The plantspec R package: a tool for spectral analysis of plant stoichiometry

Daniel M. Griffith & T. Michael Anderson April 12, 2018

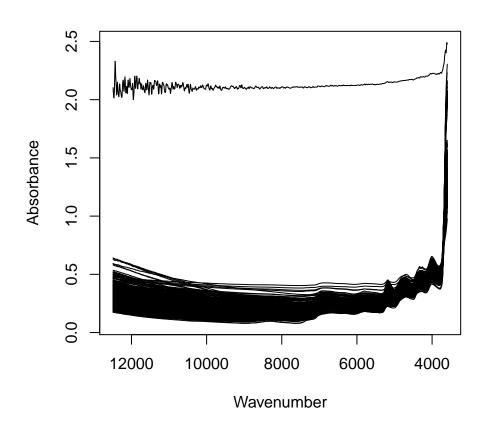
Installation

Install r3PG from GitHub using devtools.

```
install.packages("devtools", repos = "http://cran.us.r-project.org")
library(devtools)
install_github(repo = "griffithdan/plantspec")
library(plantspec)
install_github(repo = "griffithdan/plantspecDB") # data package
library(plantspecDB)
```

Load example dataset

```
# STARTING RESULTS
# BEGIN
# N: Code to create this calibration
# LOAD DATA FROM leaf.spec.DB
        data(leaf.spec.spectra) # CHANGE NAMES
        data(leaf.spec.data)
        leaf.spec.data <- leaf.spec.data[leaf.spec.data$VERSION==1,] # use dataset version 1
            table(leaf.spec.data$FUNCTIONAL_TYPE)
            table(leaf.spec.data$YEAR)
            table(leaf.spec.data$SITE_NAME)
            table(leaf.spec.data$SITE_COUNTRY)
        # ADD DISCUSSION OF SUBSETTING
        leaf.spec.data <- leaf.spec.data[!is.na(leaf.spec.data$N),] # remove data where response is NA
        leaf.spec.data <- subset(leaf.spec.data,!(FUNCTIONAL_TYPE=="bryophyte"))</pre>
        leaf.spec.data <- subset(leaf.spec.data,!(FUNCTIONAL_TYPE=="litter"))</pre>
        \#leaf.spec.data \leftarrow leaf.spec.data[!(leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data\$FUNCTIONAL\_TYPE=="bryophyte"|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.data$|leaf.spec.dat
        #ADD PLOT BEFORE HERE THAT IDs this spec
leaf.spec.data <- subset(leaf.spec.data, SCAN_FILE != "SCAN_00225.dpt ")</pre>
        NUTNET_SCANS <- leaf.spec.spectra[leaf.spec.data$SCAN_FILE,] # choose scans based on remaining data
        plot(NUTNET_SCANS) # inspect scans #ADD BOKEH PLOTTING
```



```
#component_N <- leaf.spec.data$N # for clarity, put the response in a new vector
#training_set_MDKS <- !(subdivideDataset(spectra = NUTNET_SCANS, component = component_N, method =
#N_opt <- optimizePLS(component=component_N, spectra=NUTNET_SCANS, training_set = training_set_MDKS
#N_cal <- calibrate(component=component_N, spectra=NUTNET_SCANS, optimal_params=N_opt, optimal_mode

#data(N_cal_23SEP2015)

#N_cal$R2_Val
#plot(N_cal)</pre>
```

#predict(N_cal, newdata = leaf.spec.spectra)