Spectrolab

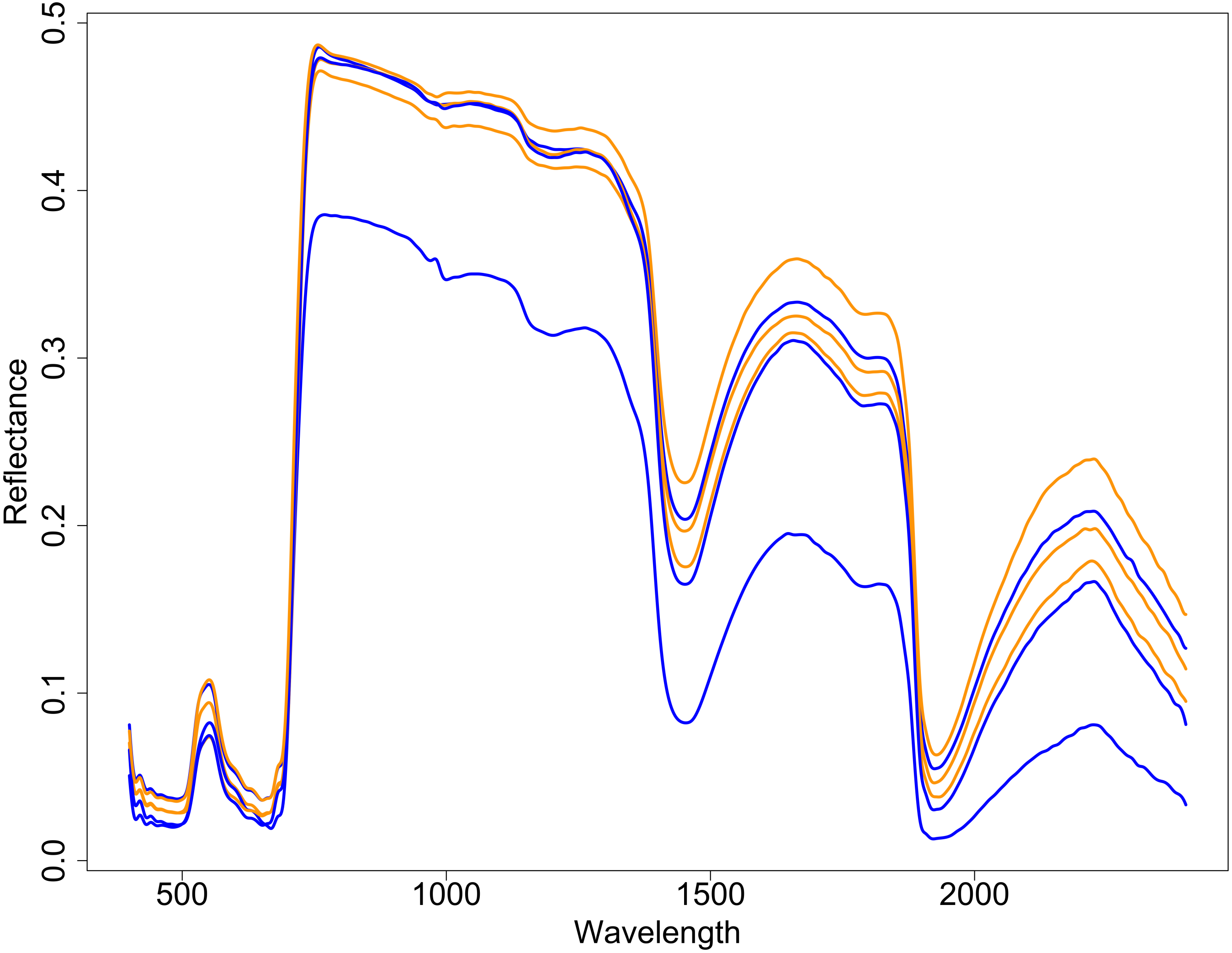
hyperspectral data wrangling with R

Jose Eduardo Meireles

jose.meireles@maine.edu

Slides and data are available on github:

https://github.com/meireles/learn_spectrolab


Reflectance
0.0
0.1
0.2
0.3
0.4
0.5
500
1000
1500
2000
Wavelength

Outline

1. Getting Started
2. Reading & Inspecting spectra
3. Visualizing spectra
4. Manipulating / Processing spectra
5. Exporting spectra
6. **Predicting leaf chemistry from spectra**

1. Getting Started

To start using spectrolab, you should:

1. Install it from either:

CRAN (stable):

```
install.packages("spectrolab")
```

GitHub (cutting edge):

```
library("devtools")
devtools::install_github("meireles/spectrolab")
```

2. Load spectrolab into R:

```
library("spectrolab")
```

2. Reading & Inspecting spectra

Read spectra from *instrument files*

```
## Read the PSR spectra from a directory
psr_spec = read_spectra(path = "../data/psr/", format = "sed")
```

This results in an object of class **spectra**

```
## Inspect spectra
psr_spec
```

```
## spectra object
## number of samples: 6
## wavelengths: 350 to 2500 (2151 bands)
## metadata: none
```

```
## Find out the name of the two first individual spectra
names(psr_spec)[1:2]
```

```
## [1] "ACSA_T_C_SH_20160630_00005.sed" "ACSA_T_C_SH_20160630_00007.sed"
```

Read spectra from a *matrix*

```
## Read CSV
csv = read.csv("../data/csv/spec_mat.csv", check.names = FALSE, as.

## Look at the matrix
csv[1:3, 1:5]
```

```
##        sp chlorophyll        400        401        402
## 1 Q. alba    18.28542 0.04117835 0.04124286 0.04129801
## 2 Q. alba    20.99738 0.04101308 0.04105510 0.04108458
## 3 Q. alba    16.70608 0.04122651 0.04129906 0.04136357
```

```
## Convert the matrix to spectra
csv_spec = as.spectra(x = csv, name_idx = 1, meta_idxs = 2)

## and look at the spectra
csv_spec
```

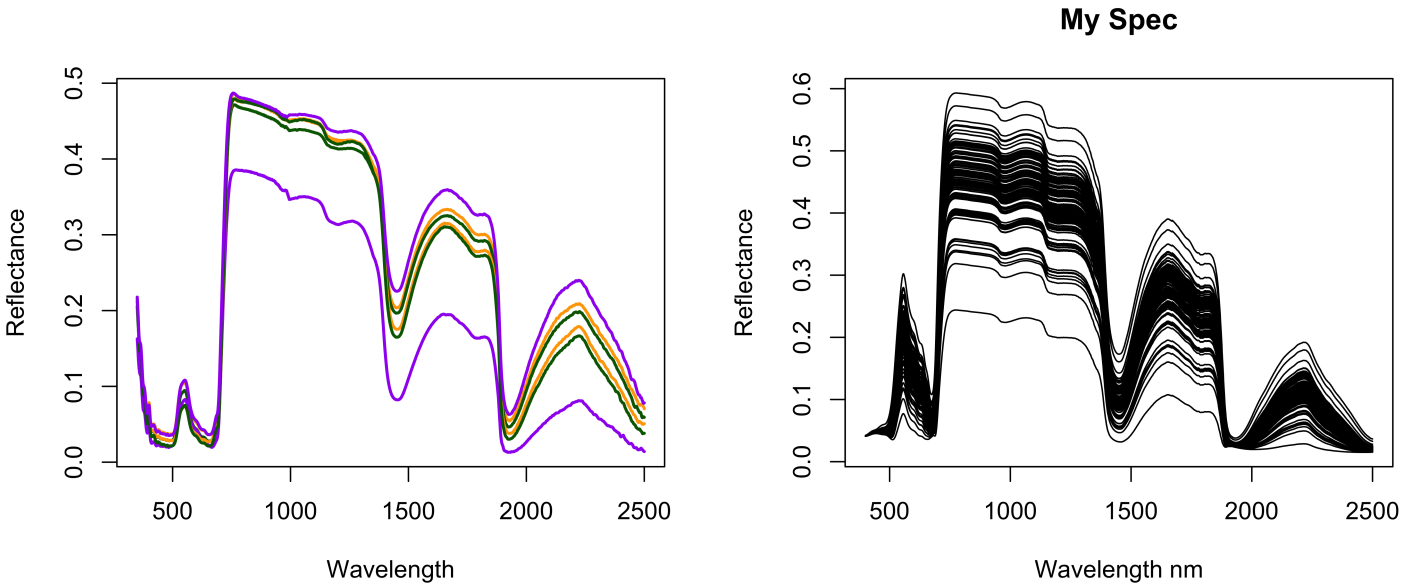
```
## spectra object
## number of samples: 100
## wavelengths: 400 to 2500 (2101 bands)
## metadata (1): chlorophyll
```

3. Visualizing spectra

Plot spectra

Statically with *plot*

```
par(mfrow = c(1, 2))
plot(psr_spec, lwd = 2, col = c("orange", "darkgreen", "purple"))
plot(csv_spec, xlab = "Wavelength nm", main = "My Spec")
```



Plot spectra

Interactively with *plot_interactive*

```
# Demo
plot_interactive(psr_spec)
```

4. Manipulating / Processing spectra

Manipulating spectra: e.g. subset

```
## Subset spectra like if they were matrices
psr_spec_trim = psr_spec[1:2 , 400:700]
psr_spec_trim
```

```
## spectra object
## number of samples: 2
## wavelengths: 400 to 700 (301 bands)
## metadata: none
```

```
## Except that you **MUST** subset bands by their wavelength, **not
psr_spec[ , 1:10]
```

```
## Error in i_match_label(wavelengths(x), j): Following labels not found: 12
```

Processing spectra: e.g. smooth

```
## Smooth spectra with a moving average
psr_spec_smooth = smooth(psr_spec,
                         method = "moving_average", n = 20)

## Plot them
par(mfrow = c(1, 2), mar = c(7, 5, 1, 1))
plot(psr_spec[1, 360:700])
plot(psr_spec_smooth[1, 360:700])
```

5. Exporting spectra

Convert spectra into other objects

Convert spectra into **matrix** or **data.frame** in order to export or use them in regressions, ordinations, etc.

```r
## Matrix
my_data_as_matrix = as.matrix(psr_spec)

## Data Frame
my_data_as_df     = as.data.frame(psr_spec)

## Quick peek
my_data_as_df[1:3, 1:4]
```

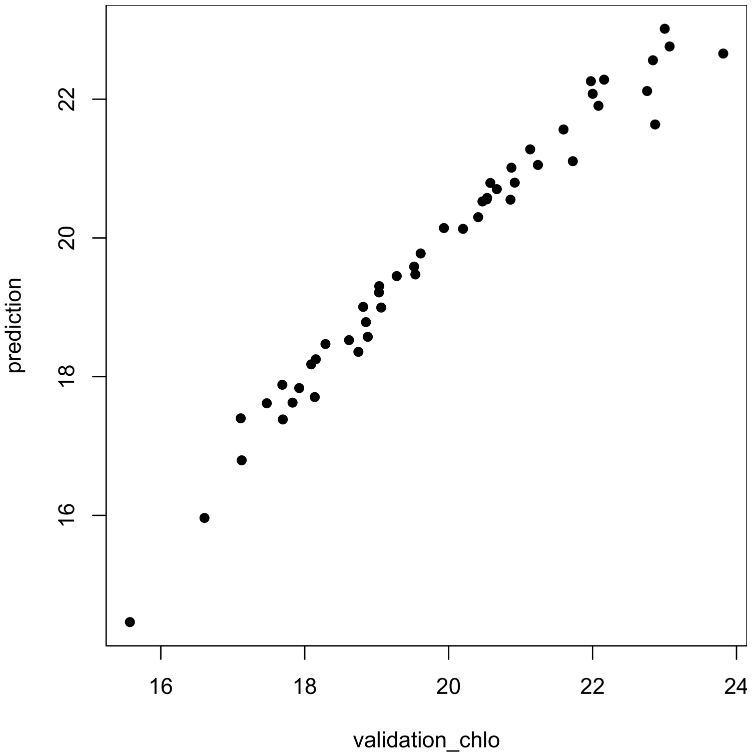
```
##                       sample_name      350      351      352
## 1 ACSA_T_C_SH_20160630_00005.sed 0.216800 0.210887 0.204364
## 2 ACSA_T_C_SH_20160630_00007.sed 0.205787 0.199252 0.192517
## 3 ACSA_T_C_SU_20160630_00003.sed 0.162666 0.158088 0.153185
```

6. Predicting leaf chemistry from spectra

Predicting leaf chlorophyll from spectra

```r
## Load the pls library
library("pls")

## Convert the spectra with the chloro metadata into a data frame
dat   = as.data.frame(csv_spec)

## Remove useless columns and fix chloro class (factor to numeric)
dat["sample_name"] = NULL
dat["chlorophyll"] = as.numeric(as.character(dat[ , "chlorophyll"])

## Divide data into "training" and "validation"
training_data   = dat[1:50,    ]
validation_spec = dat[51:100, -1]
validation_chlo = dat[51:100, 1 ]
```

```
## Fit the model
fit = plsr(chlorophyll ~ ., data = training_data, ncomp = 5)

## Predict chlorophyll from spectra!
prediction = predict(object = fit, newdata = validation_spec, ncomp

## Plot truth against predictions
par(mar = c(6,6,0,0)); plot(validation_chlo, prediction, pch = 16)
```


prediction
validation_chlo