

# Package ‘tvdenoising’

May 8, 2026

**Title** Univariate Total Variation Denoising

**Version** 1.0.0

**Description** Total variation denoising can be used to approximate a given sequence of noisy observations by a piecewise constant sequence, with adaptively-chosen break points. An efficient linear-time algorithm for total variation denoising is provided here, based on Johnson (2013) <doi:10.1080/10618600.2012.681238>.

**License** MIT + file LICENSE

**URL** <https://github.com/glmgen/tvdenoising>,  
<https://glmgen.github.io/tvdenoising/>

**BugReports** <https://github.com/glmgen/tvdenoising/issues>

**Imports** Rcpp, rlang

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0)

**LinkingTo** Rcpp

**Config/testthat/edition** 3

**Encoding** UTF-8

**RoxygenNote** 7.3.2

**NeedsCompilation** yes

**Author** Addison Hu [ctb],  
Daniel McDonald [ctb],  
Ryan Tibshirani [aut, cre, cph]

**Maintainer** Ryan Tibshirani <ryantibs@gmail.com>

**Depends** R (>= 3.5.0)

**Repository** CRAN

**Date/Publication** 2025-06-13 04:30:02 UTC

## Contents

tvdenoising . . . . .	2
<b>Index</b>	<b>3</b>

---

 tvdenoising

*Univariate total variation denoising*


---

### Description

Denoises a sequence of observations by solving the univariate total variation denoising optimization problem at a given regularization level.

### Usage

```
tvdenoising(y, lambda, weights = NULL)
```

### Arguments

y	Vector of observations to be denoised.
lambda	Regularization parameter value. Must be $\geq 0$ .
weights	Vector of observation weights. The default is NULL, which corresponds to unity weights. If specified, this vector must have the same length as y, and must have positive entries.

### Details

This function minimizes the univariate total variation denoising (also called fused lasso) criterion squares criterion

$$\frac{1}{2} \sum_{i=1}^n (y_i - \theta_i)^2 + \lambda \sum_{i=1}^{n-1} |\theta_{i+1} - \theta_i|,$$

over  $\theta$ . This is a special structured convex optimization problem which can be solved in linear time ( $O(n)$  operations) using algorithms based on dynamic programming (Viterbi) or taut string methods. The current function implements a highly-efficient dynamic programming method developed by Johnson (2013).

### Value

Vector of denoised values.

### References

Johnson (2013), "A dynamic programming algorithm for the fused lasso and L0-segmentation."

### Examples

```
y <- c(rep(0, 50), rep(3, 50)) + rnorm(100)
yhat <- tvdenoising(y, 5)
plot(y, pch = 16, col = "gray60")
lines(yhat, col = "firebrick", lwd = 2)
```

# Index

tvdenoising, [2](#)