Package: tweedie (via r-universe)

June 18, 2026

Version 3.1.0

Date 2026-05-18

Title Evaluation of Tweedie Exponential Family Models

Depends R (>= 2.8.0)

Encoding UTF-8

Imports methods, stats, graphics, lifecycle (>= 1.0.0), statmod (>= 1.4.0)

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Description Maximum likelihood computations for Tweedie families, including the series expansion (Dunn and Smyth, 2005; <doi:10.1007/s11222-005-4070-y>) and the Fourier inversion (Dunn and Smyth, 2008; <doi:10.1007/s11222-007-9039-6>), and related methods.

License GPL (>= 2)

NeedsCompilation yes

RoxygenNote 7.3.3

Config/testthat/edition 3

VignetteBuilder knitr

Author Peter K. Dunn [cre, aut]

Maintainer Peter K. Dunn <pdunn2@usc.edu.au>

Repository https://cran.r-universe.dev

Date/Publication 2026-05-19 09:20:02 UTC

RemoteUrl https://github.com/cran/tweedie

RemoteRef HEAD

RemoteSha 9d3ff488db5fbb17bf23a98f17ff23b5c58e3e2e

Contents

`tweedie-package` *Evaluation of Tweedie Exponential Family Models*

Description

This package provides maximum likelihood computations for Tweedie families, including the series expansion (Dunn and Smyth, 2005) and the Fourier inversion (Dunn and Smyth, 2008), and related methods.

Author(s)

Maintainer: Peter K. Dunn <pdunn2@usc.edu.au>

`dtweedie_inversion` *Fourier Inversion Evaluation for the Tweedie Probability Function*

Description

Evaluates the probability density function (PDF) for Tweedie distributions using Fourier inversion, for given values of the dependent variable `y`, the mean `mu`, dispersion `phi`, and power parameter `power`. *Not usually called by general users*, but can be used in the case of evaluation problems.

Usage

```
dtweedie_inversion(y, mu, phi, power, method = 3, verbose = FALSE,
                              details = FALSE, IGexact = TRUE)

dtweedie.inversion(y, power, mu, phi, method = 3, verbose, details)
```

Arguments

y	vector of quantiles.
mu	the mean parameter μ.
phi	the dispersion parameter ϕ.
power	scalar; the power parameter p.
method	the method to use; one of 1, 2, or 3 (the default).
verbose	logical; if TRUE, display some internal computation details. The default is FALSE.
details	logical; if TRUE, return a list with basic details of the integration. The default is FALSE.
IGexact	logical; if TRUE (the default), evaluate the inverse Gaussian distribution using the 'exact' values, otherwise uses inversion.

Value

A numeric vector of densities if details=FALSE; if details = TRUE, a list containing denisty (a vector of the values of the density), regions (a vector of the number of integration regions used),method (a vector giving the evaluation method used; see the Note below on the three methods), and exitstatus (a vector, where a 1 for any value means a computational problem or target relative accuracy not reached, for the corresponding observation).

Note

The 'exact' values for the inverse Gaussian distribution are not really exact, but evaluated using inverse normal distributions, for which very good numerical approximation are available in R. For special cases of p (i.e., $p = 0, 1, 2, 3$), where no inversion is needed, regions and method are set to NA for all values of y. For special cases of y for other values of p (i.e., $P(Y = 0)$), regions and method are set to NA.

The three methods are described in Dunn & Smyth (2008).

References

Dunn, P. K. and Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86. doi:10.1007/s1122200790396

Dunn, P. K. and Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86. doi:10.1007/s1122200790396

Examples

```
# Plot a Tweedie density
y <- seq(0.02, 4, length = 50)
fy <- dtweedie_inversion(y, mu = 1, phi = 1, power = 1.1)
plot(y, fy, type = "l", lwd = 2, ylab = "Density")
```

`dtweedie_saddle` *Tweedie densities evaluation using the saddlepoint approximation*

Description

Density function for the Tweedie EMDs using a saddlepoint approximation.

Usage

```
dtweedie_saddle(y, xi = NULL, mu, phi, eps = 1/6, power = NULL)

dtweedie.saddle(y, xi = NULL, mu, phi, eps = 1/6, power = NULL)
```

Arguments

`y`	vector of quantiles.
`xi`	scalar; the value of ξ such that the variance is var$[Y] = \phi\mu^{\xi}$. A synonym for `power`.
`mu`	vector of mean μ.
`phi`	vector of dispersion parameters ϕ.
`eps`	the offset in computing the variance function; the default is `eps=1/6` (as suggested by Nelder and Pregibon, 1987).
`power`	scalar; a synonym for ξ, the Tweedie index parameter.

Value

A numeric vector of densities.

References

Dunn, P. K. and Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86. doi:10.1007/s1122200790396

Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280. doi:10.1007/s112220054070y

Nelder, J. A. and Pregibon, D. (1987). An extended quasi-likelihood function *Biometrika*, **74**(2), 221–232. doi:10.1093/biomet/74.2.221

Examples

```
# Plot a Tweedie density
y <- seq(0.01, 4, length = 50)
fy <- dtweedie_saddle(y, power = 1.1, mu = 1, phi = 1)
plot(y, fy, type = "l", lwd = 2, ylab = "Density")
```

`dtweedie_series` *Series Evaluation for the Tweedie Probability Function*

Description

Evaluates the probability density function (PDF) for Tweedie distributions using an infinite series, for given values of the dependent variable `y`, the mean `mu`, dispersion `phi`, and power parameter `power`. *Not usually called by general users*, but can be used in the case of evaluation problems.

Usage

```
dtweedie_series(y, power, mu, phi, details = FALSE)

dtweedie.series(y, power, mu, phi)
```

Arguments

`y`	vector of quantiles.
`power`	scalar; the value of p such that the variance is $\text{var}[Y] = \phi\mu^p$.
`mu`	vector of mean μ.
`phi`	vector of dispersion parameters ϕ.
`details`	logical; if `TRUE`, returns the value of the distribution function and some details.

Value

A numeric vector of densities.

References

Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280. doi:10.1007/s112220054070y

Examples

```
# Plot a Tweedie density
y <- seq(0.01, 4, length = 50)
fy <- dtweedie_series(y, power = 1.1, mu = 1, phi = 1)
plot(y, fy, type = "l", lwd = 2, ylab = "Density")
```

`logLiktweedie` *Log-likelihood for Tweedie distributions*

Description

Evaluates the log-likelihood for a fitted Tweedie GLM.

Usage

```
logLiktweedie(glm.obj, dispersion = NULL)
```

Arguments

`glm.obj`	a fitted `glm` object, fitted using the `tweedie` family.
`dispersion`	the dispersion parameter, usually extracted from `glm.obj`; however, occasionally a specified value of the dispersion may be needed.

Details

The log-Likelihood is computed by evaluating the density function.

Value

The value of the computed log-likelihood.

Note

Evaluating the likelihood can be time consuming, so the function may take some time for large data sets.

References

Dunn, P. K. and Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86. doi:10.1007/s1122200790396

Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280. doi:10.1007/s112220054070y

Jorgensen, B. (1997). *Theory of Dispersion Models*. Chapman and Hall, London.

Sakamoto, Y., Ishiguro, M., and Kitagawa G. (1986). *Akaike Information Criterion Statistics*. D. Reidel Publishing Company.

See Also

`dtweedie`

Examples

```
# Fit a Tweedie density using  tweedie  family function from  statmod
pretend <- data.frame( y = stats::rgamma(20, shape = 1, rate = 1) )
fit <- glm(y ~ 1, data = pretend,
           family = statmod::tweedie(link.power = 0, var.power = 2.1))

# Compute the AIC
logLiktweedie(fit)
```

---

ptweedie_inversion *Fourier Inversion Evaluation for the Tweedie Distribution Function*

Description

Evaluates the distribution function (DF) for Tweedie distributions using Fourier inversion, for given values of the dependent variable y, the mean mu, dispersion phi, and power parameter power. *Not usually called by general users*, but can be in the case of evaluation problems.

Usage

```
ptweedie_inversion(q, mu, phi, power, verbose = FALSE, details = FALSE, IGexact = TRUE)

ptweedie.inversion(q, power, mu, phi, verbose, details)
```

Arguments

q	vector of quantiles.
mu	the mean parameter.
phi	the dispersion parameter.
power	the power parameter p.
verbose	logical; if TRUE, displays some internal computation details. The default is FALSE.
details	logical; if TRUE, returns the value of the distribution and some information about the integration. The default is FALSE.
IGexact	logical; if TRUE (the default), evaluate the inverse Gaussian distribution using the 'exact' values, otherwise uses inversion.

Value

If details = FALSE, a numeric vector of the distribution function values; if details = TRUE, a list containing CDF (a vector of the values of the distribution function), regions (a vector of the number of integration regions used), and exitstatus (a vector, where a 1 for any value means a computational problem or target relative accuracy not reached, for the corresponding observation).

For special cases of p (i.e., $p = 0, 1, 2, 3$), where no inversion is needed, regions is set to NA for all values of q. For special cases of q for other values of p (i.e., $P(Y = 0)$), regions is set to NA.

Note

The 'exact' values for the inverse Gaussian distribution are not really exact, but evaluated using inverse normal distributions, for which very good numerical approximation are available in R.

References

Dunn, P. K. and Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86. doi:10.1007/s1122200790396

Examples

```
# Plot a Tweedie distribution function
y <- seq(0.01, 4, length = 50)
Fy <- ptweedie_inversion(y, mu = 1, phi = 1, power = 1.1)
plot(y, Fy, type = "l", lwd = 2, ylab = "Distribution function")
```

`ptweedie_series` *Series Evaluation for the Tweedie Distribution Function*

Description

Evaluates the distribution function (DF) for Tweedie distributions with $1 < p < 2$ using an infinite series, for given values of the dependent variable `y`, the mean `mu`, dispersion `phi`, and power parameter `power`. *Not usually called by general users*, but can be in the case of evaluation problems.

Usage

```
ptweedie_series(q, power, mu, phi, verbose = FALSE, details = FALSE)

ptweedie.series(q, power, mu, phi, verbose = FALSE, details = FALSE)
```

Arguments

`q`	vector of quantiles.
`power`	the power parameter p.
`mu`	the mean parameter μ.
`phi`	the dispersion parameter ϕ.
`verbose`	logical; if `TRUE`, displays some internal computation details. The default is `FALSE`.
`details`	logical; if `TRUE`, returns the value of the distribution function and some details.

Value

A numeric vector of densities.

Note

The 'exact' values for the inverse Gaussian distribution are not really exact, but evaluated using inverse normal distributions, for which very good numerical approximation are available in R.

References

Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280. doi:10.1007/s112220054070y

Examples

```
# Plot a Tweedie distribution function
y <- seq(0.01, 4, length = 50)
Fy <- ptweedie_series(y, power = 1.1, mu = 1, phi = 1)
plot(y, Fy, type = "l", lwd = 2, ylab = "Distribution function")
```

`Tweedie` *Tweedie distributions*

Description

Density, distribution function, quantile function and random generation for the the Tweedie family of distributions, with mean `mu`, dispersion parameter `phi` and variance power `power` (or `xi`, a synonym for `power`).

Usage

```
dtweedie(y, xi = NULL, mu, phi, power = NULL, verbose = FALSE)

ptweedie(q, xi = NULL, mu, phi, power = NULL, verbose = FALSE)

qtweedie(p, xi = NULL, mu, phi, power = NULL)

rtweedie(n, xi = NULL, mu, phi, power = NULL)

ptweedie(q, xi = NULL, mu, phi, power = NULL, verbose = FALSE)

qtweedie(p, xi = NULL, mu, phi, power = NULL)

rtweedie(n, xi = NULL, mu, phi, power = NULL)
```

Arguments

`y`	vector of quantiles.
`xi`	scalar; the value of ξ such that the variance is $\text{var}[Y] = \phi\mu^\xi$. A synonym for `power`.

mu	vector of mean μ.
phi	vector of dispersion parameters ϕ.
power	scalar; a synonym for ξ, the Tweedie index parameter.
verbose	logical; if TRUE, some details of the algorithms used is shown. The default is FALSE.
q	vector of quantiles.
p	vector of probabilities.
n	number of observations.

Details

The Tweedie EDMs belong to the class of exponential dispersion models (EDMs), known for their role in generalized linear models (GLMs). The Tweedie distributions are the EDMs with a variance of the form $\text{var}[Y] = \phi\mu^p$ where $p \geq 1$. *This function only evaluates for* $p \geq 1$.

Special cases are the Poisson ($p = 1$ with $\phi = 1$), gamma ($p = 2$), and inverse Gaussian ($p = 3$) distributions. Evaluation is difficult for p outside of $p = 0, 1, 2, 3$. This function uses one of two primary methods, depending on the combination of parameters:

1. Evaluation of an infinite series (`dtweedie_series`).
2. Interpolation from stored values computed via a Fourier inversion technique (`dtweedie_inversion`).

This function employs a two-dimensional interpolation procedure to compute the density for some parts of the parameter space from previously computed values (interpolation) and uses the series solution for others.

When $1 < p < 2$, the density function include a positive probably for $Y = 0$.

Value

`dtweedie` gives the density, `ptweedie` gives the distribution function, `qtweedie` gives the quantile function, and `rtweedie` generates random deviates.

The length of the result is determined by `n` for `rtweedie`, and by the length of `mu` for other functions.

Note

`dtweedie` and `ptweedie` are the only functions generally to be called by users. Consequently, all checks on the function inputs are performed in these functions.

References

Dunn, P. K. and Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86. doi:10.1007/s1122200790396

Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280. doi:10.1007/s112220054070y

Jorgensen, B. (1997). *Theory of Dispersion Models*. Chapman and Hall, London.

See Also

dtweedie_series, dtweedie_inversion, ptweedie_series, ptweedie_inversion, dtweedie_saddle, tweedie_lambda

Examples

```
# Compute a Tweedie density
power <- 1.1
mu <- 1
phi <- 1
y <- seq(0, 5, by = 0.5)
dtweedie(y, power = power, mu = mu, phi = phi)

# Compare to the saddlepoint density
dtweedie_saddle(y = y, power = power, mu = mu, phi = phi)

# The DF:
ptweedie(y, power = power, mu = mu, phi = phi)
```

---

tweedie_AIC *AIC for Tweedie Glms*

Description

Evaluates the AIC for a fitted Tweedie GLM. The Tweedie family of distributions belong to the class of exponential dispersion models (EDMs), famous for their role in generalized linear models. The Tweedie distributions are the EDMs with a variance of the form $\text{var}[Y] = \phi\mu^p$ where $p \geq 1$. *This function only evaluates for* $p \geq 1$.

Usage

```
tweedie_AIC(glm.obj, dispersion = NULL, k = 2, verbose = TRUE)

AICtweedie(glm.obj, dispersion = NULL, k = 2, verbose = TRUE)
```

Arguments

glm.obj	a fitted glm object, fitted using the tweedie family.
dispersion	the dispersion parameter, usually extracted from glm.obj; however, occasionally a specified value of the dispersion may be needed.
k	the AIC penalty; k = 2 (the default) produces the AIC.
verbose	logical; if TRUE, display details of the internal process. The default is FALSE.

Details

The AIC is computed by evaluating the density function.

Value

The value of the computed AIC.

Note

Evaluating the likelihood can be time consuming, so the function may take some time for large data sets.

References

Dunn, P. K. and Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86. doi:10.1007/s1122200790396

Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280. doi:10.1007/s112220054070y

Jorgensen, B. (1997). *Theory of Dispersion Models*. Chapman and Hall, London.

Sakamoto, Y., Ishiguro, M., and Kitagawa G. (1986). *Akaike Information Criterion Statistics*. D. Reidel Publishing Company.

See Also

dtweedie

Examples

```
# Fit a Tweedie density using  tweedie  family function from  statmod
pretend <- data.frame( y = stats::rgamma(20, shape = 1, rate = 1) )
fit <- glm(y ~ 1, data = pretend,
           family = statmod::tweedie(link.power = 0, var.power = 2.1))

# Compute the AIC
tweedie_AIC(fit)
```

---

tweedie_convert *Tweedie Distribution: Convert Between Parameter Formats*

Description

Converts from the fitted GLM parameters p, μ and ϕ and the corresponding underlying Poisson and gamma parameters (when $1 < p < 2$).

Usage

```
tweedie_convert(xi = NULL, mu, phi, power = NULL)

tweedie.convert(xi = NULL, mu, phi, power = NULL)
```

Arguments

`xi`	a synonym for `power`.
`mu`	the mean parameter μ.
`phi`	the dispersion parameter ϕ.
`power`	the power parameter p; a synonym for ξ.

Value

a list of the parameters of the parameters of the corresponding underlying Poisson and gamma densities: `poisson.lambda` (λ from the underlying Poisson distribution), `gamma.shape`, `gamma.scale` (the shape and scale parameters from the underlying gamma distribution), `p0` (the probability that $Y = 0$), `gamma.mean` and `gamma.phi` (the gamma mean and dispersion parameter values)

Examples

```
### Fit a Tweedie density
pretend <- data.frame( y = rgamma(20, shape = 1, rate = 1) )
fit <- glm(y ~ 1, data = pretend,
           family = statmod::tweedie(link.power = 0, var.power = 1.4))

# Convert parameters
tweedie_convert(mu = fitted(fit, type="response"), phi = 1, power = 1.4)
```

`tweedie_dev` *Unit Deviance for a Tweedie Distribution*

Description

Computes the unit deviance for Tweedie distributions.

Usage

```
tweedie_dev(y, mu, power)

tweedie.dev(y, mu, power)
```

Arguments

`y`	vector of quantiles.
`mu`	the mean parameter μ.
`power`	the power parameter p.

Value

A numeric vector containing the unit deviance.

References

Jorgensen, B. (1997). *Theory of Dispersion Models*. Chapman and Hall, London.

Examples

```
# Unit deviance is not symmetric in general:
round( tweedie_dev(0:6, mu = 3, power = 1.1), 3)
```

---

tweedie_integrand — *Display Integrand Information for Tweedie Fourier inversion*

Description

Plots the integrand for Fourier inversion and the real and imaginary parts separately.

Usage

```
tweedie_integrand(y, power, mu, phi, t = seq(0, 5, length = 200),
                  type = "PDF", whichPlots = 1:4,
                  plot_args  = list())
```

Arguments

y	vector of quantiles.
power	a synonym for ξ; the Tweedie power-index on the variance.
mu	the mean parameter μ.
phi	the dispersion parameter ϕ.
t	the values of the variable over which to integrate; the default is `t = seq(0, 5, length = 200)`.
type	either `"PDF"` (the default) for the (probability) density function, or `"CDF"` for the (cumulative) distribution function.
whichPlots	which combination of the four plots (described below) are produced; by default, all four are produced (i.e., `whichPlots = 1:4`).
plot_args	A named list of arguments controlling the main plot. These are passed to `plot`.

Details

The Tweedie family of distributions belong to the class of exponential dispersion models (EDMs), famous for their role in generalized linear models. The Tweedie distributions are the EDMs with a variance of the form $\text{var}[Y] = \phi\mu^p$ where p is greater than or equal to one, or less than or equal to zero.

This function only evaluates for p greater than or equal to one.

Special cases include the normal ($p = 0$), Poisson ($p = 1$ with $\phi = 1$), gamma ($p = 2$) and inverse Gaussian ($p = 3$) distributions. For other values of `power`, the distributions are still defined but cannot be written in closed form, and hence evaluation is very difficult.

When $1 < p < 2$, the distribution are continuous for Y greater than zero, with a positive mass at $Y = 0$. For $p > 2$, the distributions are continuous for Y greater than zero.

This function displays the integrand that is evaluated for computing the Fourier inversion, for the PDF or CDF.

Value

A list containing the real and imaginary parts of $k(t)$, `Real` and `Imag` respectively, plus the values of the integrand as `IG`. The main purpose of the function is the side-effect of producing a 2×2 grid of plots. The first is the imaginary parts of $k(t)$. The second is $\cos \Im k(t)$ (for `type=="PDF"`) or $\sin \Im k(t)$ (for `type=="CDF"`). The third is the real part of $\Re k(t)$ The fourth is the integrand, with the envelope shown as a dashed line.

Author(s)

Peter Dunn (`<pdunn2@usc.edu.au>`)

References

Dunn, P. K. and Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86. doi:10.1007/s1122200790396

Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280. doi:10.1007/s112220054070y

Dunn, Peter K and Smyth, Gordon K (2001). Tweedie family densities: methods of evaluation. *Proceedings of the 16th International Workshop on Statistical Modelling*, Odense, Denmark, 2–6 July

Jorgensen, B. (1987). Exponential dispersion models. *Journal of the Royal Statistical Society*, B, **49**, 127–162.

Jorgensen, B. (1997). *Theory of Dispersion Models*. Chapman and Hall, London.

Tweedie, M. C. K. (1984). An index which distinguishes between some important exponential families. *Statistics: Applications and New Directions. Proceedings of the Indian Statistical Institute Golden Jubilee International Conference* (Eds. J. K. Ghosh and J. Roy), pp. 579-604. Calcutta: Indian Statistical Institute.

See Also

`dtweedie`

Examples

```
tweedie_integrand(2, power = 3, mu = 1, phi = 1,
                  plot_args = list(lwd = 2))
```

tweedie_lambda *The Probability of Observing a Zero Value for a Tweedie Density*

Description

The probability that the variable takes the value of zero.

Usage

```
tweedie_lambda(mu, phi, power)
```

Arguments

`mu`	the mean parameter μ.
`phi`	the dispersion parameter ϕ.
`power`	the power parameter p (sometimes denoted ξ).

Value

The value of λ when $1 < p < 2$ such that $P(Y = 0) = \exp(-\lambda)$. When $p > 2$, a vector of zeros is returned.

References

Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280. doi:10.1007/s112220054070y

Examples

```
lambda <- tweedie_lambda(mu = 1:3, phi = 1, power = 1.1)
exp( -lambda)

# When p > 2, there is zero probability that Y = 0:
lambda  <- tweedie_lambda(mu = 1, phi = 1, power = 3.1)
```

tweedie_plot *Plot Tweedie Models*

Description

This function produced a plot of the specified Tweedie distribution.

Usage

```
tweedie_plot(y, xi = NULL, mu, phi, type = "pdf", power = NULL,
                    add = FALSE,
                   plot_args = list(), point_args = list(), line_args = list())

tweedie.plot(
  y,
  xi = NULL,
  mu,
  phi,
  type = "pdf",
  power = NULL,
  add = FALSE,
  ...
)
```

Arguments

`y`	the values for y in the plot.
`xi`	a synonym for `power`.
`mu`	the mean of the distribution μ.
`phi`	the dispersion parameter ϕ.
`type`	the type of plot, either `pdf` (the default) or `cdf`.
`power`	the variance power p.
`add`	logical; if `TRUE`, the plot is added to the current plot; if `FALSE` (the default) the plot is produced on a fresh plot.
`plot_args`	A named list of arguments controlling the main plot. These are passed to `plot`.
`point_args`	A named list of graphical parameters for plotted points. These are passed to `points`.
`line_args`	A named list of graphical parameters for plotted lines. These are passed to `lines`.
`...`	Additional graphical arguments, passed to `plot_args` for backward compatibility. Deprecated; use `plot_args` in `tweedie_plot` instead.

Details

If $1 < p < 2$, the mass at $Y = 0$ is automatically added.

Examples

```
y <- seq(0, 4, length = 50)
tweedie_plot(y, power = 1.1, mu = 1, phi = 1,
             line_args = list(lwd = 2))
```

tweedie_profile *Profile Likelihood Estimate of Tweedie Variance Index Parameter*

Description

This function profiles the (log-)likelihood over a vector of Tweedie power-index parameter (denoted p or ξ) to find the maximum likelihood estimate (MLE) of the index parameter p (or equivalently ξ).

Usage

```
tweedie_profile(formula, p.vec = NULL, xi.vec = NULL, link.power = 0,
  data, weights = 1, offset = 0, fit.glm = FALSE, do.smooth = TRUE,
  do.plot = FALSE, do.ci = do.smooth, eps = 1/6,
  control = list( epsilon = 1e-09, maxit = stats::glm.control()$maxit,
  trace = glm.control()$trace ),
  do.points = do.plot, method = "inversion", conf.level = 0.95,
  phi.method = ifelse(method == "saddlepoint", "saddlepoint", "mle"),
  verbose = FALSE, add0 = FALSE,
  plot_args = list(), point_args = list(), line_args = list())

tweedie.profile(
  formula,
  p.vec = NULL,
  xi.vec = NULL,
  link.power = 0,
  data,
  weights = 1,
  offset = 0,
  fit.glm = FALSE,
  do.smooth = TRUE,
  do.plot = FALSE,
  do.ci = do.smooth,
  eps = 1/6,
  control = list(epsilon = 1e-09, maxit = stats::glm.control()$maxit, trace =
    glm.control()$trace),
  do.points = do.plot,
  method = "inversion",
  conf.level = 0.95,
  phi.method = ifelse(method == "saddlepoint", "saddlepoint", "mle"),
  verbose = FALSE,
  add0 = FALSE
)
```

Arguments

formula — a formula expression as for other regression models and generalized linear models, of the form `response ~ predictors`.

p.vec	a vector of p values for consideration. The values must all be larger than one. If the response has zeros, values must be $1 < p < 2$. If NULL (default), p.vec is set automatically.
xi.vec	a synonym for p.vec, as some authors use the ξ notation.
link.power	the power link function to use in the tweedie GLM family. These link functions $g(\cdot)$ are of the form $g(\eta) = \eta^{link.power}$, where link.power = 0 (default) refers to the logarithm link function.
data	an optional data frame, list or environment containing the variables.
weights	an optional vector of weights to be used in the fitting process.
offset	an *a priori* known component included in the linear predictor. See model.offset.
fit.glm	logical; if TRUE, the Tweedie GLM is fitted using the value of p found by the profiling function. The default is FALSE.
do.smooth	logical; if TRUE (default), a spline is fitted to the data to smooth the profile likelihood plot. **Note** that p.vec must contain *at least five points* for smoothing.
do.plot	logical; if TRUE, a plot of the profile likelihood is produced. The default is FALSE.
do.ci	logical; if TRUE, the nominal 100*conf.level is computed. Defaults to the value of do.smooth. Confidence intervals are only computed if do.smooth = TRUE.
eps	the offset in computing the variance function. Default is 1/6 (as recommended by Nelder and Pregibon, 1987). eps is ignored unless method = "saddlepoint".
control	a list of parameters for controlling the fitting process;
do.points	logical; if TRUE, the points used to compute the likelihood as given by p.vec (or equivalently, xi.vec) are explicitly shown by points. The defaults is the value of do.plot.
method	the method of evaluation; one of saddlepoint, interpolation, series or inversion (the default).
conf.level	the level of confidence for the confidence intervals; the default is 0.95 (for 95% confidence intervals).
phi.method	the method used to estimate ϕ; one of saddlepoint, mle (the default).
verbose	logical; if TRUE, some details of the calculations are shown. The default is FALSE.
add0	logical; if TRUE, adds $P(Y = 0)$ to the plot. The default is FALSE.
plot_args	A named list of arguments controlling the main plot. These are passed to plot.
point_args	A named list of graphical parameters for plotted points. These are passed to points.
line_args	A named list of graphical parameters for plotted lines. These are passed to lines.

Details

For each value in p.vec, the function computes an estimate of ϕ and then computes the value of the log-likelihood for these parameters. The plot of the log-likelihood against p.vec allows the maximum likelihood value of p to be found. Once p is found, the distribution within the class of Tweedie distributions is identified.

Note

The estimates of p and ϕ are printed invisibly. If the response variable has any exact zeros, the values in `p.vec` must all be between one and two.

The function can be temperamental (for theoretical reasons involved in numerically computing the density; see Dunn and Smyth (2005)) and may be very slow or fail. One solution is to change the method. The default is `method = "inversion"`; then try `"series"`, `"interpolation"`, and `"saddlepoint"` in that order. Note that `method = "saddlepoint"` is an approximate method only.

It is recommended that for the first use with a data set, use `p.vec` with only a small number of values and set `do.smooth = FALSE`, `do.ci = FALSE`. If this is successful, a larger vector `p.vec` and smoothing can be used.

References

Dunn, P. K. and Smyth, G. K. (2018). Generalized linear models with examples in R. Springer. doi:10.1007/9781441901187

Examples

```
data(rock)
out <- tweedie_profile(perm~1, data=rock, do.plot=FALSE,
                       xi.vec=seq(1.5, 2.75, length=11),
                       line_arg = list(lwd = 2))

# The estimate for the variance power index (p, or xi) is:
out$p.max
```

Index