

# Package ‘tweedie’

November 17, 2009

**Version** 2.0.2

**Date** 2009-17-11

**Title** Tweedie exponential family models

**Author** Peter K Dunn <pdunn2@usc.edu.au>

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

**Depends** R (>= 1.0), statmod(>= 1.0.8)

**Suggests** fBasics

**Description** Maximum likelihood computations for Tweedie families.

**License** GPL (>= 2)

**URL** <http://www.r-project.org/package=tweedie>

**Repository** CRAN

**Date/Publication** 2009-11-17 07:53:14

## R topics documented:

tweedie-package	2
AICtweedie	3
dtweedie.dldphi	4
dtweedie.saddle	6
Tweedie	8
Tweedie internals	11
tweedie.dev	12
tweedie.plot	14
tweedie.profile	16

<b>Index</b>	<b>20</b>
--------------	-----------

---

tweedie-package      *Tweedie Distributions*

---

## Description

Functions for computing and fitting the Tweedie family of distributions

## Details

Package: tweedie  
Type: Package  
Version: 2.0.2  
Date: 2009-17-11  
License: GPL (>=2)

## Author(s)

Peter K Dunn

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

## References

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

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

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.

## Examples

```
# Generate random numbers
set.seed(314)
y <- rtweedie( 500, p=1.5, mu=1, phi=1)
# With index p between 1 and 2, this produces continuous
```

```
# data with exact zeros
x <- rnorm( length(y), 0, 1) # Unrelated predictor

# With exact zeros, index p must be between 1 and 2

# Fit the tweedie distribution; expect p about 1.5
out <- tweedie.profile( y~1, p.vec=seq(1.1, 1.9, length=9), do.plot=TRUE)

out$p.max

# Plot this distribution
tweedie.plot( seq(0, max(y), length=1000), mu=mean(y), p=out$p.max, phi=out$phi.max)

# Fit the glm
require(statmod) # Provides tweedie family functions
summary(glm( y ~ x, family=tweedie(var.power=out$p.max, link.power=0) ))
```

---

AICtweedie

*Tweedie Distributions*

---

### Description

The AIC for Tweedie models

### Usage

```
AICtweedie( glm.obj, k = 2)
```

### Arguments

glm.obj	a fitted Tweedie glm object
k	numeric: the penalty per parameter to be used; the default is $k = 2$

### Details

See [AIC](#) for more details on the AIC; see `link{dtweedie}` for ore details on computing the Tweedie densities

### Value

Returns a numeric value with the corresponding AIC (or BIC, depending on  $k$ )

### Note

Computing the AIC can take a long time!

### Author(s)

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

## References

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

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

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

[AIC](#)

## Examples

```
library(statmod) # Needed to use tweedie family object

### Generate some fictitious data
test.data <- rgamma(n=200, scale=1, shape=1)

### Fit a Tweedie glm and find the AIC
m1 <- glm( test.data~1, family=tweedie(link.power=0, var.power=2) )

### A Tweedie glm with p=2 is equivalent to a gamma glm:
m2 <- glm( test.data~1, family=Gamma(link=log) )

### The models are equivalent, so the AIC should be the same:
AICtweedie(m1)
AIC(m2)
```

---

dtweedie.dldphi      *Tweedie Distributions*

---

## Description

Derivatives of the log-likelihood with respect to  $\phi$

## Usage

```
dtweedie.dldphi(phi, mu, power, y )
dtweedie.dldphi.saddle(phi, mu, power, y )
```

**Arguments**

y	vector of quantiles
mu	the mean
phi	the dispersion
power	the value of $p$ such that the variance is $\text{var}[Y] = \phi\mu^p$

**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.

**Value**

the value of the derivative  $\partial\ell/\partial\phi$  where  $\ell$  is the log-likelihood for the specified Tweedie distribution. `dtweedie.dldphi.saddle` uses the saddlepoint approximation to determine the derivative; `dtweedie.dldphi` uses an infinite series expansion.

**Author(s)**

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

**References**

- Dunn, P. K. & Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86.
- Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280.
- 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.
- Sidi, Avram (1982). The numerical evaluation of very oscillatory infinite integrals by extrapolation. *Mathematics of Computation* **38**(158), 517–529.
- Sidi, Avram (1988). A user-friendly extrapolation method for oscillatory infinite integrals. *Mathematics of Computation* **51**(183), 249–266.
- 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.saddle](#) [dtweedie](#) [tweedie.profile](#) [tweedie](#)

**Examples**

```
### Plot dl/dphi against candidate values of phi
power <- 2
mu <- 1
phi <- seq(2, 8, by=0.1)

set.seed(10000) # For reproducibility
y <- rtweedie( 100, mu=mu, power=power, phi=3)
# So we expect the maximum to occur at phi=3

dldphi <- dldphi.saddle <- array( dim=length(phi))

for (i in (1:length(phi))) {
  dldphi[i] <- dtweedie.dldphi( y=y, power=power, mu=mu, phi=phi[i])
  dldphi.saddle[i] <- dtweedie.dldphi.saddle( y=y, power=power, mu=mu, phi=phi[i])
}

plot( dldphi ~ phi, lwd=2, type="l",
      ylab=expression(phi), xlab=expression(paste("dl / d",phi) ) )
lines( dldphi.saddle ~ phi, lwd=2, col=2, lty=2)
legend( "bottomright", lwd=c(2,2), lty=c(1,2), col=c(1,2),
       legend=c("'Exact' (using series)", "Saddlepoint") )

# Neither are very good in this case!
```

---

dtweedie.saddle      *Tweedie Distributions (saddlepoint approximation)*

---

**Description**

Saddlepoint density for the Tweedie distributions

**Usage**

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

**Arguments**

y	the vector of responses
power	the value of $p$ such that the variance is $\text{var}[Y] = \phi\mu^p$
mu	the mean
phi	the dispersion
eps	the offset in computing the variance function. The default is $\text{eps}=1/6$ (as suggested by Nelder and Pregibon, 1987).

## 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 approximates the density using the saddlepoint approximation defined by Nelder and Pregibon (1987).

## Value

saddlepoint (approximate) density for the given Tweedie distribution with parameters `mu`, `phi` and `power`.

## Author(s)

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

## References

- Daniels, H. E. (1954). Saddlepoint approximations in statistics. *Annals of Mathematical Statistics*, **25**(4), 631–650.
- Daniels, H. E. (1980). Exact saddlepoint approximations. *Biometrika*, **67**, 59–63.
- Dunn, P. K. & Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86.
- 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
- Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280.
- 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.
- Nelder, J. A. and Pregibon, D. (1987). An extended quasi-likelihood function. *Biometrika*, **74**(2), 221–232.
- 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**

```

p <- 2.5
mu <- 1
phi <- 1
y <- seq(0, 10, length=100)
fy <- dtweedie( y=y, power=p, mu=mu, phi=phi)
plot(y, fy, type="l")
# Compare to the saddlepoint density
f.saddle <- dtweedie.saddle( y=y, power=p, mu=mu, phi=phi)
lines( y, f.saddle, col=2 )

```

Tweedie

*Tweedie Distributions***Description**

Density, distribution function, quantile function and random generation for the Tweedie family of distributions

**Usage**

```

dtweedie(y, power, mu, phi)
dtweedie.series(y, power, mu, phi)
dtweedie.inversion(y, power, mu, phi, exact=TRUE, method)
dtweedie.stable(y, power, mu, phi)
ptweedie(q, power, mu, phi)
ptweedie.series(q, power, mu, phi)
qtweedie(p, power, mu, phi)
rtweedie(n, power, mu, phi)

```

**Arguments**

<code>y, q</code>	vector of quantiles
<code>p</code>	vector of probabilities
<code>n</code>	the number of observations
<code>power</code>	the value of $p$ such that the variance is $\text{var}[Y] = \phi\mu^p$
<code>mu</code>	the mean
<code>phi</code>	the dispersion
<code>exact</code>	logical flag; if TRUE (the default), exact zeros are used with the $W$ -algorithm of Sidi (1982); if FALSE, approximate (asymptotic) zeros are used in place of exact zeros. Using asymptotic zeros requires less computation but is often less accurate; using exact zeros can be slower but generally improves accuracy.

method either 1, 2 or 3, determining which of three methods to use to compute the density using the inversion method. If `method` is `NULL` (the default), the optimal method (in terms of relative accuracy) is used, element-by-element of `y`. See the Note in the Details section below

## 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 evaluates the density or cumulative probability using one of two methods, depending on the combination of parameters. One method is the evaluation of an infinite series. The second interpolates some stored values computed from a Fourier inversion technique.

The function `dtweedie.inversion` evaluates the density using a Fourier series technique; `ptweedie.inversion` does likewise for the cumulative probabilities. The actual code is contained in an external FORTRAN program. Different code is used for  $p > 2$  and for  $1 < p < 2$ .

The function `dtweedie.series` evaluates the density using a series expansion; a different series expansion is used for  $p > 2$  and for  $1 < p < 2$ . The function `ptweedie.series` does likewise for the cumulative probabilities but only for  $1 < p < 2$ .

The function `dtweedie.stable` exploits the link between the stable distribution (Nolan, 1997) and Tweedie distributions, as discussed in Jorgensen, Chapter 4. These are computed using Nolan's algorithm as implemented in the `fBasics` package (which is therefore required to use the `dtweedie.stable` function).

The function `dtweedie` uses a two-dimensional interpolation procedure to compute the density for some parts of the parameter space from previously computed values found from the series or the inversion. For other parts of the parameter space, the series solution is found.

`ptweedie` returns either the computed series solution or inversion solution.

## Value

density (`dtweedie`), probability (`ptweedie`), quantile (`qtweedie`) or random sample (`rtweedie`) for the given Tweedie distribution with parameters `mu`, `phi` and `power`.

## Note

The methods changed from version 1.4 to 1.5 (methods 1 and 2 swapped). The methods are defined in Dunn and Smyth (2008).

## Author(s)

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

## References

- Dunn, P. K. & Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86.
- Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280.
- 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.
- Nolan, John P (1997). Numerical calculation of stable densities and distribution functions. *Communication in Statistics—Stochastic models*, **13**(4). 759–774.
- Sidi, Avram (1982). The numerical evaluation of very oscillatory infinite integrals by extrapolation. *Mathematics of Computation* **38**(158), 517–529.
- Sidi, Avram (1988). A user-friendly extrapolation method for oscillatory infinite integrals. *Mathematics of Computation* **51**(183), 249–266.
- 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.saddle](#)

## Examples

```
### Plot a Tweedie density
power <- 2.5
mu <- 1
phi <- 1
y <- seq(0, 6, length=500)
fy <- dtweedie( y=y, power=power, mu=mu, phi=phi)
plot(y, fy, type="l", lwd=2, ylab="Density")
# Compare to the saddlepoint density
f.saddle <- dtweedie.saddle( y=y, power=power, mu=mu, phi=phi)
lines( y, f.saddle, col=2 )
legend("topright", col=c(1,2), lwd=c(2,1),
      legend=c("Actual", "Saddlepoint") )

### A histogram of Tweedie random numbers
hist( rtweedie( 1000, power=1.2, mu=1, phi=1) )

### An example of the multimodal feature of the Tweedie
### family with power near 1 (from Dunn and Smyth, 2005).
y <- seq(0.001,2,len=1000)
```

```

mu <- 1
phi <- 0.1
p <- 1.02
f1 <- dtweedie(y,mu=mu,phi=phi,power=p)
plot(y, f1, type="l", xlab="y", ylab="Density")
p <- 1.05
f2<- dtweedie(y,mu=mu,phi=phi,power=p)
lines(y,f2, col=2)

### Compare series and saddlepoint methods
y <- seq(0.001,2,len=1000)
mu <- 1
phi <- 0.1
p <- 1.02
f.series <- dtweedie.series( y,mu=mu,phi=phi,power=p )
f.saddle <- dtweedie.saddle( y,mu=mu,phi=phi,power=p )

f.all <- c( f.series, f.saddle )
plot( range(f.all) ~ range( y ), xlab="y", ylab="Density",
      type="n")
lines( f.series ~ y, lty=1, col=1)
lines( f.saddle ~ y, lty=3, col=3)

legend("topright", lty=c(1,3), col=c(1,3),
      legend=c("Series","Saddlepoint") )

```

---

Tweedie internals *Tweedie internal function*

---

## Description

Internal tweedie functions.

## Usage

```

dtweedie.dlogfdphi(y, mu, phi, power)
dtweedie.logl(phi, y, mu, power)
dtweedie.logl.saddle( phi, power, y, mu, eps=0)
dtweedie.logv.bigp( y, phi, power)
dtweedie.logw.smallp(y, phi, power)
dtweedie.interp(grid, nx, np, xix.lo, xix.hi,p.lo, p.hi, power, xix)
dtweedie.jw.smallp(y, phi, power )
dtweedie.kv.bigp(y, phi, power)
dtweedie.series.bigp(power, y, mu, phi)
dtweedie.series.smallp(power, y, mu, phi)
stored.grids(power)

```

**Arguments**

<code>y</code>	the vector of responses
<code>power</code>	the value of $p$ such that the variance is $\text{var}[Y] = \phi\mu^p$
<code>mu</code>	the mean
<code>phi</code>	the dispersion
<code>grid</code>	the interpolation grid necessary for the given value of $p$
<code>nx</code>	the number of interpolation points in the $\xi$ dimension
<code>np</code>	the number of interpolation points in the $p$ dimension
<code>xix.lo</code>	the lower value of the transformed $\xi$ value used in the interpolation grid. (Note that the value of $\xi$ is from 0 to $\infty$ , and is transformed such that it is on the range 0 to 1.)
<code>xix.hi</code>	the higher value of the transformed $\xi$ value used in the interpolation grid.
<code>p.lo</code>	the lower value of $p$ value used in the interpolation grid.
<code>p.hi</code>	the higher value of $p$ value used in the interpolation grid.
<code>xix</code>	the value of the transformed $\xi$ at which a value is sought.
<code>eps</code>	the offset in computing the variance function in the saddlepoint approximation. The default is $\text{eps}=1/6$ (as suggested by Nelder and Pregibon, 1987).

**Details**

These are not to be called by the user.

**Author(s)**

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

**References**

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

---

tweedie.dev

*Tweedie Distributions: the deviance function*

---

**Description**

The deviance function for the Tweedie family of distributions

**Usage**

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

**Arguments**

y	vector of quantiles (which can be zero if $1 < p < 2$ )
mu	the mean
power	the value of $p$ such that the variance is $\text{var}[Y] = \phi\mu^p$

**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.

The deviance is defined by `deviance` as “up to a constant, minus twice the maximized log-likelihood. Where sensible, the constant is chosen so that a saturated model has deviance zero.”

**Value**

the value of the deviance for the given Tweedie distribution with parameters `mu`, `phi` and `power`.

**Author(s)**

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

**References**

- Dunn, P. K. & Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86.
- Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280.
- 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.
- Sidi, Avram (1982). The numerical evaluation of very oscillatory infinite integrals by extrapolation. *Mathematics of Computation* **38**(158), 517–529.
- Sidi, Avram (1988). A user-friendly extrapolation method for oscillatory infinite integrals. *Mathematics of Computation* **51**(183), 249–266.
- 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](#) [dtweedie.saddle](#), [tweedie](#), [deviance](#), [glm](#)

**Examples**

```
### Plot a Tweedie deviance function when 1<p<2
mu <- 1

y <- seq(0, 6, length=100)

dev1 <- tweedie.dev( y=y, mu=mu, power=1.1)
dev2 <- tweedie.dev( y=y, mu=mu, power=1.5)
dev3 <- tweedie.dev( y=y, mu=mu, power=1.9)

plot(range(y), range( c(dev1, dev2, dev3)),
      type="n", lwd=2, ylab="Deviance", xlab=expression(italic(y)) )

lines( y, dev1, lty=1, col=1, lwd=2 )
lines( y, dev2, lty=2, col=2, lwd=2 )
lines( y, dev3, lty=3, col=3, lwd=2 )

legend("top", col=c(1,2,3), lwd=c(2,2,2), lty=c(1,2,3),
       legend=c("p=1.1", "p=1.5", "p=1.9") )

### Plot a Tweedie deviance function when p>2
mu <- 1

y <- seq(0.1, 6, length=100)

dev1 <- tweedie.dev( y=y, mu=mu, power=2) # Gamma
dev2 <- tweedie.dev( y=y, mu=mu, power=3) # Inverse Gaussian
dev3 <- tweedie.dev( y=y, mu=mu, power=4)

plot(range(y), range( c(dev1, dev2, dev3)),
      type="n", lwd=2, ylab="Deviance", xlab=expression(italic(y)) )

lines( y, dev1, lty=1, col=1, lwd=2 )
lines( y, dev2, lty=2, col=2, lwd=2 )
lines( y, dev3, lty=3, col=3, lwd=2 )

legend("top", col=c(1,2,3), lwd=c(2,2,2), lty=c(1,2,3),
       legend=c("p=2 (gamma)", "p=3 (inverse Gaussian)", "p=4") )
```

**Description**

Plotting Tweedie density and distribution functions

**Usage**

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

**Arguments**

y	vector of values at which to evaluate and plot
power	the value of $p$ such that the variance is $\text{var}[Y] = \phi\mu^p$
mu	the mean
phi	the dispersion
type	what to plot: pdf (the default) means the probability function, or cdf, the cumulative distribution function
add	if TRUE, the plot is added to the current device; if FALSE (the default), a new plot is produced
...	Arguments to be passed to the plotting method

**Details**

For details, see [dtweedie](#)

**Value**

this function is usually called for side-effect of producing a plot of the specified Tweedie distribution, properly plotting the exact zero that occurs at  $y = 0$  when  $1 < p < 2$ . However, it also produces a list with the computed density at the given points, with components `y` and `x` respectively, such that `plot(y~x)` approximately reproduces the plot.

**Author(s)**

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

**References**

- Dunn, P. K. & Smyth, G. K. (2008). Evaluation of Tweedie exponential dispersion model densities by Fourier inversion. *Statistics and Computing*, **18**, 73–86.
- Dunn, Peter K and Smyth, Gordon K (2005). Series evaluation of Tweedie exponential dispersion model densities *Statistics and Computing*, **15**(4). 267–280.
- 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.

Nolan, John P (1997). Numerical calculation of stable densities and distribution functions. *Communication in Statistics—Stochastic models*, **13**(4). 759–774.

Sidi, Avram (1982). The numerical evaluation of very oscillatory infinite integrals by extrapolation. *Mathematics of Computation* **38**(158), 517–529.

Sidi, Avram (1988). A user-friendly extrapolation method for oscillatory infinite integrals. *Mathematics of Computation* **51**(183), 249–266.

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

```
### Plot a Tweedie density with  $1 < p < 2$ 
yy <- seq(0,5,length=100)
tweedie.plot( power=1.7, mu=1, phi=1, y=yy, lwd=2)
tweedie.plot( power=1.2, mu=1, phi=1, y=yy, add=TRUE, lwd=2, col="red")
legend("topright",lwd=c(2,2), col=c("black","red"), pch=c(19,19),
      legend=c("p=1.7","p=1.2") )

### Plot distribution functions
tweedie.plot( power=1.05, mu=1, phi=1, y=yy, lwd=2, type="cdf", ylim=c(0,1))
tweedie.plot( power=2, mu=1, phi=1, y=yy, add=TRUE, lwd=2, type="cdf", col="red")
legend("bottomright",lwd=c(2,2), col=c("black","red"),
      legend=c("p=1.05","p=2") )

### Now, plot two densities, combining  $p > 2$  and  $1 < p < 2$ 
tweedie.plot( power=3.5, mu=1, phi=1, y=yy, lwd=2)
tweedie.plot( power=1.5, mu=1, phi=1, y=yy, lwd=2, col="red", add=TRUE)
legend("topright",lwd=c(2,2), col=c("black","red"), pch=c(NA,19),
      legend=c("p=3.5","p=1.5") )
```

---

tweedie.profile      *Tweedie Distributions: mle estimation of p*

---

## Description

Maximum likelihood estimation of the Tweedie index parameter  $p$ .

## Usage

```
tweedie.profile(formula, p.vec=NULL, link.power=0, data, weights, offset, fit.glm=F,
do.smooth=TRUE, do.plot=FALSE, do.ci=do.smooth,
eps=1/6, do.points=do.plot, method="inversion", conf.level=0.95,
phi.method=ifelse(method == "saddlepoint", "saddlepoint", "mle"), verbose=FALSE)
```

**Arguments**

formula	a formula expression as for other regression models and generalized linear models, of the form <code>response ~ predictors</code> . For details, see the documentation for <code>lm</code> , <code>glm</code> and <code>formula</code>
p.vec	a vector of <code>p</code> values for consideration. The values must all be larger than one (if the response variable has exact zeros, the values must all be between one and two). If <code>NULL</code> (the default), <code>p.vec</code> is set to <code>seq(1.2, 1.8, by=0.05)</code> if the response contains any zeros, or <code>seq(1.5, 5, by=0.25)</code> if the response contains no zeros. See the DETAILS section below for further details.
link.power	the power link function to use. These link functions $g(\cdot)$ are of the form $g(\eta) = \eta^{\text{link.power}}$ , and the special case of <code>link.power=0</code> (the default) refers to the logarithm link function. See the documentation for <code>tweedie</code> also.
data	an optional data frame, list or environment (or object coercible by <code>as.data.frame</code> to a data frame) containing the variables in the model. If not found in <code>data</code> , the variables are taken from <code>environment(formula)</code> , typically the environment from which <code>glm</code> is called.
weights	an optional vector of weights to be used in the fitting process. Should be <code>NULL</code> or a numeric vector.
offset	this can be used to specify an <i>a priori</i> known component to be included in the linear predictor during fitting. This should be <code>NULL</code> or a numeric vector of length either one or equal to the number of cases. One or more <code>offset</code> terms can be included in the formula instead or as well, and if both are specified their sum is used. See <code>model.offset</code> .
fit.glm	logical flag. If <code>TRUE</code> , the Tweedie generalized linear model is fitted using the value of <code>p</code> found by the profiling function. If <code>FALSE</code> (the default), no model is fitted.
do.smooth	logical flag. If <code>TRUE</code> (the default), a spline is fitted to the data to smooth the profile likelihood plot. If <code>FALSE</code> , no smoothing is used (and the function is quicker). <b>Note</b> that <code>p.vec</code> must contain <i>at least five points</i> for smoothing to be allowed.
do.plot	logical flag. If <code>TRUE</code> , a plot of the profile likelihood is produce. If <code>FALSE</code> (the default), no plot is produced.
do.ci	logical flag. If <code>TRUE</code> , the nominal $100 \times \text{conf.level}$ is computed. If <code>FALSE</code> , no confidence interval is computed. By default, <code>do.ci</code> is the same value as <code>do.smooth</code> , since a confidence interval will only be accurate if smoothing has been performed. Indeed, if <code>do.smooth=FALSE</code> , confidence intervals are never computed and <code>do.ci</code> is forced to <code>FALSE</code> if it is given as <code>TRUE</code> .
eps	the offset in computing the variance function. The default is <code>eps=1/6</code> (as suggested by Nelder and Pregibon, 1987). Note <code>eps</code> is ignored unless the <code>method="saddlepoint"</code> as it makes no sense otherwise.
do.points	plot the points on the plot where the (log-) likelihood is computed for the given values of <code>p</code> ; defaults to the same value as <code>do.plot</code>
method	the method for computing the (log-) likelihood. One of <code>"series"</code> , <code>"inversion"</code> (the default), <code>"interpolation"</code> or <code>"saddlepoint"</code> . If there are any

	troubles using this function, often a change of method will fix the problem. Note that <code>method="saddlepoint"</code> is only an approximate method for computing the (log-) likelihood. Using <code>method="interpolation"</code> may produce a jump in the profile likelihood as it changes computational regimes.
<code>conf.level</code>	the confidence level for the computation of the nominal confidence interval. The default is <code>conf.level=0.95</code> .
<code>phi.method</code>	the method for estimating <code>phi</code> , one of <code>"saddlepoint"</code> or <code>"mle"</code> . A maximum likelihood estimate is used unless <code>method="saddlepoint"</code> , when the saddlepoint approximation method is used. Note that using <code>phi.method="saddlepoint"</code> is equivalent to using the mean deviance estimator of <code>phi</code> .
<code>verbose</code>	the amount of feedback requested: 0 or <code>FALSE</code> means minimal feedback (the default), 1 or <code>TRUE</code> means some feedback, or 2 means to show all feedback. Since the function can be slow and sometimes problematic, feedback can be good; but it can also be unnecessary when one knows all is well.

### Details

For each value in `p.vec`, the function computes an estimate of `phi` 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 the value of `p` is found, the distribution within the class of Tweedie distribution is identified.

### Value

The main purpose of the function is to estimate the value of the Tweedie index parameter,  $p$ , which is produced by the output list as `p.max`. Optionally (if `do.plot=TRUE`), a plot is produced that shows the profile log-likelihood computed at each value in `p.vec` (smoothed if `do.smooth=TRUE`). This function can be tempermental (for theoretical reasons involved in numerically computing the density), and this plot shows the values of  $p$  requested on the horizontal axis (using `rug`); there may be fewer points on the plot, since the likelihood some values of  $p$  requested may have returned `NaN`, `Inf` or `NA`.

A list containing the components: `y` and `x` (such that `plot(x, y)` (partially) recreates the profile likelihood plot); `ht` (the height of the nominal confidence interval); `L` (the estimate of the (log-) likelihood at each given value of `p`); `p` (the `p`-values used); `phi` (the computed values of `phi` at the values in `p`); `p.max` (the estimate of the mle of `p`); `L.max` (the estimate of the (log-) likelihood at `p.max`); `phi.max` (the estimate of `phi` at `p.max`); `ci` (the lower and upper limits of the confidence interval for `p`); `method` (the method used for estimation: `series`, `inversion`, `interpolation` or `saddlepoint`); `phi.method` (the method used for estimation of `phi`: `saddlepoint` or `phi`).

If `glm.fit` is `TRUE`, the list also contains a component `glm.obj`, a `glm` object for the fitted Tweedie generalized linear model.

### Note

The estimates of `p` and `phi` are printed. The result is printed invisibly.

If the response variable has any exact zeros, the values in `p.vec` must all be between one and two.

The function is sometimes unstable and may fail. It may also be very slow. One solution is to change the method. The default is `method="inversion"` (the default); then try `method="series"`, `method="interpolation"` and `method="saddlepoint"` in that order. Note that `method="saddlepoint"` is an approximate method only. Also make sure the values in `p.vec` are suitable for the data (see above paragraph).

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.

### Author(s)

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

### References

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

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.

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

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](#) [dtweedie.saddle](#) [tweedie](#)

### Examples

```
library(statmod) # Needed to use tweedie.profile
# Generate some fictitious data
test.data <- rgamma(n=200, scale=1, shape=1)
# The gamma is a Tweedie distribution with power=2;
# let's see if p=2 is suggested by tweedie.profile:
## Not run:
out <- tweedie.profile( test.data ~ 1,
p.vec=seq(1.5, 2.5, by=0.2) )
out$p.max
out$ci

## End(Not run)
```

# Index

## \*Topic **models**

- AICTweedie, 3
  - dtweedie.dldphi, 4
  - dtweedie.saddle, 6
  - Tweedie, 8
  - Tweedie internals, 11
  - tweedie-package, 1
  - tweedie.dev, 12
  - tweedie.plot, 14
  - tweedie.profile, 16
- AIC, 3
- AICTweedie, 3
- deviance, 12, 13
- dtweedie, 5, 7, 13, 15, 19
- dtweedie (*Tweedie*), 8
- dtweedie.dldphi, 4
- dtweedie.dlogfdphi (*Tweedie internals*), 11
- dtweedie.interp (*Tweedie internals*), 11
- dtweedie.jw.smallp (*Tweedie internals*), 11
- dtweedie.kv.bigp (*Tweedie internals*), 11
- dtweedie.logl (*Tweedie internals*), 11
- dtweedie.logv.bigp (*Tweedie internals*), 11
- dtweedie.logw.smallp (*Tweedie internals*), 11
- dtweedie.saddle, 5, 6, 10, 13, 19
- dtweedie.series.bigp (*Tweedie internals*), 11
- dtweedie.series.smallp (*Tweedie internals*), 11
- formula, 16
- glm, 13, 16
- lm, 16
- model.offset, 17
- ptweedie (*Tweedie*), 8
- qtweedie (*Tweedie*), 8
- rtweedie (*Tweedie*), 8
- rug, 18
- stored.grids (*Tweedie internals*), 11
- Tweedie, 8
- Tweedie (*tweedie-package*), 1
- tweedie, 5, 13, 16, 19
- Tweedie internals, 11
- tweedie-package, 1
- tweedie.dev, 12
- tweedie.plot, 14
- tweedie.profile, 5, 16