Package ‘tvd’

February 20, 2015

Type Package

Title Total Variation Denoising

Version 0.1.0

Date 2014-08-11

Description Total Variation Denoising is a regularized denoising method which effectively removes noise from piecewise constant signals whilst preserving edges. This package contains a C++ implementation of Condat's very fast 1D squared error loss TVD algorithm. Additional methods and loss functions may be added in future versions.

License EPL (>= 1.0)

Depends R (>= 3.1.0)

Imports Rcpp (>= 0.11.2)

LinkingTo Rcpp

URL https://bitbucket.org/marpin/r-tvd

BugReports https://bitbucket.org/marpin/r-tvd/issues

Author Mark Pinese [aut, cre, cph]

Maintainer Mark Pinese <m.pinese@garvan.org.au>

NeedsCompilation yes

Repository CRAN

Date/Publication 2014-08-13 11:23:42

R topics documented:

tvd-package — *Total Variation Denoising*

Description

Total Variation Denoising

Details

tvd is a package for Total Variation Denoising, a regularized procedure for removing noise from piecewise constant signals whilst retaining edges. Currently implements Condat's algorithm for fast 1D TVD, in function tvd1d.

Author(s)

Mark Pinese <m.pinese@garvan.org.au>

References

Condat, L. (2013) A Direct Algorithm for 1-D Total Variation Denoising. IEEE Signal Processing Letters 20(11): 1054-1057. doi:10.1109/LSP.2013.2278339

See Also

tvd1d

tvd1d — *Perform Total Variation Denoising on a 1-Dimensional Signal*

Description

When supplied a noisy sequential signal in vector y, performs TVD with regularization parameter lambda, and returns a denoised version of y.

Usage

```
tvd1d(y, lambda, method = "Condat")
```

Arguments

y	a numeric vector of sequential noisy data values
lambda	the total variation penalty coefficient
method	a string indicating the algorithm to use for denoising. Currently only supports method "Condat"

Details

1D TVD is a filtering technique for a sequential univariate signal that attempts to find a vector x_tvd that approximates a noisy vector y, as:

$$x_{tvd} = argmin_x(E(x, y) + \lambda V(x))$$

where E(x, y) is a loss function measuring the error in approximating y with x, and V(x) is the total variation of x:

$$V(x) = sum(|x_{i+1} - x_i|)$$

TVD is particularly well-suited to recovering piecewise constant signals. The degree of approximation is controlled by the parameter lambda: for lambda = 0, x_tvd = y, and as lambda increases, x_tvd contains increasingly fewer value transitions, until, for a high enough value, it is constant.

Currently only implements Condat's fast squared-error loss TVD algorithm (method "Condat"), which is restricted to vectors of length 2^32 - 1 and shorter.

Value

a numeric vector of the same length as y, containing denoised data.

Author(s)

Mark Pinese <m.pinese@garvan.org.au>

References

Condat, L. (2013) A Direct Algorithm for 1-D Total Variation Denoising IEEE Signal Processing Letters 20(11): 1054-1057. doi:10.1109/LSP.2013.2278339

Examples

```
## Generate a stepped signal
x = rep(c(1, 2, 3, 4, 2, 4, 3, 2, 1), each = 100)

## Create a noisy version of the signal
y = x + rnorm(length(x), sd = 0.5)

## Denoise the signal by Condat's methodlines(x.denoised, col = "red", lwd = 1)
x.denoised = tvd1d(y, lambda = 10, method = "Condat")

## Plot the original signal, the noisy signal, and the denoised signal
plot(y, col = "black", pch = 19, cex = 0.3)
lines(x, col = "blue", lwd = 3)
lines(x.denoised, col = "red", lwd = 3)
legend("topleft", legend = c("Original", "Noisy", "Denoised"),
  col = c("blue", "black", "red"), lty = c("solid", "solid", "solid"),
  lwd = c(2, 0, 1), pch = c(NA, 19, NA), pt.cex = c(NA, 0.3, NA), inset = 0.05)
```

Index