# Crop rotation from Crop data layer

Sudha

2023-05-19

```
library("CropScapeR")
```

```
## Warning: package 'CropScapeR' was built under R version 4.2.3
```

```
##
## Recommendated citation for the 'CropScapeR' package is:
## Bowen Chen (2020). CropScapeR: Access Cropland Data Layer Data via the 'CropScape' Web Service. R pa
##
## For more information, visit the package website at: https://github.com/cbw1243/CropScapeR. If you ha
```

```
df = read.csv('C:/Users/theox/Desktop/Summer Projects/SoilFingerPrints/Data/SamplingLocations/SamplingL
         fileEncoding="latin1")
df = df[c('ID', 'lat', 'long', 'FieldType', 'Location','SamplingYear')]
head(df)
```

```
##   ID      lat      long  FieldType       Location SamplingYear
## 1  1 46.28371 -118.6746 Non-virgin Columbia Basin         2021
## 2  2 46.28129 -118.6848     Virgin Columbia Basin         2021
## 3  4 47.08460 -119.8083 Non-virgin Columbia Basin         2021
## 4  5 47.07672 -119.8030     Virgin Columbia Basin         2021
## 5  6 47.04543 -119.6095     Virgin Columbia Basin         2021
## 6  7 47.03991 -119.6040 Non-virgin Columbia Basin         2021
```

```
df$FieldType = gsub('Field', "", df$FieldType)
df$FieldType = gsub('field', "", df$FieldType)
df$FieldType = gsub('Non-virgin ', "Non-virgin", df$FieldType)
df$FieldType = gsub('Virgin ', "Virgin", df$FieldType)
unique(df$FieldType)
```

```
## [1] "Non-virgin" "Virgin"     "Native"
```

```
# define a function that returns the crop data for listed years for a given pin
crop_data = function(long, lat, years){
  crop_year = c() #empty list
  for(i in years){
    d = GetCDLData(aoi = c(long, lat), year = i, type = 'p', crs = '+init=epsg:4326') #use CropScapeR t
    crop_year = data.frame(cbind(crop_year,d$category)) # append
  }
  names(crop_year) = years # rename the columns
```

```
return(crop_year)}

# list of dates that we need data from
years = c('2016','2017','2018','2019','2020','2021','2022')

#test test
crop_data(-118.6746, 46.28371, years)
```

```
## Warning in CPL_crs_from_input(x): GDAL Message 1: +init=epsg:XXXX syntax is
## deprecated. It might return a CRS with a non-EPSG compliant axis order.
```

```
##           2016 2017     2018         2019       2020     2021         2022
## 1 Winter Wheat Corn Potatoes Winter Wheat Sweet Corn Potatoes Winter Wheat
```

```
# list of dates that we need data from
years = c('2016','2017','2018','2019','2020','2021','2022')
data_all = c()


# extract data for all fields
for (j in 1:length(df$long)){

  data = crop_data(long = df$long[j], lat = df$lat[j], years = years)
  data_ij = cbind(df[j,], data)
  data_all = rbind(data_all, data_ij)
}

dim(data_all)
```

```
## [1] 76 13
```

```
head(data_all)
```

```
##   ID      lat      long  FieldType       Location SamplingYear         2016
## 1  1 46.28371 -118.6746 Non-virgin Columbia Basin         2021 Winter Wheat
## 2  2 46.28129 -118.6848     Virgin Columbia Basin         2021    Shrubland
## 3  4 47.08460 -119.8083 Non-virgin Columbia Basin         2021     Potatoes
## 4  5 47.07672 -119.8030     Virgin Columbia Basin         2021    Shrubland
## 5  6 47.04543 -119.6095     Virgin Columbia Basin         2021    Shrubland
## 6  7 47.03991 -119.6040 Non-virgin Columbia Basin         2021    Shrubland
##        2017          2018         2019       2020                2021
## 1      Corn      Potatoes Winter Wheat Sweet Corn             Potatoes
## 2 Shrubland     Shrubland    Shrubland  Shrubland             Potatoes
## 3  Potatoes Grass/Pasture       Canola Sweet Corn             Potatoes
## 4 Shrubland     Shrubland    Shrubland  Shrubland Fallow/Idle Cropland
## 5 Shrubland     Shrubland    Shrubland  Shrubland            Shrubland
## 6 Shrubland     Shrubland   Sweet Corn   Potatoes            Shrubland
##           2022
## 1 Winter Wheat
## 2 Winter Wheat
## 3 Winter Wheat
```

```
## 4    Shrubland
## 5    Shrubland
## 6    Dry Beans
```

```
write.csv(data_all, "crop_rotation.csv")
```