these items, the item should be dropped from the model and re-run.

- Items with infit values greater than 1.3 should be removed and the model should be re-run one by one. In other words, if more than one item has an infit value greater than 1.3, the one with the highest should be dropped. It is possible that after removing one item and re-running the Rasch model, it can lead to lower values for other items. If the other items are still greater than 1.3, then it is suggested that it be dropped as well.

If a high percentage of missing responses or a high infit affecting only one or two questions is observed, then these items may be dropped from the scale, and an analysis should be performed on the remaining six or seven items.

NOTE: Although a scale with fewer than eight items may be used for the analysis, **a minimum of five items** must be retained to produce an acceptable measure of food insecurity.[2]

Commented [ZK5]: FLAG FOR RFS: See footnote.

From Mahmoud: This is open area to investigate "how to modify the standard R code to accommodate these recommendations?"

- If an item has more than 10 percent missing responses, an analyst may decide to drop it from the scale.
- A residual correlation between a pair of items is considered high if it is >|0.3|. This shows that each item is not able to capture different aspects of food insecurity. If two items have high residual correlation, it may be necessary to drop one of them and re-run the model.
- A Rasch reliability value above 0.7 for an eight-item FIES scale is considered acceptable.
- A Rasch reliability value above 0.6 for a seven-item FIES scale is considered acceptable.

Data from the Rasch models are used in an equating process to produce prevalence estimates. This is done automatically using the `equating.fun` function in R. The same process can also be done manually using the example spreadsheet[3] (provided by FAO); this process will involve many choices that may involve a level of informed judgment and should be guided by the principles described above. The steps that follow describe the execution of the equating process and the subsequent steps in R as it is more efficient, especially with the process replicated by disaggregates, and confidence intervals and design effects are estimated.

Commented [EM6]: For RFS's consideration: This equating process is already done automatically in the R code. Although this description of the equating and prevalence estimation is good for the user to understand the process, but it might add confusion as we encountered lately with the users from Ghana. I suggest we drop this section, and we can mention that the process can be done manually as well.

Commented [S(7R6]: That makes sense. I looks like this paragraph has already been edited to reflect Mahmoud's recommendation, is that correct? Using the manual excel process is mentioned, but only steps in R are presented.

Commented [S(8]: If users use R, is use of informed judgement replaced by objective, standard criteria that are always applied?

Step 3a. The equating function is used to calibrate the measure derived from the scale to the FAO global reference scale. It then performs a probabilistic assignment of households to levels of severity along the latent trait. Next, it estimates the weighted probability of being beyond the threshold for moderate and severe food insecurity.

```
ee <- equating.fun(rr, wt.spec=wt*mem, write.file=TRUE)
ee$prevs*100 # displays the prevalence estimates
```

Note that the weights are multiplied by the household size to produce population estimates rather than household estimates.

Step 3b. Extract the probabilities for each score for moderate and severe insecurities for each case.

```
prob.rs=ee$probs.rs
prob.modsev=prob.rs[rs+1,1]                # moderate + severe
prob.sev=prob.rs[rs+1,2]                   # severe
```

[2] Currently, the FIES R code provided in the ZOI Survey Methods Toolkit does not accommodate less than eight variables. Implementing these recommendations is possible if manual equating is used (i.e., using the Excel template in the Toolkit), but not if using the R code.

[3] The spreadsheet is available at: http://www.fao.org/fileadmin/user_upload/voices_of_the_hungry/docs/EPE_Example_05.xlsx

```
prob.mod=prob.rs[rs+1,1]-prob.rs[rs+1,2]        # moderate
```

Step 3c. Obtain the margin of errors of each prevalence using the *moe* function.

```
L.modsev <- moe(prob=prob.modsev, rs=rs, wt=wt*mem, psu=fies$fcluster,
strata=fies$strata , conf.level = 0.95)
L.sev <- moe(prob=prob.sev, rs=rs, wt=wt*mem, psu=fies$fcluster,
strata=fies$strata , conf.level = 0.95)
L.mod <- moe(prob=prob.mod, rs=rs, wt=wt*mem, psu=fies$fcluster,
strata=fies$strata , con".level " 0.95)

moes <-"c("FI_"od+"=L.modse"$moe, "FI_sev"=L.sev$moe,
"FI_mod"=L.mod$moe)
moes*100
```

Step 3d. Obtain the design effect of each prevalence.

```
deffs <- c("FI_mod+"=L.modsev$deff_s, "FI_sev"=L.sev$deff_s,
"FI_mod"=L.mod$deff_s)
```

Step 3e. Obtain weighted and unweighted sample size, *nwt* and *nt*.

```
nwt <- sum((wt*mem)[!is.na(rs)])
nt <- sum((mem)[!is.na(rs)])
```

Step 4. Background characteristics and disaggregates. This guide also describes the background variables (region, urban/rural) and disaggregates required for Feed the Future reporting—gendered household type and severity (medium, severe) of food insecurity. For the purposes of the baseline report, and endline-baseline report, the analyst can add other disaggregates to the codes in Steps 2e and 4f.

NOTE: The gendered household type variable identifies each household's gendered type—that is, adult male and female, adult female-only, adult male-only, or child-only. If the number of cases is too small for any level of disaggregate, the analysis will not generate prevalence estimates for the disaggregate (gendered household type). This applies for any background disaggregates.

Step 4a. Use the `FIES_char` function to loop through all levels of background characteristics and replicate the process described above. The function returns a list of results, within an entry in the list for each individual category of the background characteristic.

```
FIES_char <- function (df.char, fies. = fies, rr. = rr, ee. = ee)
{
  list.x <- list()
  for (i in levels(df.char) ) {
    print(i)
    # selection for this individual category
    select <- !is.na(df.char) & df.char==i
```

Step 4b. Sub-set the background characteristics into the questionnaire data, the weight variable, and the de jure members, and run the Rasch model for the category.

```
xx <- fies[select,1:8]
wt <- fies$wt[select]
```

Commented [S(9]: Is this necessary anymore?

Commented [S(10]: update to reflect midline, and comprehensively cover disaggs in midline report table

Commented [S(11]: I imagine this would always be always be the case with GHHT because CNA sample is always very low.

```
mem <- fies$mem[select]

list.x[[i]]$rr <- RM.w(xx, wt, write.file = TRUE, country =
paste("COUNTRY",i))
```

NOTE: In the code above, '*i*' is the *i^th^* position for the background characteristics (e.g., urban position, rural position).

Step 4c. Calculate the prevalence of food insecurity, assuming the same model as for the full sample, applied to the sub-sample, and display the prevalence estimates. The code that follows performs a probabilistic assignment of households to levels of severity along the latent trait and estimates the weighted probability of being beyond the threshold for moderate and severe food insecurity.

```
List.x[[i]]$prev <- prob.assign(sthres = ee$adj.thres,
flex=list(a=rr$a,se.a=rr$se.a,XX=xx,wt=(wt*mem)))$sprob
list.x[[i]]$prev*100
```

Step 4d. Extract the raw scores and the probabilities for each score for moderate and severe food insecurity for each case.

```
Rs = rowSums(xx)
prob.modsev=prob.rs[rs+1,1]
prob.sev=prob.rs[rs+1,2]
prob.mod = prob.rs[rs+1,1] - prob.rs[rs+1,2]
```

Step 4e. Calculate margin of errors and design effects, taking into account the complex sampling design.

```
L.modsev <- moe(prob=prob.modsev, rs=rs, wt=wt*mem,
psu=fies$fcluster[select], strata=fies$strata[select], conf.level =
0.95)
L.sev <- moe(prob=prob.sev, rs=rs, wt=wt*mem,
psu=fies$fcluster[select], strata=fies$strata[select], conf.level =
0.95)
L.mod <- moe(prob=prob.mod, rs=rs, wt=wt*mem,
psu=fies$fcluster[select], strata=fies$strata[select], conf.level =
0.95)

 list.x[[i]]$moes <- c("FI_mod+"=L.modsev$moe, "FI_sev"=L.sev$moe,
"FI_mod"=L.mod$moe)
 list.x[[i]]$moes*100
 list.x[[i]]$deffs <- c("DF_mod+"=L.modsev$deff_s,
"DF_sev"=L.sev$deff_s, "DF_mod"=L.mod$deff_s)
```

Step 4f. Obtain weighted and unweighted sample sizes.

```
   list.x[[i]]$nwt <- sum((wt*mem)[!is.na(rs)])
   list.x[[i]]$nt <- sum((mem)[!is.na(rs)])
```

The loop for the background characteristics within the function ends with the closing brace, and then returns a list of outputs for the background characteristic.

```
  }
  return(list.x)
}
```

Step 4f. Run the `FIES_char` function to produce the results for each background characteristic.

```
list.reg <- FIES_char(fies$region)
list.res <- FIES_char(fies$urbanrural)
list.ghh <- FIES_char(fies$genhhtype_dj)
```

Additional background characteristics can be added. However, note that these should first be included under Step 2e.

Step 5. Output the results.

Step 5a The `FIES_dj` function creates a data frame of the prevalence estimates for output for a background characteristic.

```
FIES_dj <- function( df.char, list.x ) {
  df.x <- data.frame()
  for ( i in levels(df.char) ) {
    df.x <- rbind(df.x,cbind(
      "Little or no" = 100*(1-list.x[[i]]$prev[1]),
      "Moderate" = 100*(list.x[[i]]$prev[1]-list.x[[i]]$prev[2]),
      "Severe" = 100*(list.x[[i]]$prev[2]),
      "Moderate or severe" = 100*(list.x[[i]]$prev[1]),
      "Number" = sum(list.x[[i]]$nwt),
      "Unweighted Number" = sum(list.x[[i]]$nt),
      "CI Severe low"  = 100*(list.x[[i]]$prev[2]-list.x[[i]]$moes[2]),
      "CI Severe high" = 100*(list.x[[i]]$prev[2]+list.x[[i]]$moes[2]),
      "CI Moderate+ low"  = 100*(list.x[[i]]$prev[1]-
list.x[[i]]$moes[1]),
      "CI Moderate+ high" =
100*(list.x[[i]]$prev[1]+list.x[[i]]$moes[1]),
      "CI Moderate low"  = 100*((list.x[[i]]$prev[1]-
list.x[[i]]$prev[2])-list.x[[i]]$moes[3]),
      "CI Moderate high" = 100*((list.x[[i]]$prev[1]-
list.x[[i]]$prev[2])+list.x[[i]]$moes[3]),
      "DEFF Severe"  = (list.x[[i]]$deffs[2]),
      "DEFF Moderate+"  = (list.x[[i]]$deffs[1]),
      "DEFF Moderate"  = (list.x[[i]]$deffs[3])
    ))  }
  row.names(df.x) <- levels(df.char)
  return(df.x)
}
```

Step 5b. Construct the data frame of results for all background characteristics.

```
df <- data.frame()
df <- rbind(df,FIES_dj(fies$region, list.reg))
df <- rbind(df,FIES_dj(fies$urbanrural, list.res))
df <- rbind(df,FIES_dj(fies$genhhtype_dj, list.ghh))
# Add total to the bottom of the data frame
df.t <- data.frame()
prev <- ee$prevs
df.t <- cbind(
  "Little or no" = 100*(1-prev[1]),
  "Moderate" = 100*(prev[1]-prev[2]),
  "Severe" = 100*(prev[2]),
```

```
  "Moderate or severe" =100*(prev[1]),
  "Number" = sum(nwt),
  "Unweighted Number" = sum(nt),
  "CI Severe low"  = 100*(prev[2]-moes[2]),
  "CI Severe high" = 100*(prev[2]+moes[2]),
  "CI Moderate+ low"  = 100*(prev[1]-moes[1]),
  "CI Moderate+ high" = 100*(prev[1]+moes[1]),
  "CI Moderate low"  = 100*((prev[1]-prev[2])-moes[3]),
  "CI Moderate high" = 100*((prev[1]-prev[2])+moes[3]),
  "DEFF Severe" =(deffs[2]),
  "DEFF Moderate+" =(deffs[1]),
  "DEFF Moderate" =(deffs[3])
)
row.names(df.t) <- c("Total")
df <- rbind(df,df.t)

# Write the results to a CSV file
write.csv(df, file="FTF ZOI SURVEY COUNTRY FIES table.csv")
```

This code will provide prevalence estimates for four levels of food insecurity: little to no food insecurity, moderate food insecurity, severe food insecurity, and moderate or severe food insecurity. The calculations will also generate outputs for each background characteristic and the required disaggregates.

References

Ballard, T., Kepple, A.W., & Cafiero, C. (2013). *The food insecurity experience scale: Development of a global standard for monitoring hunger worldwide*. Technical Paper. Rome: FAO. Available at: http://www.fao.org/fileadmin/templates/ess/voh/FIES_Technical_Paper_v1.1.pdf

Cafiero, C., et al. (2016). *Methods for estimating comparable prevalence rates of food insecurity experienced by adults throughout the world*. Voices of the Hungry Technical Report No 1/2016. Rome: FAO. Available at: http://www.fao.org/3/a-i4830e.pdf

FAO. (2019). *Voices of the hungry: Applying the FIES*. Available at: http://www.fao.org/in-action/voices-of-the-hungry/using-fies/en/

INDDEX Project. (2018). *Food insecurity experience scale (FIES)*. Data4Diets: Building Blocks for Diet-related Food Security Analysis. Tufts University, Boston, MA. Available at: https://inddex.nutrition.tufts.edu/data4diets/indicator/food-insecurity-experience-scale-fies

Nord, M. (2014). *Introduction to item response theory applied to food security measurement: Basic concepts, parameters and statistics*. Rome: FAO. Available at: http://www.fao.org/3/a-i3946e.pdf

Viviani, S. (2016). *Manual for the implementation of the FAO Voices of the Hungry methods to estimate food insecurity: RM.weights package in R. Version 2*. Rome: FAO. Available at: http://www.sesric.org/imgs/news/1752_Manual_on_RM_Weights_Package_EN.pdf