

Testing Plan for baselineARPLss package (DATA501)

Installation

The following steps should work either on your own computer or on a free project / instance in the R Cloud (<https://posit.cloud/>)

The steps to download the package and install it are the following:

```
install.packages('remotes')
library(remotes)
remotes::install_github("econdatatech/baselineARPLss")
```

Please try these steps and report the result. The expected result is an installation without errors.

Usage

Usage with supplied sample data

Baseline estimation

Try the following instructions on in your R environment and report the result. The execution might take a couple of minutes and should finish without an error.

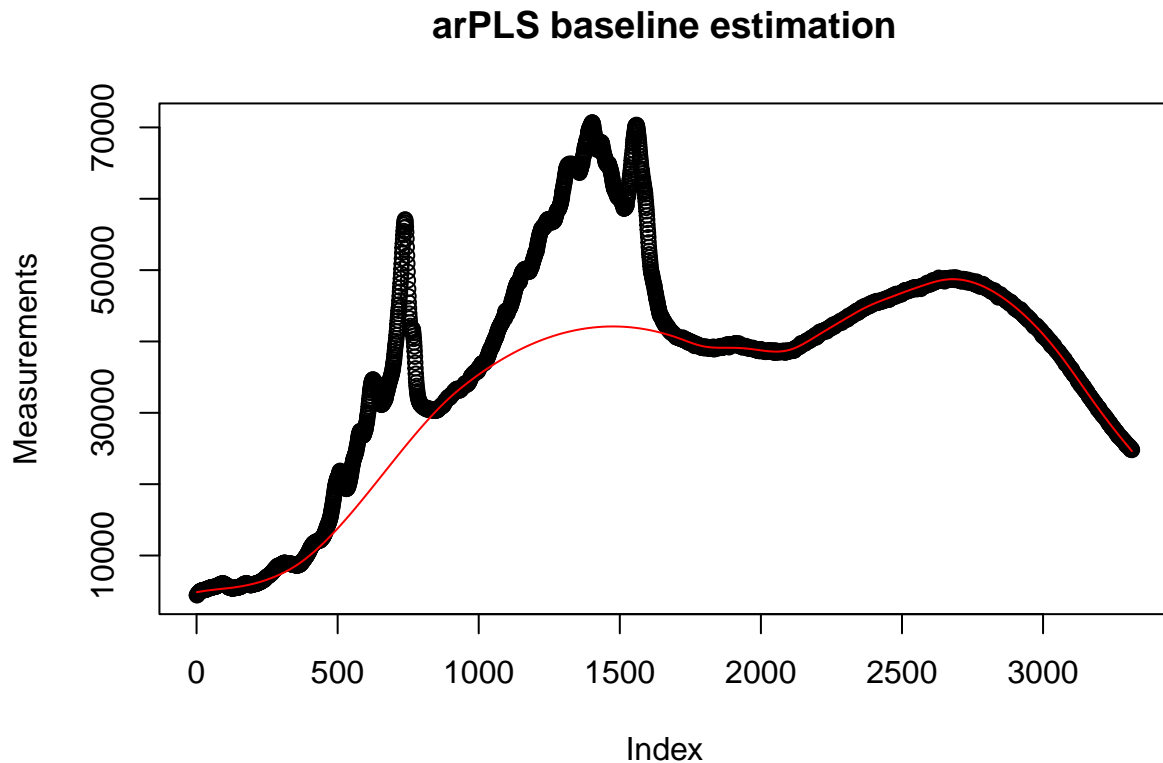
```
library(baselineARPLss)
data("Abelsonite")

res<-baseline_estimation(Abelsonite$measurement)
```

expected output: Result plot

Issue the below command and compare the result with the picture below.

```
plot(res)
```



Issue the below command and compare the result with the output below.

```
summary(res)
```

```
## [1] "The lambda parmeter value used was: 1e+06"
## [1] "The ratio parameter value used was: 1e-06"
## [1] "The max_iter parameter value used was: 50"
## [1] "The alogrithm stopped after the following number or iterations: 50"
## [1] "The last weight ratio value was: 0.000153623596796175"
## [1] "It appears that the algorithm stopped because the maximum number of iterations was reached"
```

Expected output with synthetic data

Execute the following test (with the expected result of FALSE)

```
#Generate testdata
wavenumbers <- seq(0, 1000, length.out = 1000)
raman_spectra <- 100*exp(-((wavenumbers-300)/15)^2) + 200*exp(-((wavenumbers-750)/30)^2) #+ 100*exp(-((wavenumbers-1500)/50)^2)

sum(baseline_estimation(wavenumbers+raman_spectra,verbose=FALSE)$baseline-wavenumbers)>=0.001

## [1] FALSE
```

Misc

Try to obtain the help page for the baseline_estimation function.

```
?baseline_estimation
```

```
## starting httpd help server ... done
```

Give your opinion. E.g. is anything unclear or anything missing?

Usage with new data

Select a substance or mineral of your choice from this following URL https://rruff.info/*/display=default/ (The site might take a bit to load) Click on the magnifying glass on the right hand side of the screen for the substance of your choice. On the next page look for the following section “BROAD SCAN WITH SPECTRAL ARTIFACTS” and download the “Raman Data (RAW)” file. If such a file is not available, choose a different substance in the previous screen and try to obtain a “Raman Data (RAW)” for this substance.

Read the downloaded file into a dataframe (with a command similar to the one below)

```
df<-read.table('C:\\Users\\corvini\\Downloads\\Actinolite_R040063_Broad_Scan_532_0_unoriented_Ram
```

Try to estimate the baseline for the spectrum of this substance with the help of the documentation available (e.g. help files and vignette and manual) Report your findings.

Input validation

The `baseline_estimation` function has a lot of parameters. Most (apart from the input data in form of the raw spectrum) have default parameters.

```
baseline_estimation <- function(y, lambda = 1e6, ratio = 1e-6, max_iter = 50, verbose=FALSE, algo="banded")
```

- y: Numeric vector representing the spectrum.
- lambda: Smoothing parameter. floating point number
- ratio: Stopping criterion based on changes in weight vector per iteration. floating point number
- max_iter: Maximum number of iterations. Integer number
- verbose: Boolean to print intermediary outputs
- algo: String to choose solver between Armadillo CPP `armaInv` (“cpp”) and native solver function “native” and `limSolve::Solve.banded` solver (“banded”)

Try different values than the default value for each parameter and report your findings. Furthermore try to feed the parameters with wrong data types and report your findings. Here are some example tests to perform: Wrong input type for the spectrum

```
baseline_estimation(y = "This is a sting")
```

```
## Error in baseline_estimation(y = "This is a sting"): The input data must be of type numeric.
```

Missing input for the spectrum

```
baseline_estimation(lambda = 1e6, ratio = 1e-6, max_iter = 50, verbose=FALSE)
```

```
## Error in baseline_estimation(lambda = 1e+06, ratio = 1e-06, max_iter = 50, : Missing input data. You
```

Presence of NAs in the spectrum

```
baseline_estimation(c(1,2,3,4,NA))
```

```
## Error in baseline_estimation(c(1, 2, 3, 4, NA)): The input spectrum contains NA values. You need to
```

Presence of Infs in the spectrum

```
baseline_estimation(c(1,2,3,4,Inf))
```

```
## Error in baseline_estimation(c(1, 2, 3, 4, Inf)): The input spectrum contains Inf values. You need to
```

Spectrum has less than 3 values

```
baseline_estimation(c(1,2))
```

```
## Error in baseline_estimation(c(1, 2)): The input spectrum must contain at least 3 values
```

Spectrum has negative values

```
baseline_estimation(c(-1,2,0))
```

```
## Error in baseline_estimation(c(-1, 2, 0)): The input spectrum must contain only positive values
```

Lambda has wrong format and negative values

```
baseline_estimation(c(1,2,0),lambda=c(-1,2))
```

```
## Error in baseline_estimation(c(1, 2, 0), lambda = c(-1, 2)): The parameter 'lambda' must be a single value  
## NA, Inf or negative values are not valid.
```

Ratio parameter has wrong format and negative values

```
baseline_estimation(c(1,2,0),ratio=c(-1,2))
```

```
## Error in baseline_estimation(c(1, 2, 0), ratio = c(-1, 2)): The parameter 'ratio' must be a single value  
## NA, Inf or negative values are not valid.
```

Max_iter parameter has wrong format and negative values

```
baseline_estimation(c(1,2,0),max_iter=c(-1,2))
```

```
## Error in baseline_estimation(c(1, 2, 0), max_iter = c(-1, 2)): The parameter 'max_iter' must be a single value  
## NA, Inf or negative values or fractions and decimals are not valid.
```

Feel free to come up with your own input data “violation” tests.

Performance testing

Run the performance test below involving different solver algorithms. Report back the values for nativesolve, cppsolve and bandedsolve. CAVE: This will take quite a long time!

```
start_time <- Sys.time()
rn<-baseline_estimation(Abelsonite$measurement,algo="native")
end_time<- Sys.time()
nativesolve<-(end_time - start_time)
nativesolve
start_time <- Sys.time()
rc<-baseline_estimation(Abelsonite$measurement,algo="cpp")
end_time<- Sys.time()
cppsolve<-(end_time - start_time)
cppsolve
start_time <- Sys.time()
rb<-baseline_estimation(Abelsonite$measurement,algo="banded")
end_time<- Sys.time()
bandedsolve<-(end_time - start_time)
bandedsolve
```