methods and results

Luke Brogna, Jessie Xu, Mina Wei

2022-05-06

This vignette explains how we got the data, the methods we used in the project to analyze the planting activities and climate influence (precipitation) in Zambia croplands, and the results.

Introduction

This project is a study of the relationship between planting dates and rainfall. In this project, this team used qualitative analysis to hypothesize that the variation in the timing of when farmers plant their crops is correlated with rainfall variability and whether there is a huge difference in when farmers plant their crops because of rainfall. Comparing farmland in areas with similar long-term mean rainfall, we will analyze whether there is a relationship between the coefficient of variations of rainfall in the first two months of the growing season and the coefficient of variation of greening dates. To do this, the covariance and coefficient of correlation between the two variables will be calculated. We expect farmlands with similar rainfall to have similar green-up dates when farmers plant their crops on similar dates. We believe that the timing of rainfall has a positive effect on the timing of planting crops, and that farmers choose to plant crops near a specific rainfall time. The temporal relationship between rainfall and green-up date we predict should be normally distributed, with too much rainfall causing earlier green-up dates and too little rainfall causing a delay in green-up date. The greening dates of farmlands with the same characteristics (rainfall, rainfall time, sowing time, etc.) should be similar.

```
library(ZamPlantingRainfall)
library(raster)
#> Loading required package: sp
#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_FileGDB.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)

#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_FileGDB.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)
#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_MSSQLSpatial.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)

#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_MSSQLSpatial.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)
#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_OCI.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)

#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_OCI.dll
#> 126: The specified module could not be found.
```

```
#>  (GDAL error 1)
#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_PG.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)

#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_PG.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)
#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_FileGDB.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)

#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_FileGDB.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)
#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_MSSQLSpatial.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)

#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_MSSQLSpatial.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)
#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_OCI.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)

#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_OCI.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)
#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_PG.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)

#> Warning: Can't load requested DLL: D:\GeoDa\GeoDa Software\ogr_PG.dll
#> 126: The specified module could not be found.
#>  (GDAL error 1)
library(geojson)
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
         Software\ogr_FileGDB.dll
#> 126: The specified module could not be found.
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
         Software\ogr_FileGDB.dll
#> 126: The specified module could not be found.
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
         Software\ogr_MSSQLSpatial.dll
#> 126: The specified module could not be found.

#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
         Software\ogr_MSSQLSpatial.dll
#> 126: The specified module could not be found.
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
         Software\ogr_OCI.dll
#> 126: The specified module could not be found.
```

```
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_OCI.dll
#> 126: The specified module could not be found.
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_PG.dll
#> 126: The specified module could not be found.

#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_PG.dll
#> 126: The specified module could not be found.
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_FileGDB.dll
#> 126: The specified module could not be found.

#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_FileGDB.dll
#> 126: The specified module could not be found.
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_MSSQLSpatial.dll
#> 126: The specified module could not be found.

#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_MSSQLSpatial.dll
#> 126: The specified module could not be found.
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_OCI.dll
#> 126: The specified module could not be found.

#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_OCI.dll
#> 126: The specified module could not be found.
#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_PG.dll
#> 126: The specified module could not be found.

#> Warning in CPL_gdal_init(): GDAL Error 1: Can't load requested DLL: D:\GeoDa\GeoDa
       Software\ogr_PG.dll
#> 126: The specified module could not be found.
#>
#> Attaching package: 'geojson'
#> The following object is masked from 'package:graphics':
#>
#>     polygon
library(abind)
library(tidyverse)
#> -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
#> v ggplot2 3.3.6     v purrr   0.3.4
#> v tibble  3.1.6     v dplyr   1.0.9
#> v tidyr   1.2.0     v stringr 1.4.0
#> v readr   2.1.2     v forcats 0.5.1
#> -- Conflicts ------------------------------------------ tidyverse_conflicts() --
#> x tidyr::extract() masks raster::extract()
#> x dplyr::filter()  masks stats::filter()
#> x dplyr::lag()     masks stats::lag()
```

```
#> x dplyr::select()  masks raster::select()
library(sf)
#> Linking to GEOS 3.9.1, GDAL 3.2.1, PROJ 7.2.1; sf_use_s2() is TRUE
library(stars)
library(rgdal)
#> Please note that rgdal will be retired by the end of 2023,
#> plan transition to sf/stars/terra functions using GDAL and PROJ
#> at your earliest convenience.
#>
#> rgdal: version: 1.5-31, (SVN revision 1171)
#> Geospatial Data Abstraction Library extensions to R successfully loaded
#> Loaded GDAL runtime: GDAL 3.4.1, released 2021/12/27
#> Path to GDAL shared files: D:/R/R-4.1.3/library/rgdal/gdal
#> GDAL binary built with GEOS: TRUE
#> Loaded PROJ runtime: Rel. 7.2.1, January 1st, 2021, [PJ_VERSION: 721]
#> Path to PROJ shared files: D:\GeoDa\GeoDa Software\proj
#> PROJ CDN enabled: FALSE
#> Linking to sp version:1.4-6
#> To mute warnings of possible GDAL/OSR exportToProj4() degradation,
#> use options("rgdal_show_exportToProj4_warnings"="none") before loading sp or rgdal.
library(ggplot2)
library(dplyr)
library(tiff)
```

Data

Three main input datasets were used in this analysis, one for rainfall, one for green-up dates, and one which spatially identified the cropland landcover type. The scope of this analysis is an 18-year period from 2001 through 2018. When necessary, the data was filtered to this temporal extent.

Using the rgee package, data was pulled from the Google Earth Engine's data catalog. Because of long download times preprocessing of the data was done mostly with rgee Earth Engine code. The code which preprocesses and calculates the data from the online database was only run once, and the resulting images were saved locally for further analysis and presentation as .tif files.

```
library(rgee)
library(googledrive)
library(future)

# download data from google earth engine
# Initializing GEE, and downloading a Zambia mask for later use

# ee_clean_pyenv()

# ee_install()
# # Restart R session
#
# ee_check()

ee_Initialize(user = 'bluestarrystar2@gmail.com', drive = TRUE)  # google drive account
```

To aid in the spatial filtering of the data, the geometry of Zambia was extracted from a country-level administrative borders database. This geometry collection is only used to filter Earth Engine objects, so there was no need to save it locally.

```
#extracting Zambia geometry
zamGeo <- ee$FeatureCollection("FAO/GAUL_SIMPLIFIED_500m/2015/level0")$
  filter('ADM0_NAME == "Zambia"')$
  geometry()
```

Information about the spatial distribution of agriculture was obtained using the NASA-funded USGS Global Food Security-Support Analysis Data (GFSAD) extracted at 1000-meter resolution. The dataset consists of a single classified image for 2010 (produced using data from 2007-2012). The cropland cover types were isolated producing a mask to focus the analysis on agricultural areas. Below is the rgee code originally used to preprocess the data.

```
#cropland data mask
cropland <- ee$Image("USGS/GFSAD1000_V1")$
  select('landcover')$
  reduce(ee$Reducer$allNonZero())$
  clip(zamGeo)$
  rename('cropland')
```

The green-up date values used were precalculated by USGS and NASA in their 1-kilometer resolution MODIS Global Vegetation Phenology product, from which the Greenup_1 band was extracted. The data came in the form of annual images, with cell values representing green-up date as number of days since January 1, 1970. This was modified to reflect days since January 1 in the given image's collection year. These data were used to calculate the coefficient of variation of green-up date for each pixel. The rgee code which was used to preprocess, calculate, and eventually obtain these images for the rest of the analysis can be seen underneath this.

```
# We extracted the data via google earth engine, but this part of code doesn't work in R. We think
        this is because of a bug in rgee's handling of the $cast() function.
# preparing greenup date data to show number of days since the image year's start
greenup <- ee$ImageCollection("MODIS/006/MCD12Q2")$
  select('Greenup_1')$
  map(function (i) {
    dateCorrection = i$date()$advance(-1, 'year')$difference(ee$Date(0), 'day')
    return(i$subtract(dateCorrection)$
      mask(cropland)
      # $cast({'Greenup_1': ee$PixelType('float', -43725.0, 21810.0)}))
    )})

#coefficient of variation of greenup dates
greenupcv <- greenup$select('Greenup_1')$
  reduce(ee$Reducer$stdDev())$
  divide(greenup$reduce(ee$Reducer$mean()))$
  rename('greenup_cv')
```

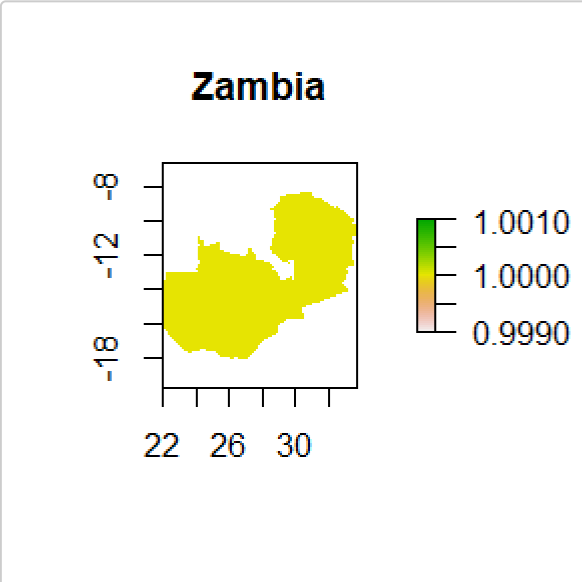
Gridded daily rainfall estimates given in millimeters per day were pulled from CHIRPS (Climate Hazards Group InfraRed Precipitation with Station Data) at a resolution of 5,566 meters. The CHIRPS data was manipulated to calculate both the long term mean rainfall, and the coefficient of variation of rainfall in the first two months of the growing season. Below is a typical single day CHIRPS image, along with the images containing the coefficient of variation for the first two months of the growing season, and the calculated long term mean rainfall, which was used to stratify data later in the analysis.

```
#chirps dataset
rainfall <-  ee$ImageCollection("UCSB-CHG/CHIRPS/DAILY")$
  select('precipitation')$
  filterDate("2001-01-01", "2018-12-31")$
  map(function (i) {
    return(i$mask(cropland))
  })

#mean rainfall of Zambian cropland
rfmean <- rainfall$mean()

#rainfall for first two months of growing season
growseasonrf <- rainfall$filter(ee$Filter$calendarRange(11, 12))

#coefficient of variation for rainfall in first two months of growing season
growseasonrfcv <- growseasonrf$reduce(ee$Reducer$stdDev())$
  divide(growseasonrf$reduce(ee$Reducer$mean()))$
  rename('rainfall_cv')
```

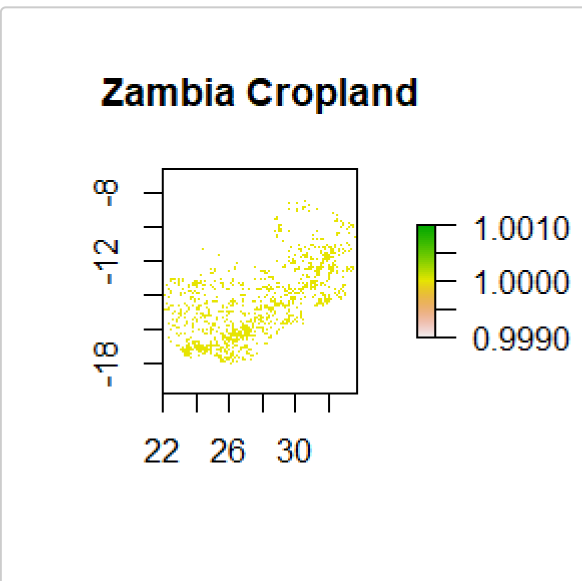
Visualizing maps based on data that are already downloaded to the local paths under the project.

```
zam <- system.file("extdata/zamMask.tif", package = "ZamPlantingRainfall")
zam <- raster(zam)
zam[zam == 0] <- NA

cropland <- system.file("extdata/cropmask.tif", package = "ZamPlantingRainfall")
cropland <- raster(cropland)
cropland[cropland == 0] <- NA

rfmean <- system.file("extdata/rfmean.tif", package = "ZamPlantingRainfall")
rfmean <- raster(rfmean)
rfmean[rfmean == NA] <- 0

rfcv <- system.file("extdata/rfcv.tif", package = "ZamPlantingRainfall")
rfcv <- raster(rfcv)
rfcv[rfcv == NA] <- 0

greenup_cv <- system.file("extdata/greenup_cv.tif", package = "ZamPlantingRainfall")
greenup_cv <- raster(greenup_cv)
greenup_cv[greenup_cv == NA] <- 0
```

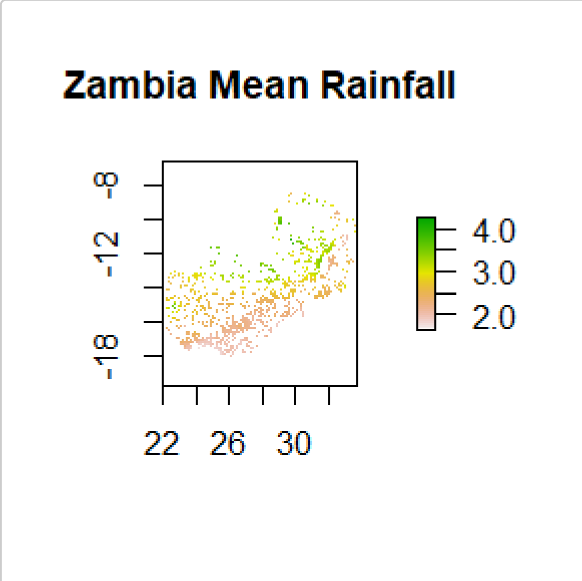
```
p1 <- plot(zam) + title("Zambia")
```



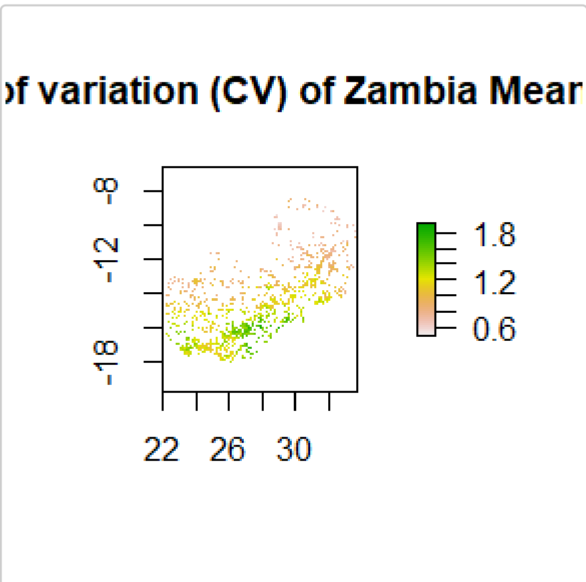

```
p2 <- plot(cropland) + title("Zambia Cropland")
```



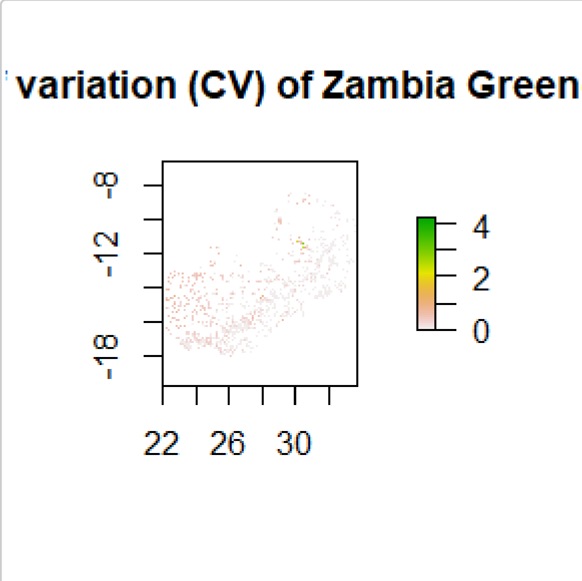

```
p3 <- plot(rfmean) + title("Zambia Mean Rainfall")
```




```
p4 <- plot(rfcv) + title("Coefficient of variation (CV) of Zambia Mean Rainfall")
```



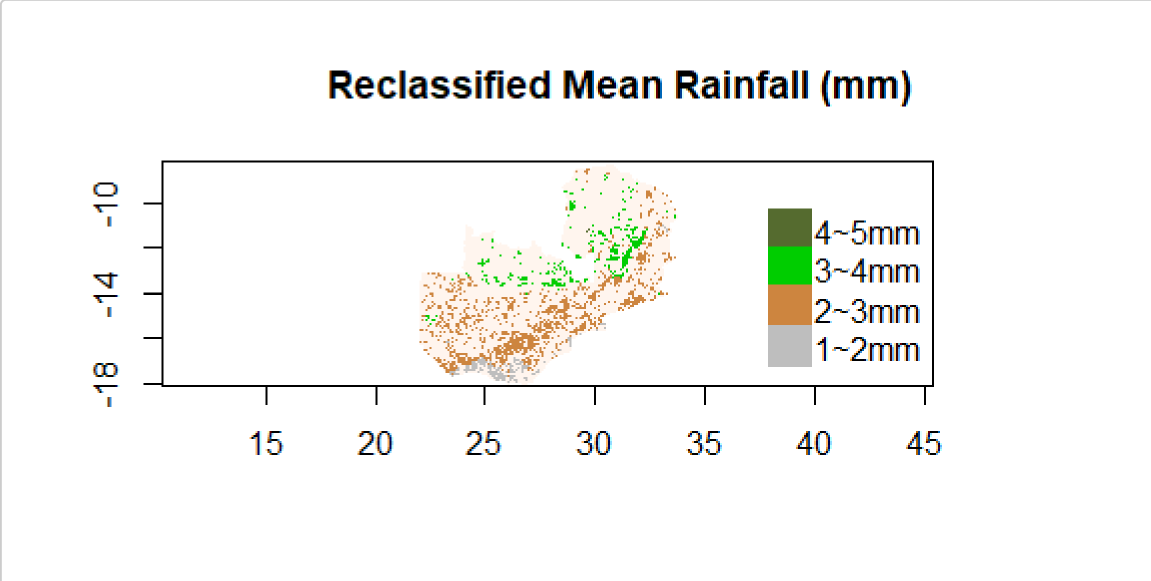

```
p5 <- plot(greenup_cv) + title("Coefficient of variation (CV) of Zambia Green-up Dates")
```



Visualize the reclassified mean rainfall areas.

The long term mean precipitation image was then reclassified from continuous data into four categories. Each category represents a bin of rainfall data (in mm per day). The total mean rainfall range from about 1.6 mm to 4.3 mm per day. Therefore, the mean rainfall values were broken into 4 classes: 1~2mm, 2~3mm, 3~4mm, and 4~5mm. This categorized image was used to stratify both previously calculated coefficients of variation. It is shown plotted below.

```
rfmeancut <- cut(x = rfmean, breaks = c(1, 2, 3, 4, 5), include.lowest = TRUE)

plot(zam, col = "seashell", legend = FALSE)
cols1 <- c("grey", "tan3", "green3", "darkolivegreen")
plot(rfmeancut, legend = FALSE, col = cols1, mar = c(0.5, 0.5, 1, 0), add = TRUE) +
        title("Reclassified Mean Rainfall (mm)")
#> integer(0)
legend(x = "bottomright", legend = c("4~5mm", "3~4mm", "2~3mm", "1~2mm"), pch = 15, pt.cex = 3, col =
        rev(cols1), bty = "n")
```



The next step was to evaluate the relationship between the coefficients of variation for green-up date and rainfall in the first two months of the growing season. To do this, the covariance and the correlation coefficient between the two variables was calculated within each stratum. Missing values in one or both of the images are ignored on a case by case basis (acheived by specifying the argument use = “complete.obs” within each cor() and cov() function). Together, these metrics show the strength and the direction of the linear relationship between the two variables.

```
#stratified values for rfcv image
rfcv1 <- rfcv[rfmeancut == 1]
rfcv2 <- rfcv[rfmeancut == 2]
rfcv3 <- rfcv[rfmeancut == 3]
rfcv4 <- rfcv[rfmeancut == 4]
rfcv5 <- rfcv[rfmeancut == 5]

#stratified values for greenupcv image
gucv1 <- greenup_cv[rfmeancut == 1]
gucv2 <- greenup_cv[rfmeancut == 2]
gucv3 <- greenup_cv[rfmeancut == 3]
gucv4 <- greenup_cv[rfmeancut == 4]
gucv5 <- greenup_cv[rfmeancut == 5]

#calculating correlation and covariance for each strata
cor1 <- cor(x = rfcv1, y = gucv1, method = 'pearson', use = "complete.obs")
cov1 <- cov(x = rfcv1, y = gucv1, method = 'pearson', use = "complete.obs")

cor2 <- cor(x = rfcv2, y = gucv2, method = 'pearson', use = "complete.obs")
cov2 <- cov(x = rfcv2, y = gucv2, method = 'pearson', use = "complete.obs")

cor3 <- cor(x = rfcv3, y = gucv3, method = 'pearson', use = "complete.obs")
cov3 <- cov(x = rfcv3, y = gucv3, method = 'pearson', use = "complete.obs")

cor4 <- cor(x = rfcv4, y = gucv4, method = 'pearson', use = "complete.obs")
cov4 <- cov(x = rfcv4, y = gucv4, method = 'pearson', use = "complete.obs")

cor1
```

```
#> [1] -0.1110667
cor2
#> [1] -0.09536359
cor3
#> [1] -0.1388347
cor4
#> [1] 0.1861437

cov1
#> [1] -0.002234126
cov2
#> [1] -0.004068045
cov3
#> [1] -0.006337246
cov4
#> [1] 0.007426003
```

Results

In general, the majority of croplands in Zambia tend to have a low level of precipitation at the scale of 2 to 4 mm per day. 1 to 2 mm is concentrated in the southern part of the map and accounts for a relatively small percentage of the map. The reason for this small percentage may be that the rainfall range starts from 1.6 mm to 2 mm rather than 1mm. 2-3 mm is the most widespread range on the map. It is concentrated in the south-central part of the map, with a few occurrences in the northern and eastern areas. 3-4 mm rainfall range is mainly concentrated in the northern croplands in Zambia. Although there are not many areas that have 3-4 mm average rainfall in Zambia, a high density of croplands could be observed in the western part of the map. Areas that have rainfall higher than 4mm are rarely observed. The phenomenon might be due to the highest value of total average rainfall being 4.3 mm per day, and areas with 4mm to 5mm daily rainfall are too small to be presented on a large-scale map. By inference, rainfall within 4 to 5 mm rainfall interval might be considered a very rare average rainfall in Zambia's croplands. Overall, the correlation coefficients between the two variables in all of Zambia's croplands are rounded to -0.1. The result presents a very low level of the adaptive response of farmers to climate change. Negative relationships were found in most of the areas within Zambia's croplands that have mean rainfall that is less than 4mm per day. A weak relationship indicated by a small value of correlation coefficient (0.186) only shows in a small area of croplands where the long-term mean rainfall is higher than 4mm per day. The positive relationship is hard to observe on a large scale map at the country level. The covariance shows the same pattern as the correlation coefficient.

Conclusion

At first, we expected to find a community-level action responding to climate change. Actually, the results match our hypothesis in some areas to some extent. The adaptive response for dealing with climatic uncertainty could be discovered only in a small area of croplands that have more remarkable climate features. In this case, a higher level of precipitation might lead to a change in agricultural activities. In these large areas of croplands, the change of precipitation in such a small amount is hard to be observed directly by farmers. Therefore, the negative correlation coefficients indicate that farmers are not likely to change their habits of planting in most croplands in Zambia when the rainfall is low. One issue that was encountered had to do with the rgee package. Although rgee worked flawlessly for almost everything that it was needed for, there seems to be some sort of small bug in how it handles GEE's cast() function when iterating over multiple layers within a map() function. The same code works in the actual GEE code editor, but not in R using rgee. The only reason that cast() had to be used in this manner was due to a quirk in the green-up data, where a few of the images contained different pixel types than the rest of the collection. Troubleshooting was not successful, so it was decided that the green-up data would be sent to google drive storage directly from the GEE code editor and placed into the project as a .tif. The code which would allow us to do this in R is still there, but it does not evaluate (along with the rest of the rgee code). Possibly, a future update of rgee might allow for the mapping of the cast() function over an image collection. There is also a possible issue of green-up dates extending into the subsequent year. Because the maximum green-up date value is 365, there may be low values (green-up dates in January that spilled over from the previous years growing season) in the data influencing the variation and linear relationship metrics. Researchers who plan to do a similar analysis to this one may wish to use a more selective cropland data mask, as green-up dates for crops become less accurate when forest and other landcover types are polluting the data.