```
##########################################
#    Bangladesh Forest Reference level    #
##########################################

# Anatoli Poultouchidou
# Gael Sola

# 20 Nov 2017

# Content
#   1. Read tables and checks
#   2. Harmonize land cover classes
#   3. Associate REDD+ activity
#   4. Calculate E/R
#   5. Export the data


###### STARTER
## - Empty the environment
rm(list=ls())

## - Remove the graphes
dev.off()

## Set the plot theme
#source("theme_gs_print.R")

## - Load libraries
library(ggplot2)
library(foreign)

## - Set the working directory
#setwd("C:/Users/NFI_058/Desktop/FRL_BD_V5")
setwd("/media/gs/OSDisk/Users/solag/Documents/Work/Bangladesh FRL")


#-----------------------------------------------------------------------#
#                     1. Read tables and checks
#-----------------------------------------------------------------------#

poly_full <- read.dbf("5_LC_2000_2015_FRL_LC_TC_Zone.dbf", as.is = TRUE)
agb <- read.csv("agb_FRL_alltrees_selectNMFandFP_FINAL.csv", stringsAsFactors = F)
harmo <- read.csv("harmo_lcover_04dec.csv", stringsAsFactors = F)

## Check
str(poly_full)

## Check the LCCS classes in the polygons
data.frame(lc_00 = sort(unique(poly_full$FRL_LC2000)),
           lc_15 = sort(unique(poly_full$FRL_LC2015))
           )

## Check LCCS classes with agb
sort(unique(agb$class))

## Check NAs
length(poly_full[which(is.na(poly_full$FRL_LC2000)),"OBJECTID"])
length(poly_full[which(is.na(poly_full$FRL_LC2015)),"OBJECTID"])

## Check area
sum(poly_full$Shape_Area)/10000 # 14,768,227 ha


#-----------------------------------------------------------------------#
#                  2. Harmonize land cover classes                      #
#-----------------------------------------------------------------------#

## Harmonize the classes in AGB file
agb_harmo <- agb
```

```
agb_harmo[which(agb_harmo$class == "Cultivated Trees"                       ), "class"] <- "Orchard
agb_harmo[which(agb_harmo$class == "Forest Tree Dominated Area (Terrestrial)"), "class"] <- "Forest
agb_harmo[which(agb_harmo$class == "Shrub Dominated Area"                   ), "class"] <- "Shrub D

sort(unique(agb_harmo$class))

## Create a table to associate each class with a code (1 -> 15)
sort(unique(poly_full$FRL_LC2000))

code <- data.frame(
  text = c("Forest Tree Dominated Area (Aquatic/ Regularly Flooded) (Swamp Forest, Swamp Plantation)
           "Forest Tree Dominated Area (Terrestrial) and Cultivated Trees (Hill Forest, Bamboo Fores
           "Herbaceous Crops",
           "Herb Dominated Area",
           "Mangrove Forest",
           "Mangrove Plantation",
           "Mud Flats or Intertidal Area",
           "Non vegetated",
           "Orchards and Other Plantations (Trees)",
           "Plain Land Forest (Sal Forest)",
           "Rivers and Khals",
           "Rubber Plantation",
           "Rural Settlement",
           "Shrub Dominated Area (Terrestrial, Orchards and Other Plantations (Shrub), Shifting Cult
           "Swamp Reed Land"),
  code = c("06","01","09","08","02","03","10","11","07","04","12","05","13","15","14")
  )

code$text_full <- paste(code$code, code$text, sep="-")

## Associate each class with its code
poly_full <- merge(poly_full, code, by.x = "FRL_LC2000", by.y = "text")
names(poly_full)[8] <- "code_lc00"
names(poly_full)[9] <- "desc_lc00_full"

poly_full <- merge(poly_full, code, by.x = "FRL_LC2015", by.y = "text")
names(poly_full)[10] <- "code_lc15"
names(poly_full)[11] <- "desc_lc15_full"

names(poly_full)

## Re-order and change column names
poly <- poly_full[,c("OBJECTID" ,
                     "code_lc00", "FRL_LC2000", "desc_lc00_full",
                     "code_lc15", "FRL_LC2015", "desc_lc00_full",
                     "ptc_2000" , "ptc_2015"  , "Zone"          , "Shape_Area")]

names(poly) <- c("id_poly"  ,
                 "code_lc00", "desc_lc00_harmo", "desc_lc00_full",
                 "code_lc15", "desc_lc15_harmo", "desc_lc15_full",
                 "ptc_2000" ,"ptc_2015"        , "bfi_zone"      , "area_ha")

## Convert m2 to ha and reduce the nb of digits
poly$area_ha  <- round(poly$area_ha/10000, digits = 5)
poly$ptc_2000 <- round(poly$ptc_2000     , digits = 1)
poly$ptc_2015 <- round(poly$ptc_2015     , digits = 1)

## Calculate delta_ptc (2000 - 2015, if > 0 Degradation, if < 0 Enhancement)
poly$delta_ptc <- poly$ptc_2000 - poly$ptc_2015

## Back up
poly_harmo <- poly

#-----------------------------------------------------------------#
#         3. Associate REDD+ activity
#-----------------------------------------------------------------#

## Define forest and non_forest
unique(poly_harmo$desc_lc00_harmo)
```

```
forest <- c("Forest Tree Dominated Area (Aquatic/ Regularly Flooded) (Swamp Forest, Swamp Plantation
            "Forest Tree Dominated Area (Terrestrial) and Cultivated Trees (Hill Forest, Bamboo Fore
            "Mangrove Forest",
            "Mangrove Plantation",
            "Plain Land Forest (Sal Forest)",
            "Rubber Plantation"
            )

non_forest <- c("Herbaceous Crops",
                "Herb Dominated Area",
                "Mud Flats or Intertidal Area",
                "Non vegetated",
                "Orchards and Other Plantations (Trees)",
                "Rivers and Khals",
                "Rural Settlement",
                "Shrub Dominated Area (Terrestrial, Orchards and Other Plantations (Shrub), Shiftin
                "Swamp Reed Land"
                )

## initiate REDD+ Activity
poly_harmo$redd_activity <- NA

## Associate deforestation
poly_harmo[poly_harmo$desc_lc00_harmo %in% forest     &
           poly_harmo$desc_lc15_harmo %in% non_forest  , "redd_activity"] <- "deforestation"

## Associate reforestation
poly_harmo[poly_harmo$desc_lc00_harmo %in% non_forest &
           poly_harmo$desc_lc15_harmo %in% forest      , "redd_activity"] <- "reforestation"

## Associate stable non forest
poly_harmo[poly_harmo$desc_lc00_harmo %in% non_forest &
           poly_harmo$desc_lc15_harmo %in% non_forest  , "redd_activity"] <- "non-forest stable"

## Associate stable forest
poly_harmo[poly_harmo$desc_lc00_harmo %in% forest &
           poly_harmo$desc_lc15_harmo %in% forest &
           poly_harmo$delta_ptc >= -5             &
           poly_harmo$delta_ptc <= 5               , "redd_activity"] <- "forest stable"

## Associate degradation low
poly_harmo[poly_harmo$desc_lc00_harmo %in% forest &
           poly_harmo$desc_lc15_harmo %in% forest &
           poly_harmo$delta_ptc >  5              &
           poly_harmo$delta_ptc <= 25              , "redd_activity"] <- "degradation low"

## Associate degradation high
poly_harmo[poly_harmo$desc_lc00_harmo %in% forest &
           poly_harmo$desc_lc15_harmo %in% forest &
           poly_harmo$delta_ptc > 25               , "redd_activity"] <- "degradation high"


## Associate restoration low
poly_harmo[poly_harmo$desc_lc00_harmo %in% forest &
           poly_harmo$desc_lc15_harmo %in% forest &
           poly_harmo$delta_ptc <  -5             &
           poly_harmo$delta_ptc >= -25             , "redd_activity"] <- "restoration low"

## Associate restoration high
poly_harmo[poly_harmo$desc_lc00_harmo %in% forest &
           poly_harmo$desc_lc15_harmo %in% forest &
           poly_harmo$delta_ptc < -25              , "redd_activity"] <- "restoration high"

## Control
summary(factor(poly_harmo$redd_activity))

## Area per activity
test <- aggregate(area_ha~redd_activity, data = poly_harmo, FUN = sum)
```

```
test$area_ha <- round(test$area_ha,0)
test

## Area total
sum(poly_harmo$area_ha)
sum(poly_full$Shape_Area/10000)


#-------------------------------------------------------------------#
#            3. Calculate avg biomass per ptc
#-------------------------------------------------------------------#

## Intiate table of info per land cover
area_lc00        <- aggregate(area_ha~desc_lc00_harmo, data = poly_harmo, FUN = sum)
names(area_lc00) <- c("desc_lc", "area_2000")
area_lc15        <- aggregate(area_ha~desc_lc15_harmo, data = poly_harmo, FUN = sum)
names(area_lc15) <- c("desc_lc", "area_2015")

ptc00_lc00        <- Reduce(function(...)merge(..., by = "desc_lc00_harmo"),
                            list(a = aggregate(ptc_2000~desc_lc00_harmo, data = poly_harmo, FUN = me
                                 b = aggregate(ptc_2000~desc_lc00_harmo, data = poly_harmo, FUN = sd
                                 c = aggregate(ptc_2000~desc_lc00_harmo, data = poly_harmo, FUN = mi
                                 d = aggregate(ptc_2000~desc_lc00_harmo, data = poly_harmo, FUN = ma
                            )
)
names(ptc00_lc00) <- c("desc_lc", "ptc_2000_avg", "ptc_2000_sd", "ptc_2000_min", "ptc_2000_max")

ptc15_lc15        <- Reduce(function(...)merge(..., by = "desc_lc15_harmo"),
                            list(a = aggregate(ptc_2015~desc_lc15_harmo, data = poly_harmo, FUN = me
                                 b = aggregate(ptc_2015~desc_lc15_harmo, data = poly_harmo, FUN = sd
                                 c = aggregate(ptc_2015~desc_lc15_harmo, data = poly_harmo, FUN = mi
                                 d = aggregate(ptc_2015~desc_lc15_harmo, data = poly_harmo, FUN = ma
                            )
)
names(ptc15_lc15) <- c("desc_lc", "ptc_2015_avg", "ptc_2015_sd", "ptc_2015_min", "ptc_2015_max")

## Merge into one table
info <- Reduce(function(...)merge(..., by = "desc_lc"),
               list(area_lc00, area_lc15, ptc00_lc00, ptc15_lc15)
)

## Round values to 0 decimals
info[,2:dim(info)[2]] <- round(info[,2:dim(info)[2]], 0)

## Check
info
summary(info)

## Remove temp object
rm(area_lc00, area_lc15, ptc00_lc00, ptc15_lc15)


# Merge agb with info per harmonized land cover
info2 <- merge(info, agb_harmo[,c("class", "agb_class_w", "ci95_mean_w") ],
               by.x = "desc_lc", by.y = "class", all.x = T)


## Check
info2[,c("desc_lc", "area_2000", "area_2015", "agb_class_w", "ci95_mean_w")]

## Get the weights for calculation avg agb per ptc
area_ptc00 <- aggregate(area_ha*ptc_2000~desc_lc00_harmo, data = poly_harmo, FUN=sum)
names(area_ptc00) <- c("desc_lc", "areaxptc00")
area_ptc15 <- aggregate(area_ha*ptc_2015~desc_lc15_harmo, data = poly_harmo, FUN=sum)
names(area_ptc15) <- c("desc_lc", "areaxptc15")

## Regroup with area per desc_lc and remove temp files
info3  <- Reduce(function(...)merge(..., by = "desc_lc"),
                 list(info2, area_ptc00, area_ptc15)
```

```
                )

info3
rm(area_ptc00,area_ptc15)

## Calculate weighted average ptc
info3$w_avg_ptc <- round((info3$areaxptc00+info3$areaxptc15)/(info3$area_2000+info3$area_2015),0)

## Calculate agb per ptc
info3$agb_ptc <- round(info3$agb_class_w/(info3$w_avg_ptc),3)

## Replace NA with 0
summary(info3$agb_ptc)
info3$agb_ptc <- ifelse(is.na(info3$agb_ptc), 0, info3$agb_ptc)

## Write data
write.csv(info3, "info per harmonized land class.csv", row.names = F)


#-------------------------------------------------------------------#
#                    4. Calculate E/R
#-------------------------------------------------------------------#

## Convert agb confint to percentage
info3$ci_agb_p <- round(info3$ci95_mean_w/info3$agb_class_w*100, 0)
info3[,c("agb_class_w", "ci_agb_p")]

## Add agb_ptc to each polygon
test <- merge(poly_harmo, info3[,c("desc_lc", "agb_ptc", "ci_agb_p")], by.x = "desc_lc00_harmo", by
names(test)[c(dim(test)[2]-1, dim(test)[2])] <- c("agb_ptc_00", "ci_agb_00")
str(test)

test2 <- merge(test, info3[,c("desc_lc", "agb_ptc", "ci_agb_p")], by.x = "desc_lc15_harmo", by.y = "
names(test2)[c(dim(test2)[2]-1, dim(test2)[2])] <- c("agb_ptc_15", "ci_agb_15")
str(test2)

## Change the table name back to poly_harmo2
poly_harmo2 <- test2

## Calculate the polygon carbon stock in ton biomass/ha
poly_harmo2$agb_00 <- round(poly_harmo2$ptc_2000*poly_harmo2$agb_ptc_00,3)
poly_harmo2$agb_15 <- round(poly_harmo2$ptc_2015*poly_harmo2$agb_ptc_15,3)

## Calculate ef/rf per polygon (1.2 agb+bgb, 0.47 carbon fraction, 44/12 convert C to CO2)
poly_harmo2$efrf <- (poly_harmo2$agb_00 - poly_harmo2$agb_15)*1.2*0.47*44/12
poly_harmo2$efrf <- round(poly_harmo2$efrf, 3)

## Calculate the confidence interval of the ef/rf: sum/sub ci%(A+B) = sqrt((ci%A*A)^2 + (ci%B*B)^2)/
poly_harmo2$efrf_ci <- sqrt((poly_harmo2$ci_agb_00*poly_harmo2$agb_00)^2 + (poly_harmo2$ci_agb_15*po

## Calculate e/r per polygon
poly_harmo2$er <- poly_harmo2$area_ha * poly_harmo2$efrf

## Check
summary(poly_harmo2)

#-------------------------------------------------------------------#
#                 5. export the data
#-------------------------------------------------------------------#

## Prepare the sum of confidence intervals
poly_harmo2$sum_sq_ci_efrf <- (poly_harmo2$efrf_ci*poly_harmo2$efrf)^2

## Count the number of polygons
poly_harmo2$count <- 1

## Aggregate polygons per REDD+ activity, land class change and BFI zones to prepare the land use ch
er_zone <- Reduce(function(...)merge(..., by = c("bfi_zone","code_lc00","code_lc15","redd_activity")
                  list(a = aggregate(formula = count ~ bfi_zone+code_lc00+code_lc15+redd_activity,
```

```
                                data    = poly_harmo2,
                                FUN     = sum),
                     b = aggregate(formula = area_ha ~ bfi_zone+code_lc00+code_lc15+redd_activity,
                                data    = poly_harmo2,
                                FUN     = sum),
                     c = aggregate(formula = er ~ bfi_zone+code_lc00+code_lc15+redd_activity,
                                data    = poly_harmo2,
                                FUN     = sum),
                     d = aggregate(formula = efrf ~ bfi_zone+code_lc00+code_lc15+redd_activity,
                                data    = poly_harmo2,
                                FUN     = sum),
                     e = aggregate(formula = sum_sq_ci_efrf ~ bfi_zone+code_lc00+code_lc15+redd_ac
                                data    = poly_harmo2,
                                FUN     = sum)
                     )
                )



## Check
summary(er_zone)

## Check FREL/FRL
sum(er_zone[which(er_zone$redd_activity != "non-forest stable" & er_zone$er > 0), "er"])/15
sum(er_zone[which(er_zone$redd_activity == "deforestation"), "er"])/15
sum(er_zone[which(er_zone$redd_activity == "degradation high"), "er"])/15
sum(er_zone[which(er_zone$redd_activity == "degradation low"), "er"])/15

## Confidence interval of the efrf
er_zone$ci_efrf <- round(sqrt(er_zone$sum_sq_ci_efrf)/abs(er_zone$efrf),1)

## Write table
write.csv(er_zone, "er_zone.csv", row.names = F)

## Aggregate polygons per REDD+ activity and land class change (no BFI zones) to prepare the land us
er_national <- Reduce(function(...)merge(..., by = c("code_lc00","code_lc15","redd_activity"), all =
                      list(a = aggregate(formula = count ~code_lc00+code_lc15+redd_activity,
                                         data    = poly_harmo2,
                                         FUN     = sum),
                           b = aggregate(formula = area_ha ~ code_lc00+code_lc15+redd_activity,
                                         data    = poly_harmo2,
                                         FUN     = sum),
                           c = aggregate(formula = er ~ code_lc00+code_lc15+redd_activity,
                                         data    = poly_harmo2,
                                         FUN     = sum),
                           d = aggregate(formula = efrf ~ code_lc00+code_lc15+redd_activity,
                                         data    = poly_harmo2,
                                         FUN     = sum),
                           e = aggregate(formula = sum_sq_ci_efrf ~ code_lc00+code_lc15+redd_activit
                                         data    = poly_harmo2,
                                         FUN     = sum)
                           )
                      )

## Check
sum(er_national[which(er_national$redd_activity == "deforestation"), "er"])/15

## Write table
write.csv(er_national, "er_national.csv", row.names = F)
```