Publication: Doggart, N., Mugasha, W.A., Mpiri, A., Morgan-Brown, T., Sallu, S.M., Spracklen, D.V., Agricultural fallows are the main driver of natural forest regeneration in Tanzania.

Script author: W. A. Mugasha

setwd("c:/RData/remnants")

###packages/libraries to be used

library(dplyr)

library(rio)

library(data.table)

library(tidyr)

library(stringr)

##There are three main datasets to be analysed (1) Tree Data; (2) Stump data; (3) regeneration from the subplot

## (1) Tree Data#####

###There are three dataset which need to be merged since we need all variables in a single file.

#Files are:

#(i)Regeneration\_Plot\_Form.csv

#(ii)Regeneration\_Plot\_Form-stems.csv

#(iii)Regeneration\_Plot\_Form-trees.csv

###Importing the Files in R

plots = read.csv(file = "Regeneration\_Plot\_Form.csv")

stems = read.csv(file = "Regeneration\_Plot\_Form-stems.csv")

trees = import(file = "Regeneration\_Plot\_Form-trees.csv")

##Checking the forest cover

table(trees$forest\_cover)

##the common variable will be used to merge the file: we merge 2 files at a time.

##we start with stems and trees file: common variable in Tree = KEY; common variable in Stem = PARENT\_KEY

##However, we have the variable called KEY in "stems" data and variable PARENT\_KEY in "trees" data. To avoid confusion it has to be renamed as KEY <- KY for "stems" and PARENT\_KEY <- PARENT\_KY for "trees" data.

##the variable name have been changed manually in excel

##merging Stems and Trees.

tree\_stem = merge(stems, trees, by.x = c("PARENT\_KEY"), by.y = c("KEY"), all.x = TRUE) ##all.x = TRUE: the number of observation will be dictated by observations in "stems" data.

##Now we need plot information that are missing in our data "tree\_stem". This information is available in "plots" data.

##the variable "PARENT\_KY" is the common variable in "tree\_stem" data that will help us merge with "plots" data using the common variable called "KEY"

plot\_tree = merge(plots, tree\_stem, by.x = c("KEY"), by.y = c("PARENT\_KY"), all.y = TRUE) ##all.y = TRUE: the number of observation will be dictated by observations in "tree\_stem" data.

##Now we can proceed with analysis

##since the stand variables will be presented per ha basis, we need to compute the area of the plot.

## Calculate plot area: this will vary depending on the location of the tree from the plot centre because the concentric plot design was used

## In small plot (radius = 1m): trees with dbh less than 5 cm; and large plot (radius = 15m): tree which have dbh >= 5cm.

###prepare important variables for computation of volume and biomass

##convert "cbh: circumference" to diameter "D: diameter at breast height (DBH)"

plot\_tree$D = plot\_tree$cbh/pi

###wood basic density: we have to import Naforma file with all the names

naforma = import("naforma\_names.csv")

plot\_tree.1 = merge(plot\_tree, naforma, by.x = c("latin\_name\_id"), by.y = c("abre"), all.x = TRUE)

plot\_tree.1$new\_name = ifelse(plot\_tree.1$latin\_name\_id == "other", plot\_tree.1$latin\_name\_other, plot\_tree.1$botanical\_name)

##separate the new\_name into species and genus

# load stringr library

#install.packages("stringr")

library(stringr)

# Split name column into firstname and last name

plot\_tree.1[c('genus', 'species')] <- str\_split\_fixed(plot\_tree.1$new\_name, " ", 2)

###getting the wood density using the BIOMASS package that extract the values from the data base

library("BIOMASS")

library(terra)

#install.packages("terra", type = "binary")

Taxo <- correctTaxo(genus = plot\_tree.1$genus, species = plot\_tree.1$species)

plot\_tree.1$genusCorr <- Taxo$genusCorrected

plot\_tree.1$speciesCorr <- Taxo$speciesCorrected

plot\_tree.1$nameModified = Taxo$nameModified

# Retrieving APG III Families and Orders from Genus names

APG <- getTaxonomy(plot\_tree.1$genusCorr, findOrder = TRUE)

plot\_tree.1$familyAPG <- APG$family

plot\_tree.1$orderAPG <- APG$order

# 2-RETRIEVE WOOD DENSITY

dataWD <- getWoodDensity(

 genus = plot\_tree.1$genusCorr,

 species = plot\_tree.1$speciesCorr,

 stand = plot\_tree.1$plotID)

plot\_tree.1$WD = dataWD$meanWD

plot\_tree\_stem = plot\_tree.1

##establishing the variable "area"

plot\_tree\_stem$area = with(plot\_tree\_stem, ifelse(D<4.7, pi\*(1/100)^2, pi\*(15/100)^2))

plot\_tree\_stem$ht = with(plot\_tree\_stem, ifelse(cover\_id =="BA", 1.3+37.0396\*(1-exp(-0.03778\*D^0.6063)),

 ifelse(cover\_id=="CL", 1.3+24.9862\*(1-exp(-0.0579\*D^0.7682)),

 ifelse(cover\_id =="BT", 1.3+37.0396\*(1-exp(-0.03778\*D^0.6063)),0))))

##computation of stand parameters, i.e. basal area, biomass, volume and number of trees per ha

library(dplyr)

plot\_tree\_stem.1 = mutate(plot\_tree\_stem, G = ifelse(!is.na(D), 0.0000785\*D^2/area,0),

 B = ifelse(!is.na(D) & cover\_id =="MW", 0.1027\*D^2.479/1000/area,

 ## Mugasha et al., 2013

 ifelse(!is.na(D) & cover\_id =="BA", 0.0292\*D^2.0647\*ht^1.0146/1000/area,

 ## Mugasha et al., 2016 (Book Chapter)

 ifelse(!is.na(D) & cover\_id =="CL", 0.0873\*(WD\*D^2\*ht)^0.9458/1000/area, ##Mugasha et al., 2016 (Article)

 ifelse(!is.na(D) & cover\_id =="BT", 1.2013\*D^1.5076/1000/area,0)))), ##Makero et al., 2016

 N = ifelse(!is.na(D), 1/area,0),

 V = ifelse(!is.na(D) & cover\_id =="MW", (0.00016\*D^2.463)/area,

 ## ### Mauya et al., 2014

 ifelse(!is.na(D) & cover\_id =="BA", 0.00009\*D^2.0993\*ht^0.4914/area,

 ## Mugasha et al., 2016 (Book Chapter)

 ifelse(!is.na(D) & cover\_id =="CL", 0.000076\*D^2.3488\*ht^0.3848/area, ##Mugasha et al., 2016 (Article)

 ifelse(!is.na(D) & cover\_id =="BT", 0.00042\*D^1.5009\*ht^0.6419/area,0))))) ##Makero et al., 2016

##insert diameter classes

plot\_tree\_stem.1$class = with(plot\_tree\_stem.1, ifelse(D<4.7, "0-5",

 ifelse(D<10, "5-10",

 ifelse(D<15, "10-15",

 ifelse(D<20, "15-20",

 ifelse(D<25, "20-25",

 ifelse(D<30, "25-30", "30>")))))))

###Overall stand parameters per ha for trees (live and dead)

stand\_trees = plot\_tree\_stem.1%>%

 group\_by(location.plot\_id, location.village\_id, health\_id)%>%

 summarise(Basal\_area = sum(G),

 Biomass = sum(B),

 stems = sum(N),

 Volume = sum(V))

###Overall stand parameters per ha for trees (combining live and dead)

stand\_trees\_all = plot\_tree\_stem.1%>%

 group\_by(location.plot\_id, location.village\_id)%>%

 summarise(Basal\_area = sum(G),

 Biomass = sum(B),

 stems = sum(N),

 Volume = sum(V))

export(stand\_trees, "stand\_parameters\_trees\_based\_health\_status - 070522.xlsx")

##Assess distribution by diameter classes

stand\_trees.1 = plot\_tree\_stem.1%>%

 group\_by(location.plot\_id,class)%>%

 summarise(stems = sum(N)/18)

library(rio)

##Export to excel to plot class versus stems

export(stand\_trees.1, "class\_stems.xlsx")

#####WORKING WITH STUMPS: BIOMASS LOST (emission) THROUGH TREE CUTTING####

###Importing the Files in R

stumps = read.csv(file = "Regeneration\_Plot\_Form-stumps.csv")

###Include the vegetation type from the tree data

cover = plot\_tree\_stem[,c("KEY", "plot\_id1", "cover\_id", "forest\_cover", "location.slope\_radius")]

##remove duplicates

library(data.table)

cover.1 = unique(setDT(cover), by = c("KEY", "plot\_id1", "cover\_id", "forest\_cover", "location.slope\_radius")) ## remove duplicates values

plots\_stump = merge(cover.1, stumps, by.x = c("KEY"), by.y = c("PARENT\_KEY"), all.y = TRUE) ##all.y = TRUE since the "stumps" data is the determinant of number of observations in the resulting "plots\_stumps" data

###get wood density process#####

###the names in our data base have to be translated in the long form: the long form of the tree name are found in NAFORMA data

naforma = import("naforma\_names.csv")

plot\_stump.1 = merge(plots\_stump, naforma, by.x = c("stump\_latin\_name\_id"), by.y = c("abre"), all.x = TRUE)

plot\_stump.1$new\_name = ifelse(plot\_stump.1$stump\_latin\_name\_id == "other", plot\_stump.1$stump\_latin\_name\_other, plot\_stump.1$botanical\_name)

##separate the new\_name into species and genus

# load stringr library

library(stringr)

# Split name column into genus and species

plot\_stump.2 = plot\_stump.1%>%

 separate(new\_name, c('genus', 'species'))

library(BIOMASS)

Taxo <- correctTaxo(genus = plot\_stump.2$genus, species = plot\_stump.2$species)

plot\_stump.2$genusCorr <- Taxo$genusCorrected

plot\_stump.2$speciesCorr <- Taxo$speciesCorrected

plot\_stump.2$nameModified = Taxo$nameModified

# Retrieving APG III Families and Orders from Genus names

APG <- getTaxonomy(plot\_stump.2$genusCorr, findOrder = TRUE)

plot\_stump.2$familyAPG <- APG$family

plot\_stump.2$orderAPG <- APG$order

# 2-RETRIEVE WOOD DENSITY

dataWD <- getWoodDensity(

 genus = plot\_stump.2$genusCorr,

 species = plot\_stump.2$speciesCorr,

 stand = plot\_stump.2$plotID)

plot\_stump.2$WD = dataWD$meanWD

########################

###################

#############

##Converts the stump perimeter to diameter

plots\_stumps = plot\_stump.2

plots\_stumps$SD = plots\_stumps$stump\_cbh/pi

##Plot area

plots\_stumps$area = pi\*(15/100)^2

names(plots\_stumps)

####Preparation of important variables for volume and biomass estimation

###estimate D: DBH from the SD: stump Diameter

plots\_stumps$D = -0.8812+0.86696\*(plots\_stumps$stump\_cbh/pi)

##estimate height

plots\_stumps$ht = with(plots\_stumps, ifelse(cover\_id =="BA", 1.3+37.0396\*(1-exp(-0.03778\*D^0.6063)),

 ifelse(cover\_id=="CL", 1.3+24.9862\*(1-exp(-0.0579\*D^0.7682)),

 ifelse(cover\_id =="BT", 1.3+37.0396\*(1-exp(-0.03778\*D^0.6063)),0))))

plots\_stumps.1 = mutate(plots\_stumps, G = ifelse(!is.na(D), 0.0000785\*D^2/area,0),

 B = ifelse(cover\_id =="MW", 0.1027\*D^2.479/1000/area, ## Mugasha et al., 2013

 ifelse(cover\_id =="BA", 0.0292\*D^2.0647\*ht^1.0146/1000/area, ## Mugasha et al., 2016 (Book Chapter)

 ifelse(cover\_id =="CL", 0.0873\*(WD\*D^2\*ht)^0.9458/1000/area, ##Mugasha et al., 2016 (Article)

 ifelse(cover\_id =="BT", 1.2013\*D^1.5076/1000/area, ##Makero et al., 2016

 ifelse(is.na(cover\_id),0.1027\*D^2.479/1000/area,0))))), ## Mugasha et al., 2013

 N = ifelse(D>0, 1/area,0),

 V = ifelse(cover\_id =="MW", 0.00016\*D^2.463/area, ## ### Mauya et al., 2014

 ifelse(cover\_id =="BA", 0.00009\*D^2.0993\*ht^0.4914/area, ## Mugasha et al., 2016 (Book Chapter)

 ifelse(cover\_id =="CL", 0.000076\*D^2.3488\*ht^0.3848/area, ##Mugasha et al., 2016 (Article)

 ifelse(cover\_id =="BT", 0.00042\*D^1.5009\*ht^0.6419/area, ##Makero et al.,

 ifelse(is.na(cover\_id),0.00016\*D^2.463/area,0)))))) ## ### Mauya et al., 2014

##Compute per plot stand parameters (N and Biomass) of the stumps

table(plots\_stumps.1$cover\_id)

stump\_cover.2 = plots\_stumps.1%>%

 group\_by(plot\_id1)%>%

 summarise(Basal\_area\_stump = sum(G),

 Biomass\_stump = sum(B),

 stems\_stump = sum(N),

 Volume\_stump = sum(V))

stump\_cover.3 = stump\_cover.2[!is.na(stump\_cover.2$Biomass\_stump),]

write.csv(stump\_cover.3, "stump\_stand\_results\_revised\_16.06.2022.csv")

##Combining tree and stump results

##tree data: stand\_trees\_all

stand\_trees\_all$plot\_id1 = stand\_trees\_all$location.plot\_id

tree\_data = stand\_trees\_all[,c("plot\_id1","Basal\_area", "Biomass", "stems", "Volume")]

##merging the stump and tree data

Stumps\_trees = merge(tree\_data, stump\_cover.3, by = c("plot\_id1"), all.x = TRUE)

##Export Results

export(Stumps\_trees, "stand\_parameters\_biomass\_volume\_number\_stems\_stump - 16.06.2022.xlsx")

###Species Richness

##Tree data = plot\_tree\_stem\_

names(plot\_tree\_stem\_)

## Updated from the correct names from the BIOMASS Package

##for trees, the data used is "plot\_tree\_1"

##to get the species encountered in a plot we have to remove the duplicates

plot\_tree.1$plot\_id1 = plot\_tree.1$location.plot\_id

rich.1 = unique(setDT(plot\_tree.1), by = c("location.plot\_id", "genusCorr", "speciesCorr"))

rich.2 = rich.1[, c("plot\_id1", "genusCorr", "speciesCorr")]

rich.2$k = 2

##richness to each plot###

richness = rich.1%>%

 group\_by(plot\_id1)%>%

 summarise(species\_richness = length(genusCorr))

export(richness, "Tree\_richness\_15.04.2022.xlsx")

#####Combining Tree and stump data: stump and tree data have to be concatenated######

##prepare stump data: "plot\_stump.2"

rich\_stump.1 = unique(setDT(plot\_stump.2), by = c("plot\_id1", "genusCorr", "speciesCorr"))

rich\_stump.1$k = 1

rich\_stump.2 = rich\_stump.1[, c("plot\_id1", "genusCorr", "speciesCorr", "k")]

##merge the stump and tree data()

all = rbind(rich.2, rich\_stump.2)

all.1 = unique(setDT(all), by = c("plot\_id1", "genusCorr", "speciesCorr"))

richness\_all = all.1%>%

 group\_by(plot\_id1)%>%

 summarise(species\_richness\_tree\_stumps = length(genusCorr))

export(richness\_all, "Tree\_stumps\_richness\_15.04.2022.xlsx")