

# CWRU DSCI351-351M-453: Week10a-HyperSpec

## Package Review

*Devin Gordon, Roger French*

*01 November, 2018*

## Contents

10.1.0.1	Background . . . . .	1
10.1.0.1.1	Spectra - What are they? . . . . .	1
10.1.0.1.2	Spectral Data - How do we store them? . . . . .	2
10.1.0.1.3	hyperSpec - How to manipulate .spc files in R . . . . .	2
10.1.0.2	hyperSpec Examples . . . . .	2
10.1.0.2.1	File Import . . . . .	3
10.1.0.2.2	Basic Plotting . . . . .	3
10.1.0.2.3	Multiple .spc Files? . . . . .	4
10.1.0.2.4	Combine into one object (Good News!) . . . . .	4
10.1.0.2.5	Basic Plotting . . . . .	4
10.1.0.2.6	Spectral Range Selection . . . . .	5
10.1.0.2.7	Basic Plotting . . . . .	5
10.1.0.2.8	Spectral Arithmetic (Subtraction) . . . . .	6
10.1.0.2.9	Removing Bad Data . . . . .	6
10.1.0.2.10	Removing Bad Data . . . . .	7
10.1.0.2.11	Removed the bad spectra . . . . .	7
10.1.0.2.12	cq.abs.step0 Correction (One of many options) . . . . .	8
10.1.0.2.13	Corrected Spectra! . . . . .	8
10.1.0.3	Example of use, Key Files for sample Metadata . . . . .	9
10.1.0.3.1	Load Cyclic QUV Absorbance Data Set . . . . .	9
10.1.0.4	Utilizing spectral arithmetic to change ordinate axis . . . . .	12
10.1.0.4.1	Better plotting using ggplot2 graphics . . . . .	13
10.1.0.5	Peak Information extraction . . . . .	13
10.1.0.6	Other plots . . . . .	16
10.1.0.6.1	Matrix Plot - Fluorescence Data . . . . .	16
10.1.0.6.2	Matrix Plot - IR Data . . . . .	16
10.1.0.6.3	Stacked Spectra . . . . .	17
10.1.0.7	Summary . . . . .	17
10.1.0.7.1	hyperSpec . . . . .	17

### 10.1.0.1 Background

#### 10.1.0.1.1 Spectra - What are they?

- “a specific set of values that can vary infinitely within a continuum”
- Generally: 2-dimensional (sometimes higher) dimensional data,
  - where the first dimension identifies points along a continuum,
  - and the second dimension the values of a response of interest
  - at the corresponding points
- Examples: Optical, Mass, Political Alignment

#### 10.1.0.1.2 Spectral Data - How do we store them?

- Proprietary file format: not so useful
- .csv files (ASCII): better ~ excel, R
- .spc files (Binary): good ~ R

#### 10.1.0.1.3 hyperSpec - How to manipulate .spc files in R

- <http://hyperspec.r-forge.r-project.org/>
- Handles spectral data
- Features:
  - Convenient import,
  - spectral range selection,
  - shfitting,
  - plotting/viewing,
  - normalization,
  - smoothing,
  - correction,
  - arithmetic,
  - initial data analysis

#### 10.1.0.2 hyperSpec Examples

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

## Loading required package: lattice

## Loading required package: grid

## Loading required package: ggplot2

## Package hyperSpec, version 0.99-20180627
##
## To get started, try
##   vignette("hyperspec")
##   package?hyperSpec
##   vignette(package = "hyperSpec")
##
## If you use this package please cite it appropriately.
##   citation("hyperSpec")
## will give you the correct reference.
##
## The project homepage is http://hyperspec.r-forge.r-project.org
##
## Attaching package: 'hyperSpec'
```

```
## The following object is masked from 'package:dplyr':  
##  
## collapse
```

#### 10.1.0.2.1 File Import

```
# Read one .spc file and check its structure  
# setwd("V:/vuv-data/instr/spectra/")  
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_00_ex320.spc"  
step0 <- read.spc(curfile)  
class(step0)
```

```
## [1] "hyperSpec"  
## attr(,"package")  
## [1] "hyperSpec"
```

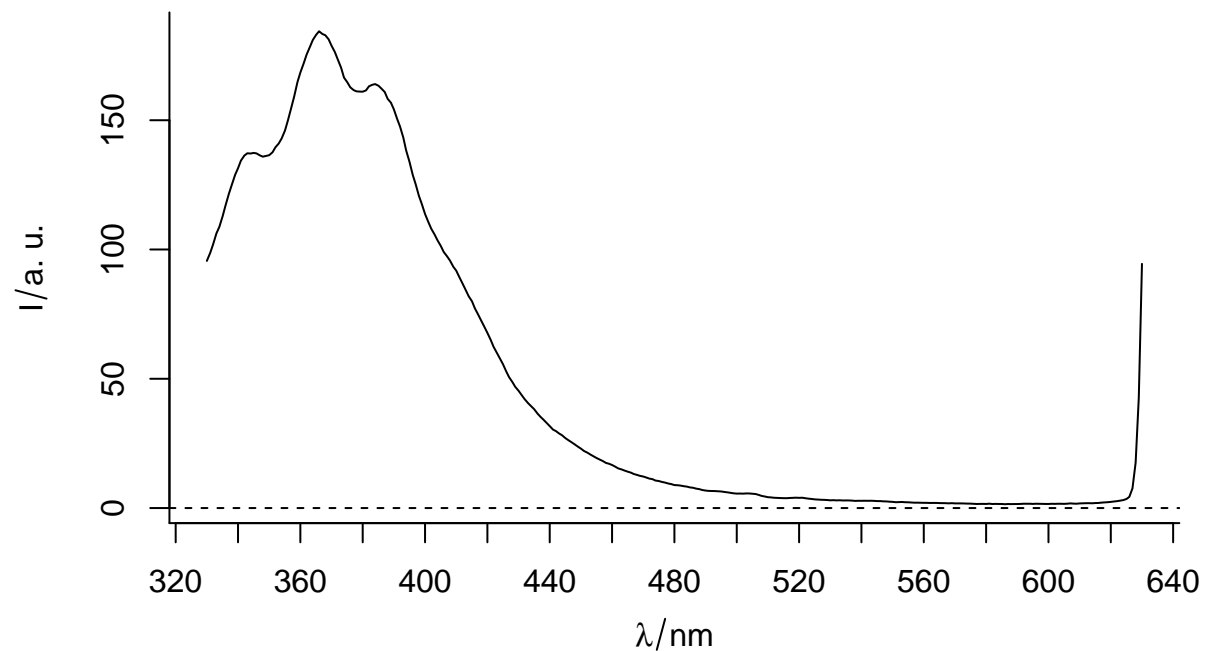
```
step0
```

```
## hyperSpec object  
## 1 spectra  
## 4 data columns  
## 301 data points / spectrum  
## wavelength: lambda/nm [numeric] 330.00 331.07 ... 630  
## data: (1 rows x 4 columns)  
## 1. z: x/"a. u." [numeric] 0  
## 2. z.end: x/"a. u." [numeric] 0  
## 3. spc: I/"a. u." [matrix301] 95.5881 98.6830 ... 94.45682  
## 4. filename: filename [character] ./data/pet_unstab_cyclic_quv/sa19603_00_ex320.spc
```

#### 10.1.0.2.2 Basic Plotting

```
# Plot with hyperSpec base plotting
```

```
plotspc(step0)
```



### 10.1.0.2.3 Multiple .spc Files?

- (Purposefully without a loop or apply)

```
# Read multiple .spc files

# setwd("./data/pet_unstab_cyclic_quv_ex320/")
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_00_ex320.spc"
step0 <- read.spc(curfile)
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_22_ex320.spc"
step1 <- read.spc(curfile)
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_23_ex320.spc"
step2 <- read.spc(curfile)
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_24_ex320.spc"
step3 <- read.spc(curfile)
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_25_ex320.spc"
step4 <- read.spc(curfile)
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_26_ex320.spc"
step5 <- read.spc(curfile)
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_27_ex320.spc"
step6 <- read.spc(curfile)
curfile <- "./data/pet_unstab_cyclic_quv/sa19603_28_ex320.spc"
step7 <- read.spc(curfile)
```

### 10.1.0.2.4 Combine into one object (Good News!)

```
# Use Collpase to combine spectra

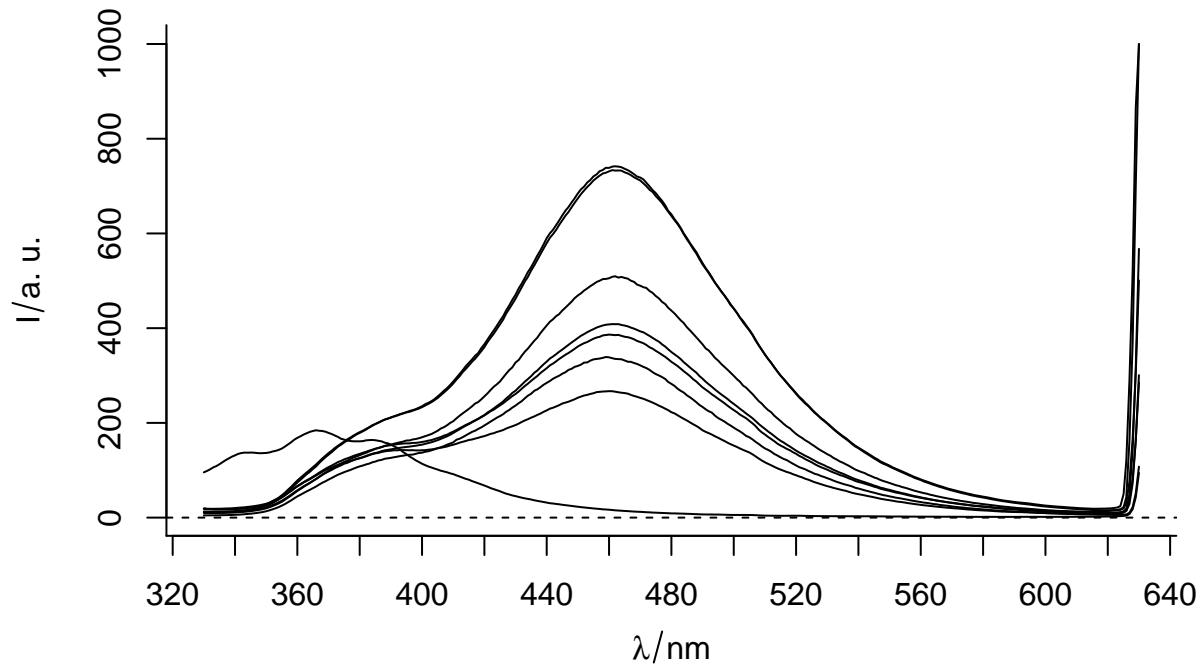
spec <- hyperSpec::collapse(step0, step1, step2, step3, step4, step5,
                             step6, step7)
class(spec)
```

```
## [1] "hyperSpec"
## attr(,"package")
## [1] "hyperSpec"
spec
```

```
## hyperSpec object
##      8 spectra
##      4 data columns
##      301 data points / spectrum
## wavelength: lambda/nm [numeric] 330.00 331.07 ... 630
## data: (8 rows x 4 columns)
##      1. z: x/"a. u." [numeric] 0 0 ... 0
##      2. z.end: x/"a. u." [numeric] 0 0 ... 0
##      3. spc: I/"a. u." [matrix301] 95.58810 18.86919 ... 1000
##      4. filename: filename [character] ./data/pet_unstab_cyclic_quv/sa19603_00_ex320.spc ./data/pet_unstab_cyclic_quv/sa19603_22_ex320.spc ...
```

### 10.1.0.2.5 Basic Plotting

```
plotspc(spec)
```



#### 10.1.0.2.6 Spectral Range Selection

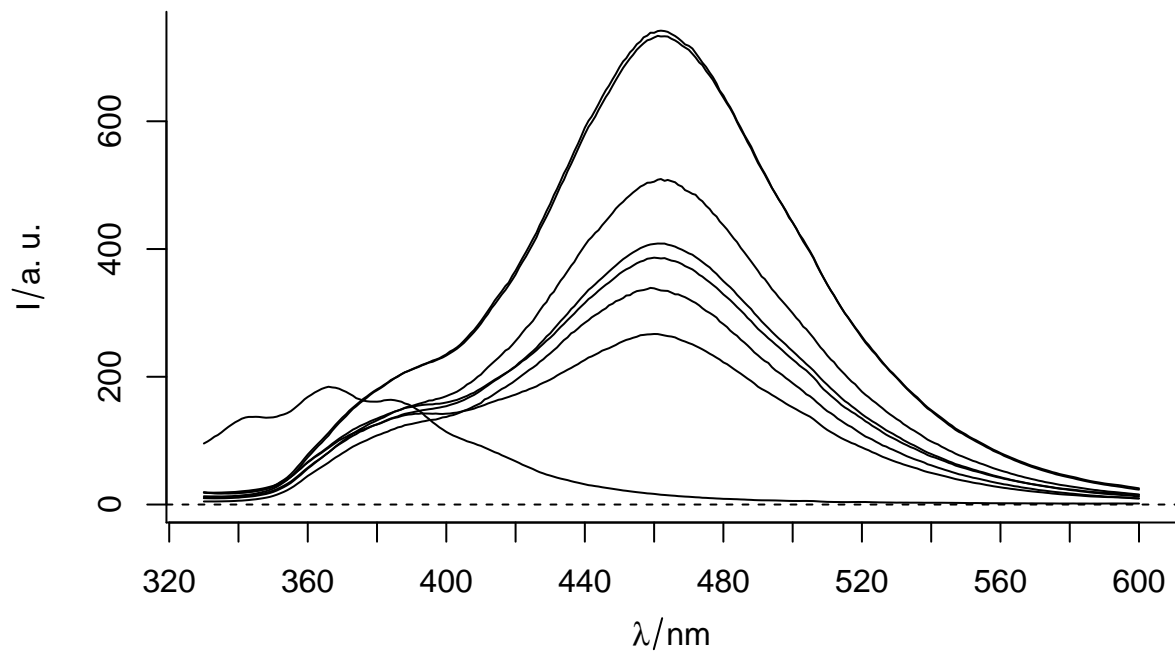
*# Subset and redefine the hyperSpec object according to wavelength*

```
spec <- spec[, , min ~ 600]
spec
```

```
## hyperSpec object
##      8 spectra
##      4 data columns
##      271 data points / spectrum
## wavelength: lambda/nm [numeric] 330.00 331.07 ... 600
## data: (8 rows x 4 columns)
##      1. z: x/"a. u." [numeric] 0 0 ... 0
##      2. z.end: x/"a. u." [numeric] 0 0 ... 0
##      3. spc: I/"a. u." [matrix271] 95.58810 18.86919 ... 25.40894
##      4. filename: filename [character] ./data/pet_unstab_cyclic_quv/sa19603_00_ex320.spc ./data/pet_unstab_cyclic_quv/sa19603_00_ex320.spc
```

#### 10.1.0.2.7 Basic Plotting

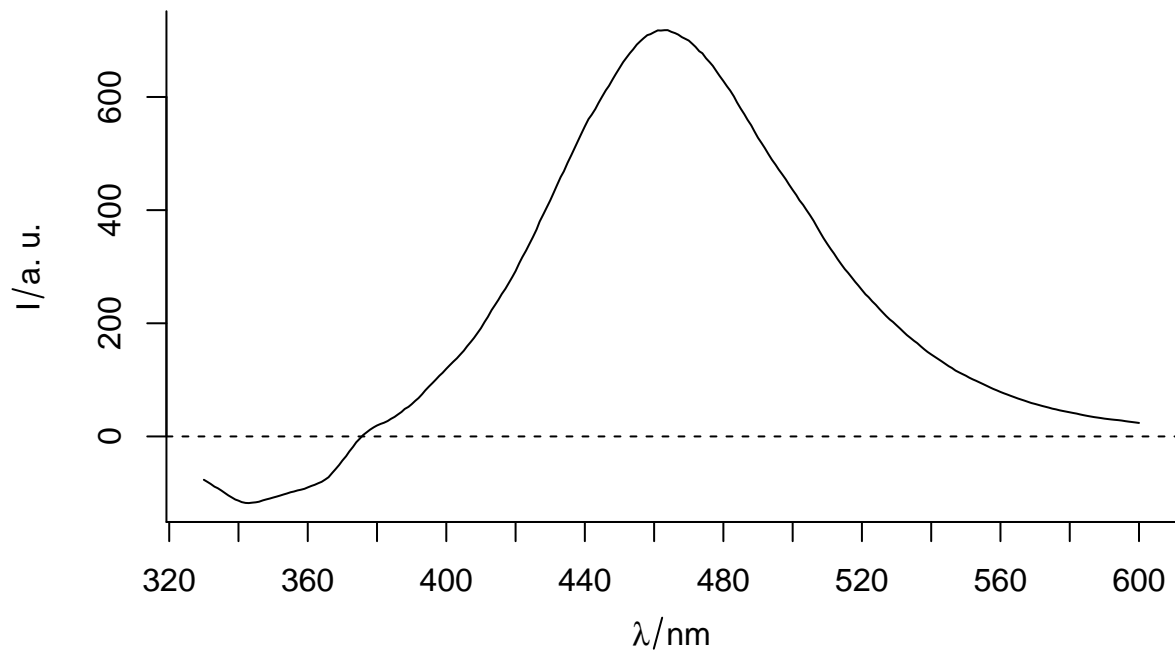
```
plotspc(spec)
```



#### 10.1.0.2.8 Spectral Arithmetic (Subtraction)

*# Perform spectral subtraction*

```
sub <- step7 - step0
plotspc(sub[, min ~ 600])
```



#### 10.1.0.2.9 Removing Bad Data

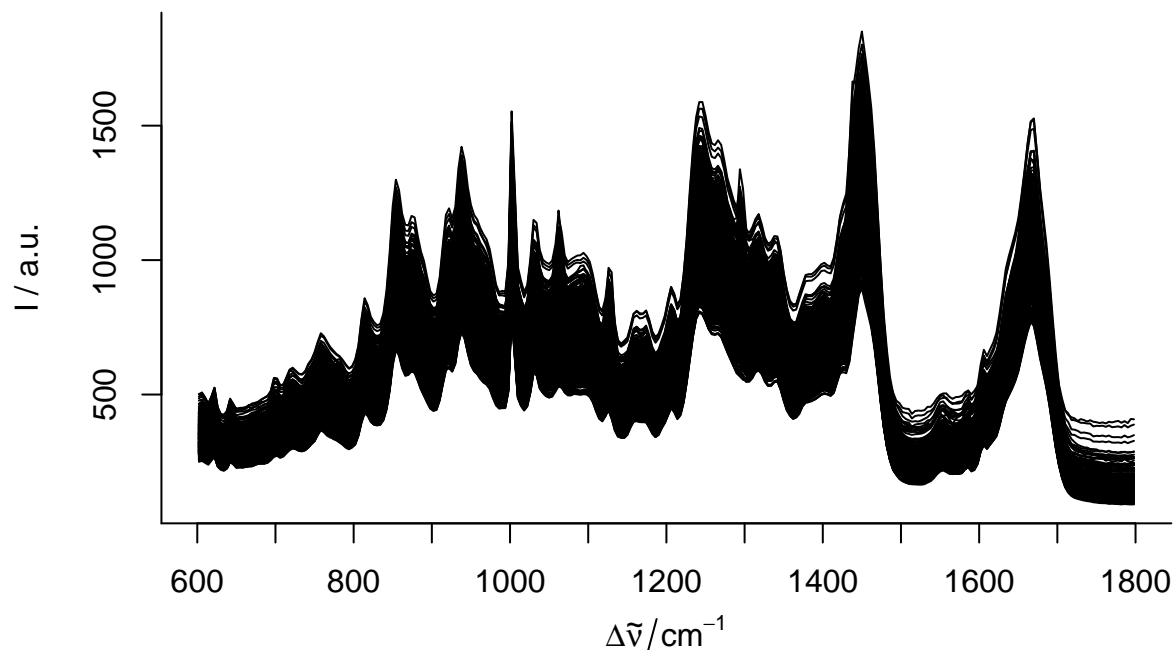
*# Let's look at some fake IR data*

```
ir.spc <- chondro
```

```
ir.spc
```

```
## hyperSpec object
##      875 spectra
##      5 data columns
##      300 data points / spectrum
## wavelength: Delta * tilde(nu)/cm^-1 [numeric] 602 606 ... 1798
## data: (875 rows x 5 columns)
##      1. y: y [numeric] -4.77 -4.77 ... 19.23
##      2. x: x [numeric] -11.55 -10.55 ... 22.45
##      3. filename: filename [character] rawdata/chondro.txt rawdata/chondro.txt ... rawdata/chondro.txt
##      4. clusters: clusters [factor] matrix matrix ... lacuna + NA
##      5. spc: I / a.u. [matrix300] 501.8194 500.4552 ... 169.2942
```

```
plotspc(ir.spc, spc.nmax = length(ir.spc))
```



#### 10.1.0.2.10 Removing Bad Data

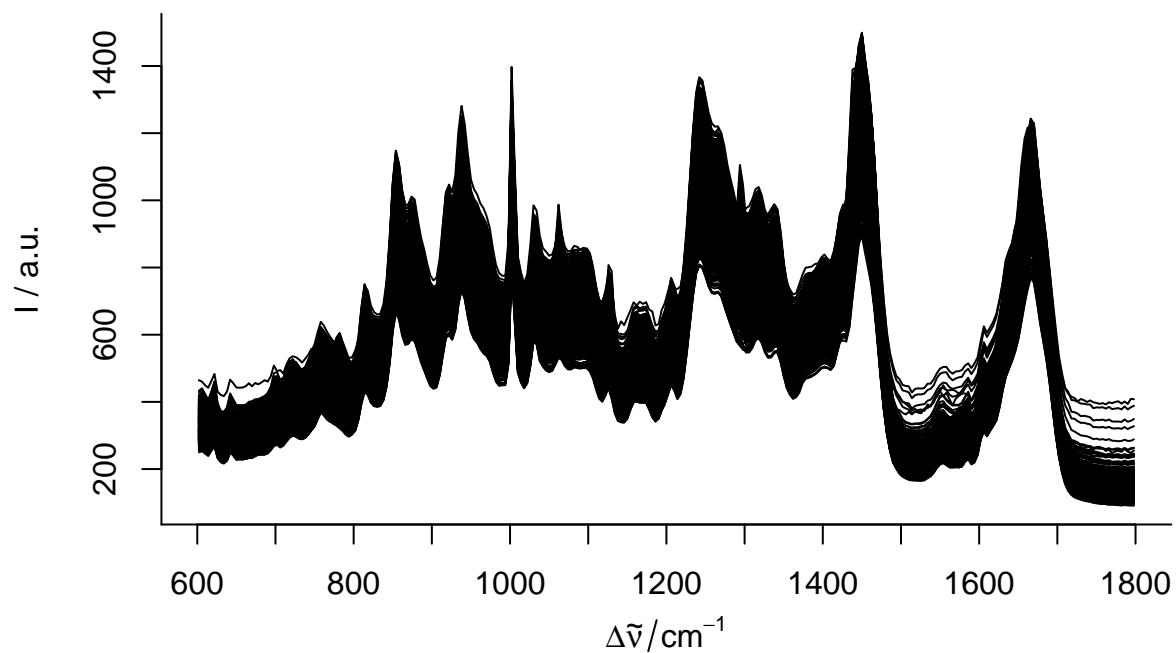
```
# Define any point above 1500 as bad
high.int <- apply(ir.spc > 1500, 1, any)

# Maximum should atleast be 0.1
low.int <- apply(ir.spc, 1, max) < 0.1

# Apply Conditions
ir.spc <- ir.spc[!high.int & !low.int]
```

#### 10.1.0.2.11 Removed the bad spectra

```
plotspc(ir.spc, spc.nmax = length(ir.spc))
```



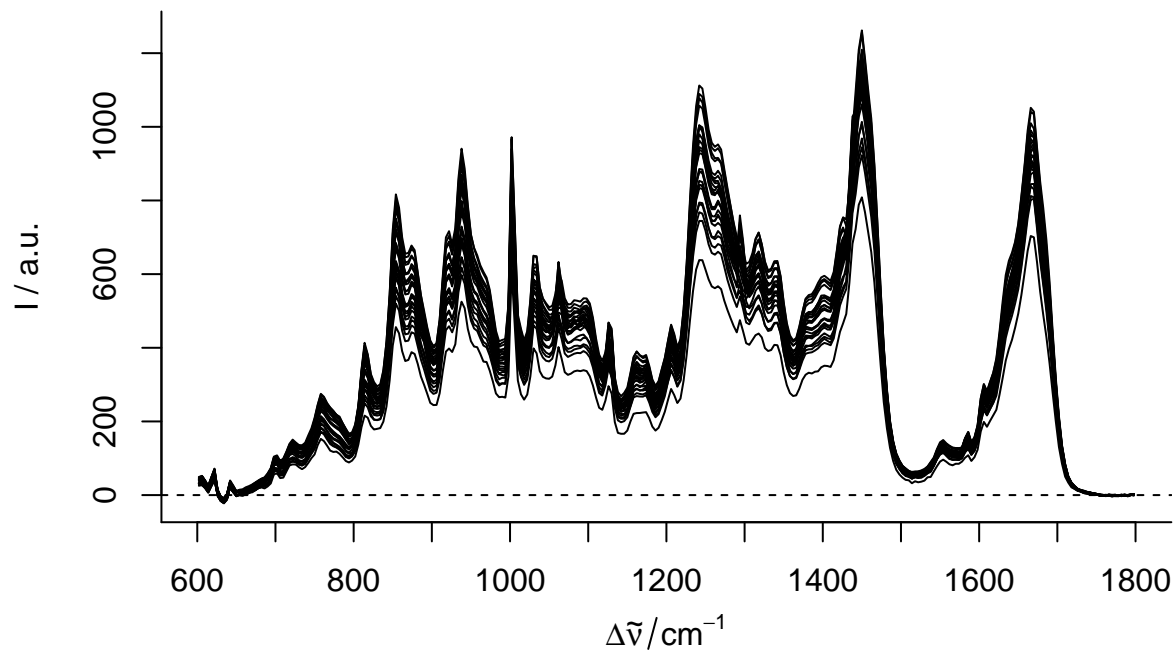
#### 10.1.0.2.12 cq.abs.step0 Correction (One of many options)

```
# Apply baseline correction function
blcorr <- spc.fit.poly.below(ir.spc)

# Subtract away correction from original
ir.spc <- ir.spc - blcorr
```

#### 10.1.0.2.13 Corrected Spectra!

```
plotspc(ir.spc)
```





### 10.1.0.3 Example of use, Key Files for sample Metadata

#### 10.1.0.3.1 Load Cyclic QUV Absorbance Data Set

```
# setwd("v:/vuv-data/proj/3M")

## Read keyfile(s) and set classes

pet.key <- read.csv("./data/pet_unstab_cyclic_quv/3m_sample_key.csv")
pet.key$Sample <- as.character(pet.key$Sample)
pet.key$Product <- as.character(pet.key$Product)
pet.key$Exposure <- as.character(pet.key$Exposure)
pet.key$Step_Retained <- as.character(pet.key$Step_Retained)

## Read step 0 (Unexposed) CyclicQUV Optical Absorbance Data

filenames <- list.files(path = "./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step0/",
                        pattern = "\\*.spc$")

# Read Files, Here using lapply

cq.abs.step0 <- lapply(filenames,function(i){
  read.spc(paste("./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step0/", i,
                sep = ""))
})

# Combine resulting list into one hyperSpec object

cq.abs.step0 <- hyperSpec::collapse(cq.abs.step0[1:length(cq.abs.step0)])

# Add step data

cq.abs.step0@data$step <- 0

## Extract sample number from file name using sub (grep)
## to pull the sample number from the file name
## match to row in the keyfile

cq.abs.step0@data$sample <- filenames %>%
  sub(pattern = "-es00-ms00-mn01_uvvs01.spc", replacement = "") %>%
  sub(pattern = "_", replacement = ".")

# Match material info from key file and add to hyperSpec object

rows <- cq.abs.step0@data$sample %>% pmatch(pet.key[,1])
cq.abs.step0@data$material <- pet.key[rows,2]

## Read step 1 CyclicQUV Optical Absorbance Data

filenames <- list.files(path = "./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step1/",
                        pattern = "\\*.SPC$")

# Read Files
```

```

cq.abs.step1 <- lapply(fileNames,function(i){
  read.spc(paste("./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step1/",i,sep = ""))
})

# Combine resulting list into one hyperSpec object

cq.abs.step1 <- hyperSpec::collapse(cq.abs.step1[1:length(cq.abs.step1)])

# Add step data

cq.abs.step1@data$step <- 1

## Extract sample number from file name using sub (grep)
## to pull the sample number from the file name match to row in the keyfile

cq.abs.step1@data$sample <- fileNames %>%
  sub(pattern = "-es01-ms01-mn01_uvvs01.SPC",replacement = "") %>%
  sub(pattern = "_", replacement = ".")

# Match material info from key file and add to hyperSpec object

rows <- cq.abs.step1@data$sample %>% pmatch(pet.key[,1])
cq.abs.step1@data$material <- pet.key[rows,2]

## Read step 2 CyclicQUV Optical Absorbance Data

fileNames <- list.files(path = "./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step2/",
  pattern = "\\..SPC$")

# Read Files

cq.abs.step2 <- lapply(fileNames,function(i){
  read.spc(paste("./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step2/",
    i, sep = ""))
})

# Combine resulting list into one hyperSpec object

cq.abs.step2 <- hyperSpec::collapse(cq.abs.step2[1:length(cq.abs.step2)])

# Add step data

cq.abs.step2@data$step <- 2

## Extract sample number from file name using sub (grep)
## to pull the sample number from the file name match to row in the keyfile

cq.abs.step2@data$sample <- fileNames %>%
  sub(pattern = "-es02-ms02-mn01_uvvs01.SPC",replacement = "") %>%
  sub(pattern = "_",replacement = ".")

```

```
# Match material info from key file and add to hyperSpec object
```

```
rows <- cq.abs.step2@data$sample %>% pmatch(pet.key[,1])  
cq.abs.step2@data$material <- pet.key[rows,2]
```

Lets check out the contents of each hyperSpec Object

```
cq.abs.step0
```

```
## hyperSpec object  
##      6 spectra  
##      7 data columns  
##      3201 data points / spectrum  
## wavelength: lambda/nm [numeric] 1800.0 1799.5 ... 200  
## data: (6 rows x 7 columns)  
##      1. z: x/"a. u." [numeric] 0 0 ... 0  
##      2. z.end: x/"a. u." [numeric] 0 0 ... 0  
##      3. spc: A [matrix3201] 0.1272863 0.1177481 ... 0.2635259  
##      4. filename: filename [character] ./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step0/sa25000_01-es  
##      5. step: [numeric] 0 0 ... 0  
##      6. sample: [character] sa25000.01 sa25002.01 ... sa25010.01  
##      7. material: [character] B-Melinox243 B-Melinox238 ... B-TeteronU2L92W
```

```
cq.abs.step1
```

```
## hyperSpec object  
##      54 spectra  
##      7 data columns  
##      3201 data points / spectrum  
## wavelength: lambda/nm [numeric] 1800.0 1799.5 ... 200  
## data: (54 rows x 7 columns)  
##      1. z: x/"a. u." [numeric] 0 0 ... 0  
##      2. z.end: x/"a. u." [numeric] 0 0 ... 0  
##      3. spc: A [matrix3201] 0.1714494 0.1342367 ... 0.2494305  
##      4. filename: filename [character] ./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step1/sa25000_26-es  
##      5. step: [numeric] 1 1 ... 1  
##      6. sample: [character] sa25000.26 sa25000.27 ... sa25010.34  
##      7. material: [character] B-Melinox243 B-Melinox243 ... B-TeteronU2L92W
```

```
cq.abs.step2
```

```
## hyperSpec object  
##      48 spectra  
##      7 data columns  
##      3201 data points / spectrum  
## wavelength: lambda/nm [numeric] 1800.0 1799.5 ... 200  
## data: (48 rows x 7 columns)  
##      1. z: x/"a. u." [numeric] 0 0 ... 0  
##      2. z.end: x/"a. u." [numeric] 0 0 ... 0  
##      3. spc: A [matrix3201] -0.030848697 -0.001725956 ... 0.09484347  
##      4. filename: filename [character] ./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step2/sa25000_27-es  
##      5. step: [numeric] 2 2 ... 2  
##      6. sample: [character] sa25000.27 sa25000.28 ... sa25010.34  
##      7. material: [character] B-Melinox243 B-Melinox243 ... B-TeteronU2L92W
```

We'll put these all together

```
# Combine cq.abs.step0s with step 1
```

```
cq.abs <- hyperSpec::collapse(cq.abs.step0, cq.abs.step1, cq.abs.step2)
```

```
cq.abs
```

```
## hyperSpec object
```

```
## 108 spectra
```

```
## 7 data columns
```

```
## 3201 data points / spectrum
```

```
## wavelength: lambda/nm [numeric] 1800.0 1799.5 ... 200
```

```
## data: (108 rows x 7 columns)
```

```
## 1. z: x/"a. u." [numeric] 0 0 ... 0
```

```
## 2. z.end: x/"a. u." [numeric] 0 0 ... 0
```

```
## 3. spc: A [matrix3201] 0.1272863 0.1177481 ... 0.09484347
```

```
## 4. filename: filename [character] ./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step0/sa25000_01-es
```

```
## 5. step: [numeric] 0 0 ... 2
```

```
## 6. sample: [character] sa25000.01 sa25002.01 ... sa25010.34
```

```
## 7. material: [character] B-Melindex243 B-Melindex238 ... B-TeteronU2L92W
```

#### 10.1.0.4 Utilizing spectral arithmetic to change ordinate axis

```
# Divide by sample thickness for Abs/cm
```

```
t.melindex243 <- 0.005
```

```
t.melindex238 <- 0.0127
```

```
t.melindex626 <- 0.0127
```

```
t.melindex618 <- 0.0127
```

```
t.teteronu2 <- 0.005
```

```
t.teteronu2l92w <- 0.0125
```

```
for (i in 1:length(cq.abs@data$material)) {
```

```
  if (cq.abs@data$material[i] == "B-Melindex243") {cq.abs[i] <- cq.abs[i]/t.melindex243}
```

```
  if (cq.abs@data$material[i] == "B-Melindex238") {cq.abs[i] <- cq.abs[i]/t.melindex238}
```

```
  if (cq.abs@data$material[i] == "A-Melindex626") {cq.abs[i] <- cq.abs[i]/t.melindex626}
```

```
  if (cq.abs@data$material[i] == "A-Melindex618") {cq.abs[i] <- cq.abs[i]/t.melindex618}
```

```
  if (cq.abs@data$material[i] == "B-TeteronU2") {cq.abs[i] <- cq.abs[i]/t.teteronu2}
```

```
  if (cq.abs@data$material[i] == "B-TeteronU2L92W") {cq.abs[i] <- cq.abs[i]/t.teteronu2l92w}
```

```
}
```

```
cq.abs
```

```
## hyperSpec object
```

```
## 108 spectra
```

```
## 7 data columns
```

```
## 3201 data points / spectrum
```

```
## wavelength: lambda/nm [numeric] 1800.0 1799.5 ... 200
```

```
## data: (108 rows x 7 columns)
```

```
## 1. z: x/"a. u." [numeric] 0 0 ... 0
```

```
## 2. z.end: x/"a. u." [numeric] 0 0 ... 0
```

```
## 3. spc: A [matrix3201] 25.457266 9.271504 ... 7.587478
```

```
## 4. filename: filename [character] ./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step0/sa25000_01-es
```

```
## 5. step: [numeric] 0 0 ... 2
```

```
## 6. sample: [character] sa25000.01 sa25002.01 ... sa25010.34
## 7. material: [character] B-Melinox243 B-Melinox238 ... B-TetersonU2L92W

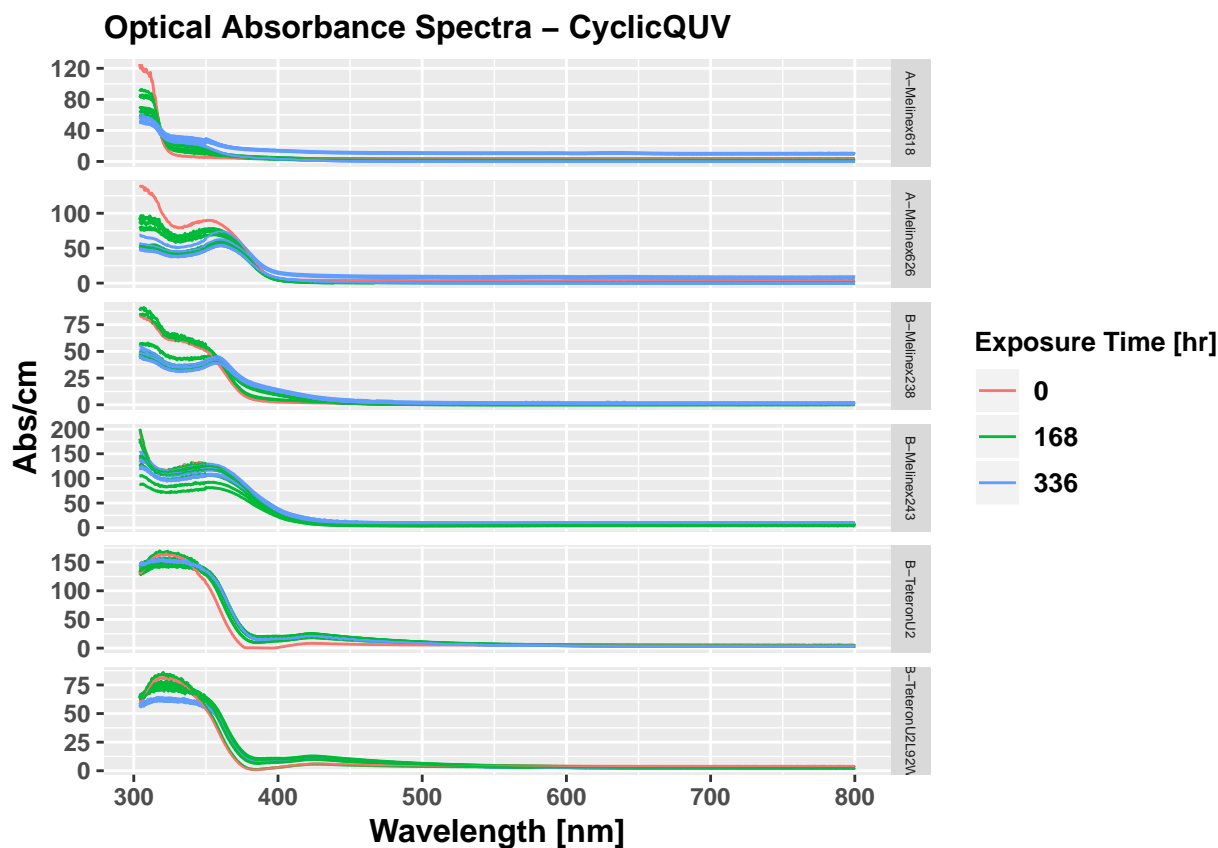
# Remove erroneous points (negative)

`cq.abs`[[`cq.abs` < 0]] <- NA

cq.abs

## hyperSpec object
## 108 spectra
## 7 data columns
## 3201 data points / spectrum
## wavelength: lambda/nm [numeric] 1800.0 1799.5 ... 200
## data: (108 rows x 7 columns)
## 1. z: x/"a. u." [numeric] 0 0 ... 0
## 2. z.end: x/"a. u." [numeric] 0 0 ... 0
## 3. spc: A [matrix3201] 25.457266 9.271504 ... 7.587478 + NA
## 4. filename: filename [character] ./data/pet_unstab_cyclic_quv/Cary/CyclicQUV/step0/sa25000_01-es
## 5. step: [numeric] 0 0 ... 2
## 6. sample: [character] sa25000.01 sa25002.01 ... sa25010.34
## 7. material: [character] B-Melinox243 B-Melinox238 ... B-TetersonU2L92W
```

#### 10.1.0.4.1 Better plotting using ggplot2 graphics



#### 10.1.0.5 Peak Information extraction

We can directly address the intensity value at a specific wavelength

```
spec[2,,460]@data$spc
```

```
##           460
## [1,] 266.9985
```

Using this concept,

- we can also find full width at half max
  - by approximating the wavelengths
  - at half the intensity of the peak

```
lower <- approx(spec[2,,330 ~ 460]$spc, spec[2,,330 ~ 460]@wavelength,
               xout = c(spec[2,,460]$spc/2))
lower
```

```
## $x
## [1] 133.4993
##
## $y
## [1] 383.7909
```

```
upper <- approx(spec[2,,460 ~ 600]$spc, spec[2,,460 ~ 600]@wavelength,
               xout = c(spec[2,,460]$spc/2))
upper
```

```
## $x
## [1] 133.4993
##
## $y
## [1] 506.0615
```

```
fwhm <- upper$y - lower$y
fwhm
```

```
## [1] 122.2706
```

Imagine this as a function that could operate on multiple spectra

```
fwhm.spc <- function(spec, peak.wl, lower.bound, upper.bound){

  peak.wl <- as.numeric(peak.wl)

  lower <- approx(spec[, ,lower.bound ~ peak.wl]$spc,
                  spec[, ,lower.bound ~ peak.wl]@wavelength,
                  xout = c(spec[, ,peak.wl]$spc/2))

  upper <- approx(spec[, ,peak.wl ~ upper.bound]$spc,
                  spec[, ,peak.wl ~ upper.bound]@wavelength,
                  xout = c(spec[, ,peak.wl]$spc/2))

  fwhm <- upper$y - lower$y
  center <- mean(c(upper$y, lower$y))
  skew <- 1 - (upper$y - peak.wl)/(fwhm/2)

  return(c(fwhm, center, skew))
}
```

Let's try it

```

#Start a data frame to hold the data

ex320data <- read.csv("./data/petpilot-key.csv")

ex320data$Em.Peak <- NA
ex320data$Em.Peak.int <- NA
ex320data$Em.Peak.fwhm <- NA
ex320data$Em.Peak.center <- NA
ex320data$Em.Peak.skew <- NA
ex320data$Em.Peak.skew.mag <- NA

# We'll use a for loop to run each spectrum in the hyperSpec Object
row <- 1

for (i in 1:length(spec)) {

  # Define temporary holder for spectrum of interest

  temp <- spec[i]

  #Define the upper and lower bounds for the approx function

  lower.bound <- min(temp@wavelength)
  upper.bound <- max(temp@wavelength)

  # Find the maximum value in the region of interest
  peak <- max(temp[,450~470])
  ex320data$Em.Peak.int[row] <- peak

  # Find the peak wavelength

  peak.out <- approx(temp[,450 ~ 470]$spc, temp[,450 ~ 470]@wavelength,
                     xout = max(temp[,450~470]))
  peak.wl <- ifelse(is.na(peak.out$y) == TRUE, 450, peak.out$y)
  ex320data$Em.Peak[row] <- peak.wl

  # Run our function

  spec.data <- fwhm.spc(temp, as.numeric(peak.wl), lower.bound, upper.bound)

  # Store values in data frame

  ex320data$Em.Peak.fwhm[row] <- spec.data[1]
  ex320data$Em.Peak.center[row] <- spec.data[2]
  ex320data$Em.Peak.skew[row] <- spec.data[3]
  row <- row + 1
}

```

Check the result

```
head(ex320data)
```

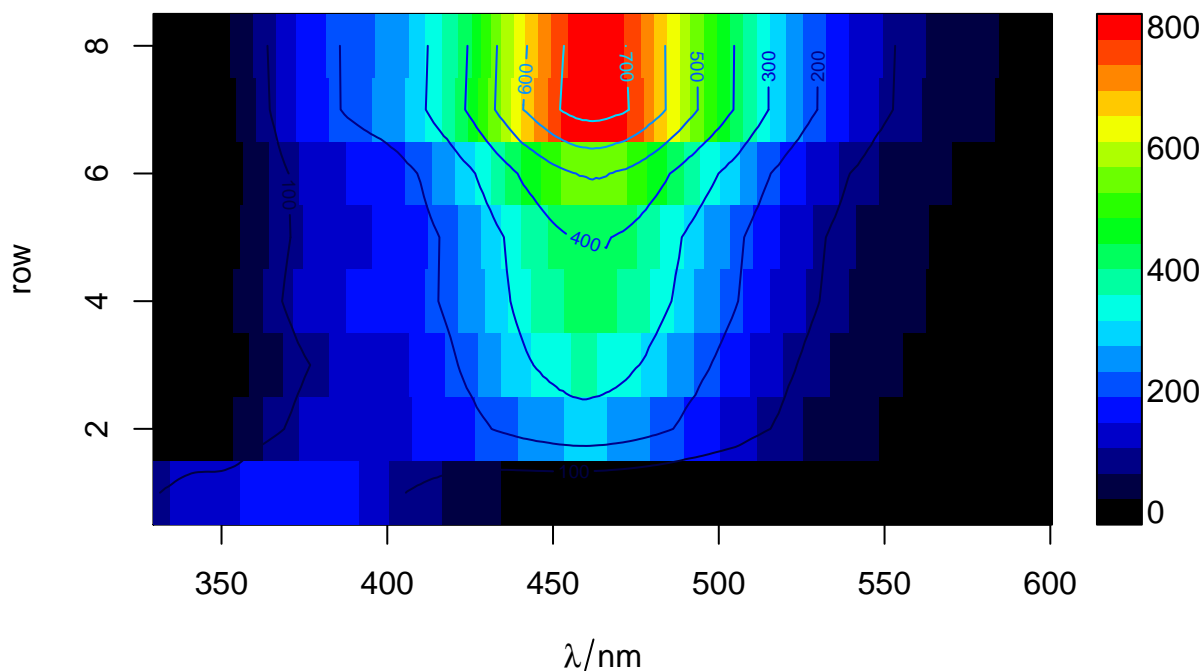
```
##           Sample           Material.Type Exposure.Type Exposure.Step
## 1 sa19601.00 hydrolytically.stabilized      baseline              0
```

```
## 2 sa19601.01 hydrolytically.stabilized      dampheat      1
## 3 sa19601.02 hydrolytically.stabilized      dampheat      2
## 4 sa19601.03 hydrolytically.stabilized      dampheat      3
## 5 sa19601.04 hydrolytically.stabilized      dampheat      4
## 6 sa19601.05 hydrolytically.stabilized      dampheat      5
##   Exposure.Time Total.UV.Dose Em.Peak Em.Peak.int Em.Peak.fwhm
## 1           0           0 450.00   22.99323         NA
## 2          168           0 460.00  266.99850  122.27058
## 3          336           0 458.93  339.14621   92.05087
## 4          504           0 460.00  386.45648   93.64712
## 5          672           0 461.06  408.56882   90.03078
## 6          840           0 461.96  509.58521   87.02260
##   Em.Peak.center Em.Peak.skew Em.Peak.skew.mag
## 1             NA             NA             NA
## 2         444.9262   0.246564551             NA
## 3         458.9813  -0.001115496             NA
## 4         460.1745  -0.003725775             NA
## 5         461.9307  -0.019342267             NA
## 6         463.3233  -0.031332943             NA
```

### 10.1.0.6 Other plots

#### 10.1.0.6.1 Matrix Plot - Fluorescence Data

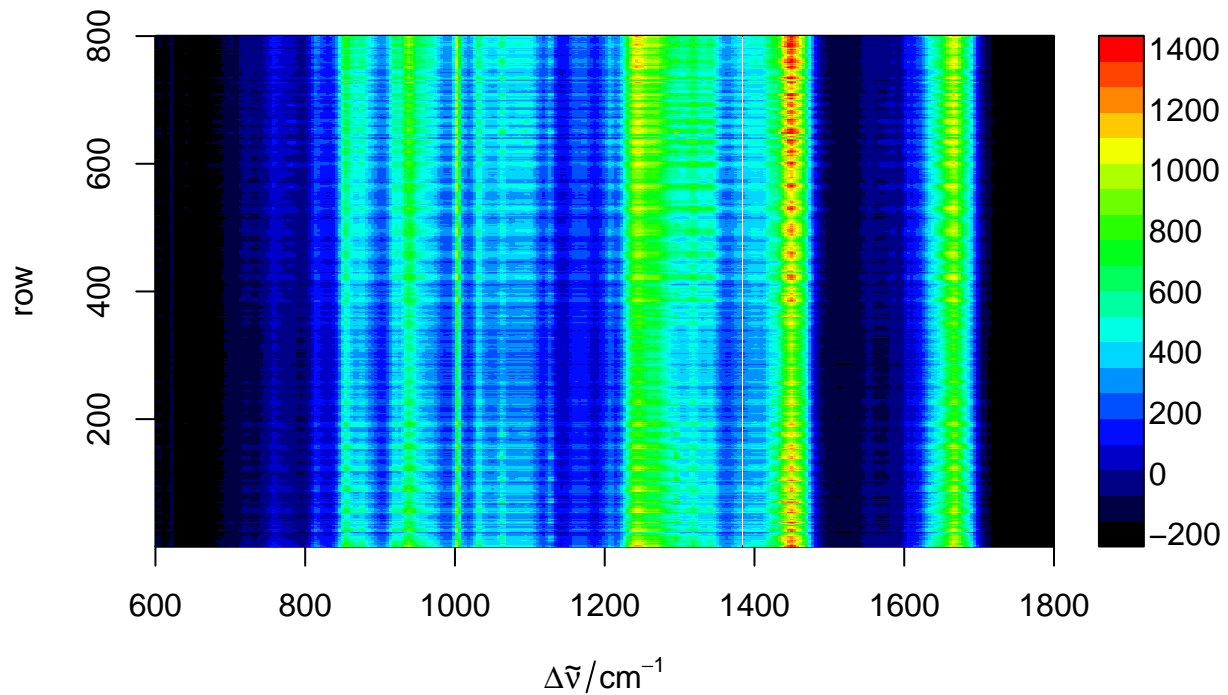
```
plotmat(spec)
plotmat(spec, contour = TRUE, add = TRUE)
```



#### 10.1.0.6.2 Matrix Plot - IR Data

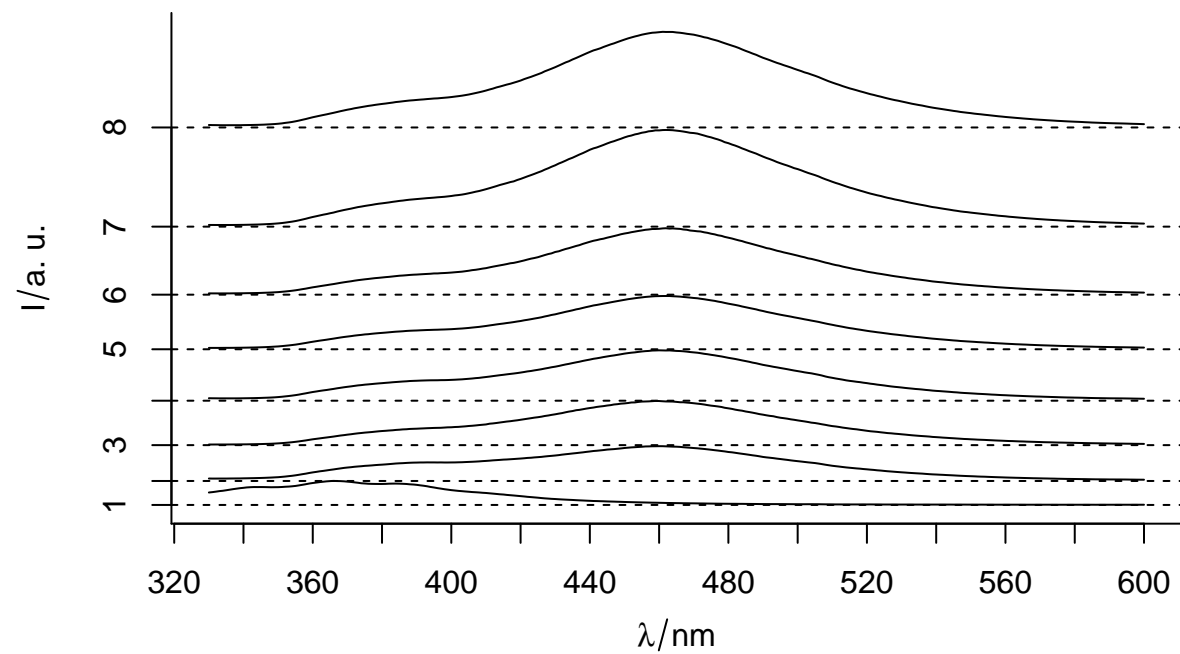
```
plotmat(ir.spc)
```





### 10.1.0.6.3 Stacked Spectra

```
plotspc(spec, stacked = TRUE)
```



### 10.1.0.7 Summary

#### 10.1.0.7.1 hyperSpec

- hyperSpec is a powerful R package for handling and manipulating spectral data in R

- More to come!