Appendix 2. Fitting Models and Generating Carbon Stock Maps

1. Introduction

This appendix is a quick tutorial on how to apply the method used in our study to improve carbon stock estimates and generate carbon stock maps. Here, we show how to fit the Random Forest (RF) and Generalized Additive Model (GAM), and how to use these models to make predictions on a grid cell-level for the desired area.

For our methodology to work, users only need geolocated field plots with their centroid's spatial coordinates and carbon stock density estimates, and their correspondent large-scale map information. For this tutorial, we are using the canopy height information from the global map published by Lang et al. [1]. More information about the global map and how to download it can be found at https://langnico.github.io/globalcanopyheight/. Note that, as the spatial resolution of the large-scale map is smaller than the field plot dimensions, we aggregated the grid-cells to a 35-meter resolution to match the field plot size.

2. Simulated Data

For this tutorial, we use a small set of simulated data that can be accessed using the code below.

```
#install.packages('readr')
library(readr)

#Loading Simulated Data
urlfile="https://raw.githubusercontent.com/LeoHaneda/Haneda_etal_2024_Simple_methods/main/sim_data"
dat <- data.frame(readr::read_csv(url(urlfile)))
head(dat)
```

```
    CH      Lat       Lon    CSD
1 35.0 25.26136 -18.36433 671.42
2 19.5 25.27866 -18.36527  67.45
3 12.0 25.28023 -18.37659  34.11
4 15.8 25.25413 -18.35270  63.87
5 18.5 25.27237 -18.34829  84.34
6 26.0 25.25571 -18.39325 449.46
```

where:

- CH: Canopy height from large-scale map [m];
- Lat and Lon: Spatial coordinates of the field plots centroid (Latitude and Longitude);
- CSD: Carbon stock density from field plots [$tCO_2e.ha^{-1}$];

3. Fitting Models

After having all the data structured, fitting the models is a straightforward step.

3.1. Random Forest (RF)

To fit the RF model, we rely on the `randomForest` R package [2], using the standard formula syntax to inform the response and explanatory variable.

```
#install.packages('randomForest')
library(randomForest)
RF_mod <- randomForest::randomForest(CSD ~ CH + Lat + Lon,
                            data = dat)
```

3.2. Generalized Additive Model (GAM)

To fit the GAM, we rely on the `mcgv` R package [3], using the standard formula syntax to inform the response and spline functions with the exploratory variables.

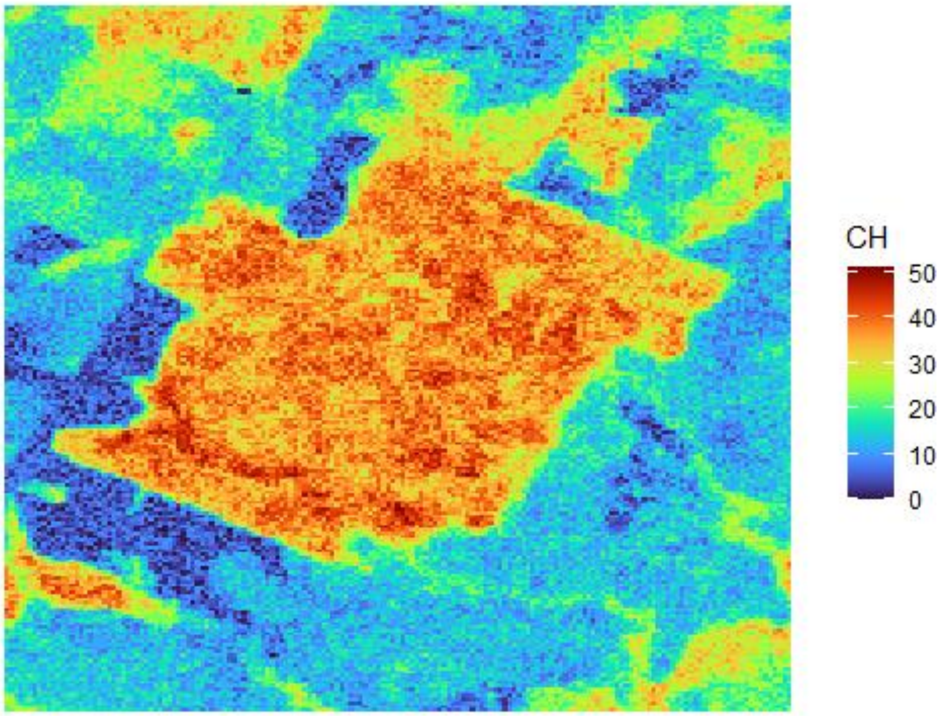
```
#install.packages('mgcv')
library(mgcv)
GAM_mod <- mgcv::gam(CSD ~ s(CH) + s(Lat, Lon, bs="tp"),
               data = dat, method = "REML")
```

4. Canopy Height Map for Prediction

After fitting the model, we use it to make predictions for each grid-cell inside the REDD+ project area. First, we need to extract the large-scale map information and the spatial coordinates (Latitude and Longitude) for each grid-cell.

```
#install.packages('terra')
#install.packages('ggplot2')
#install.packages('viridis')
library(terra)
library(ggplot2)
library(viridis)

#Load Simulated Canopy Height Map
download.file("https://github.com/LeoHaneda/Haneda_etal_2024_Simple_methods/b
lob/main/sim_rast.zip?raw=TRUE",
  "sim_rast.zip", mode = "wb")
Map <- terra::rast(unzip("./sim_rast.zip"))

#Extract the values and spatial coordinates of each grid-cell from the large-
scale map
df <- terra::as.data.frame(Map,xy = TRUE, na.rm = FALSE)
names(df) <- c("Lat", "Lon", "CH")

#Plot Canopy Height Map
ggplot(df)+
    geom_tile(aes(x = Lat, y = Lon, fill = CH)) +
```

```
    scale_fill_viridis(option = "H") +
    theme_void() +
    theme(legend.position = "right")
```




```
head(df)

       Lat       Lon   CH
1 25.23307 -18.34546 12.2
2 25.23338 -18.34546 13.7
3 25.23370 -18.34546 23.9
4 25.23401 -18.34546 21.8
5 25.23433 -18.34546 28.8
6 25.23464 -18.34546 24.5
```

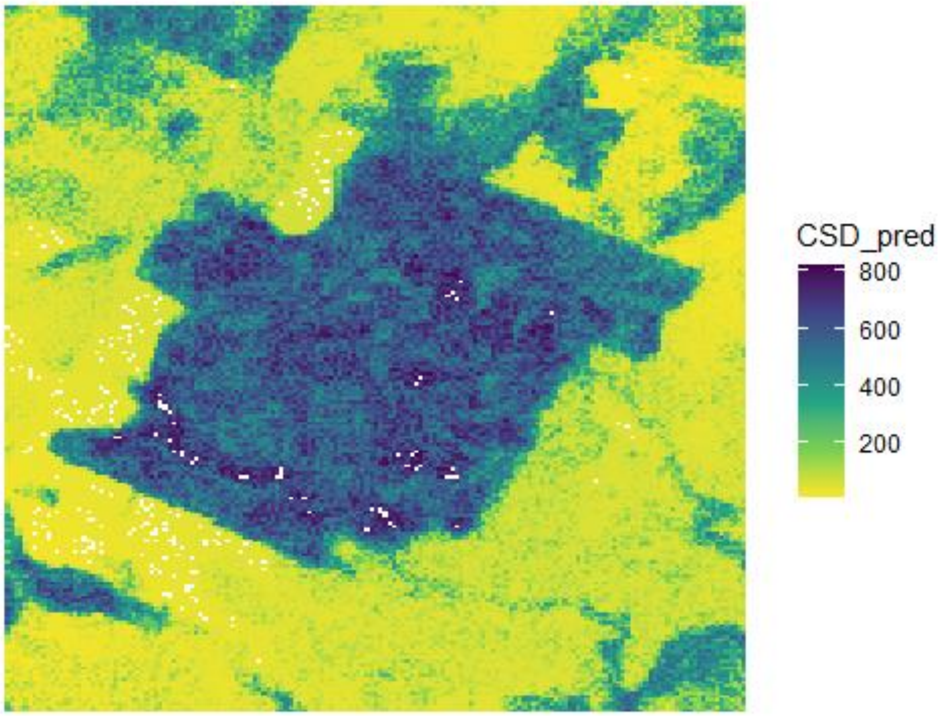
5. Carbon Stock Maps

Then, we make the carbon stock prediction for each grid-cell and rasterize them using their spatial coordinates. To avoid extrapolation, we remove all grid-cell that are outside the range of values used in the training dataset, which are represented on the map as empty pixels.

```
#install.packages('terra')
#install.packages('ggplot2')
#install.packages('viridis')
library(terra)
library(ggplot2)
library(viridis)
```

```r
#Removing grid-cells outside the range of the trainnig dataset to avoid
extrapolation
range.lat=range(dat$Lat)
cond.lat=df$Lat < range.lat[1] | df$Lat > range.lat[2]
df$Lat[cond.lat] <- NA

range.lon=range(dat$Lon)
cond.lon=df$Lon < range.lon[1] | df$Lon > range.lon[2]
df$Lon[cond.lon] <- NA

range.ch=range(dat$CH)
cond.ch=df$CH < range.ch[1] | df$CH > range.ch[2]
df$CH[cond.ch] <- NA

df <- na.omit(df)

#Make predictions using the Random Forest model
pred_RF <- predict(RF_mod, df)

#Make predictions using the Generalized Additive Model
pred_GAM <- predict(GAM_mod, df)

#Matching the grid-cell predictions with respective spatial coordinates
df_pred <- data.frame(X = df$Lat,
                      Y = df$Lon,
                      CSD_pred = pred_GAM)

head(df_pred)

           X         Y  CSD_pred
509 25.23338 -18.34641 373.92732
510 25.23370 -18.34641 374.21700
511 25.23401 -18.34641 360.77943
512 25.23433 -18.34641 204.18496
513 25.23464 -18.34641 134.75723
514 25.23495 -18.34641  50.75225

#Raterize
CSD_pred_map <- terra::rast(df_pred, type = "xyz")

#Setting Coordinate System
terra::crs(CSD_pred_map) <- terra::crs('+init=EPSG:4326') #WGS 84

#Save raster
writeRaster(CSD_pred_map, "./CSD_map.tif", overwrite = T)

#Plotting Carbon Stock Map
ggplot(df_pred)+
```

```
    geom_raster(aes(x = X, y = Y, fill = CSD_pred))+
    scale_fill_viridis_c(direction = -1) +
    theme_void() +
    theme(legend.position = "right")
```



Reference

1. Lang, N., Jetz, W., Schindler, K. & Wegner, J. D. A high-resolution canopy height model of the Earth. *Nat. Ecol. Evol.* **7**, 1778–1789 (2023).

2. Liaw, A. & Wiener, M. Classification and Regression by randomForest. *R News* **Vol. 2/3**, (2002).

3. Wood, S. N. Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models. *J. Royal Statistical Soc. B* **73**, 3–36 (2011).