|  |
| --- |
| title: “FIA example in R” |
| author: “Jeff Atkins” |
| date: ’ r format(Sys.time(), "%B %d, %Y) |
| output: |
| word\_document: default |
| pdf\_document: default |
| html\_document: default |

### Installation of rFIA

# install the package if not already installed and call it via library()  
if(!require(rFIA)){install.packages("rFIA")}

## Loading required package: rFIA

## Warning: package 'rFIA' was built under R version 4.0.5

library(rFIA)  
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.0.5

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.3 v purrr 0.3.4  
## v tibble 3.1.1 v dplyr 1.0.6  
## v tidyr 1.1.3 v stringr 1.4.0  
## v readr 1.4.0 v forcats 0.5.1

## Warning: package 'ggplot2' was built under R version 4.0.5

## Warning: package 'tibble' was built under R version 4.0.5

## Warning: package 'tidyr' was built under R version 4.0.5

## -- Conflicts ------------------------------------------ tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(sf)

## Warning: package 'sf' was built under R version 4.0.5

## Linking to GEOS 3.9.0, GDAL 3.2.1, PROJ 7.2.1

## Getting started with rFIA

Please visit <https://rfia.netlify.app/tutorial/downloading/> for a full run down on how to do this. Here, we work through their example using FIA data from the state of Virginia.

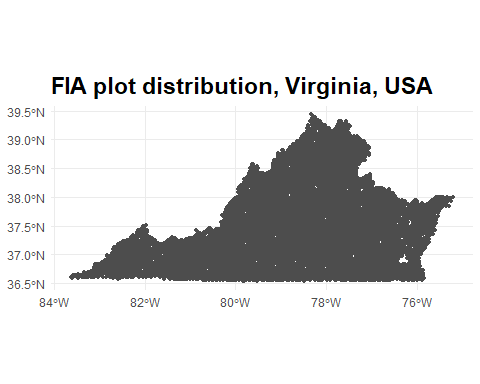
You can also embed plots, for example:

## Download the state subset or Connecticut (requires an internet connection)  
## Save as an object to automatically load the data into your current R session!  
getFIA(states = 'VA', dir = './data/fia/')  
  
## Get the forest type and forest type group reference tables  
ref <- getFIA(states = 'ref', tables = c('FOREST\_TYPE', 'FOREST\_TYPE\_GROUP'))

## Warning: The `.dots` argument of `group\_by()` is deprecated as of dplyr 1.0.0.

Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.

## Check spatial coverage of plots held in the database  
plotFIA(va, plot.title = "FIA plot distribution, Virginia, USA" )



## let's make a bounding box  
snp <- st\_read("./data/fia/SNP\_boundary/shen\_PkBndryPly.shp")

## Reading layer `shen\_PkBndryPly' from data source `C:\R\rforestanalysis\data\fia\SNP\_boundary\shen\_PkBndryPly.shp' using driver `ESRI Shapefile'  
## Simple feature collection with 4 features and 6 fields  
## Geometry type: POLYGON  
## Dimension: XY  
## Bounding box: xmin: 687853.1 ymin: 4211705 xmax: 747375.5 ymax: 4310539  
## Projected CRS: NAD83 / UTM zone 17N

## Most Recent Subset (2017)  
snp.clip <- clipFIA(va, mask = snp)   
  
  
# let's look at some basic population estimates  
tpaSNP <- tpa(snp.clip)

# we can also analyze by plot  
## Plot-level  
snp\_plot <- tpa(snp.clip, byPlot = TRUE)  
  
## or by tree  
## Plot-level  
snp\_plot\_tree <- tpa(snp.clip, byPlot = TRUE, grpBy = TREE)  
  
## Size class  
## Group estimates by size class  
snp\_sizeClass <- tpa(snp.clip, bySizeClass = TRUE)  
  
## Using our estimates from above (all inventory years in RI)  
plotFIA(snp\_sizeClass, y = BAA, x = sizeClass, plot.title = 'Size class distribution of trees in SNP')

