Package ‘ZIDW’

September 30, 2025

Version 0.1.0

Date 2025-09-22

Title Zero-Inflated Discrete Weibull Models

Depends R (>= 4.1.0)

Imports DWreg, actuar, maxLik, COUNT, gtools, matrixcalc, DiscreteWeibull, dplyr, ggplot2, purrr, tibble

URL https://github.com/dsy109/ZIDW

Description Parameter estimation for zero-inflated discrete Weibull (ZIDW) regression models, the univariate setting, distribution functions, functions to generate randomized quantile residuals a pseudo R2, and plotting of rootograms. For more details, see Kalktawi (2017) <https://bura.brunel.ac.uk/handle/2438/14476>, Taconeli and Rodrigues de Lara (2022) <doi:10.1080/00949655.2021.2005597>, and Yeh and Young (2025) <doi:10.1080/03610918.2025.2

License GPL (>= 2)

NeedsCompilation no

Author Peng Yeh [aut],
Derek S. Young [aut, cre] (ORCID: <https://orcid.org/0000-0002-3048-3803>)

Maintainer Derek S. Young <derek.young@uky.edu>

Repository CRAN

Date/Publication 2025-09-30 07:20:22 UTC

Contents

`ZIDW-package` *Zero-Inflated Discrete Weibull Models*

Description

Parameter estimation for zero-inflated discrete Weibull (ZIDW) regression models, the univariate setting, distribution functions, functions to generate randomized quantile residuals a pseudo R2, and plotting of rootograms.

Details

Package:	ZIDW
Type:	Package
Version:	0.1.0
Date:	2025-09-22
Imports:	DWreg, actuar, maxLik, COUNT, gtools, matrixcalc, DiscreteWeibull, dplyr, ggplot2, purrr, tibble
License:	GPL (>= 2)

Author(s)

Peng Yeh, Ph.D.

Derek S. Young, Ph.D.

Maintainer: Derek S. Young <derek.young@uky.edu>

References

Kalktawi, H. S. (2017), *Discrete Weibull Regression Model for Count Data*. Ph.D. Thesis, Brunel University London.

Taconeli, C. A. and Rodrigues de Lara, I. A. (2022), Discrete Weibull Distribution: Different Estimation Methods Under Ranked Set Sampling and Simple Random Sampling. *Journal of Statistical Computation and Simulation*, **92**:8, 1740–1762.

Yeh, P. and Young, D. S. (2025), Some Estimation and Inference Considerations for the Zero-Inflated Discrete Weibull Distribution. *Communications in Statistics - Simulation and Computation (in press)*, 1–22.

`AIC.zidw` *AIC and BIC for ZIDW Model Fits*

Description

Generic function calculating AIC or BIC for ZIDW model fits.

Usage

```
## S3 method for class 'zidw'
AIC(object, ..., k = 2)
## S3 method for class 'zidw'
BIC(object, ...)
```

Arguments

`object`	A fitted ZIDW model object for which there exists a `logLik` method to extract the corresponding log-likelihood, or an object inheriting from class `logLik`.
`...`	Optionally more fitted model objects.
`k`	Numeric value for the penalty per parameter to be used; default = 2.

Value

A numeric value with the corresponding AIC or BIC.

References

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

See Also

`logLik` `AIC`

Examples

```
## data
data("rwm1984", package = 'COUNT')

zidw_out <- zidw_reg(docvis ~ outwork + female + age,
                     ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)

AIC(zidw_out)
BIC(zidw_out)
```

`bootstrap_lrt` *Bootstrap Likelihood Ratio Test for Zero-Inflation*

Description

Perform the bootstrap likelihood ratio test for comparing DW and ZIDW model fits.

Usage

```
bootstrap_lrt(data, B, tol = -1)
```

Arguments

`data`	Data to test.
`B`	Number of bootstrap samples to draw.
`tol`	Threshold of bootstrap likelihood ratio test statistics. See details.

Details

Theoretically, the likelihood ratio is always positive. In practice, however, if a negative (but close to zero) likelihood ratio occurrs, then the calculations are typically numerically unstable or the fitting procedure failed to converge to the MLEs. We, therefore, set the threshold to -1 by default. If the bootstrap likelihood ratio is less than `tol`, we drop this sample and resample. If it is between `tol` and 0, we truncate it at 0.

Value

Return a list containing four vectors:

`pvalue`	P-value of the test.
`Observe likelihood ratio test statistics`	Likelihood ratio test statistics from the data.
`Bootstrap likelihood ratio test statistics`	Bootstrap likelihood ratio test statistics.
`count`	Number of times that bootstrap LRT is less than `tol`.

See Also

`glm`, `lm`

Examples

```
## data
data("rwm1984", package = 'COUNT')

set.seed(1)
```

```
test <- suppressWarnings(bootstrap_lrt(rwm1984, B = 20))
test
```

coef.zidw *Print Coefficients from a ZIDW Model Fit*

Description

Print the coefficients of a ZIDW object.

Usage

```
## S3 method for class 'zidw'
coef(object, ...)
```

Arguments

`object`	A ZIDW object to extract the model coefficients.
`...`	Further arguments.

Value

Coefficients extracted from the ZIDW object `object`. This will be a named numeric vector.

Examples

```
## data
data("rwm1984", package = 'COUNT')

zidw_out <- zidw_reg(docvis ~ outwork + female + age,
                     ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)

coef.zidw(zidw_out)
```

covid *COVID-19 cases at the 2020 Summer Olympics and 2020 Summer Paralympics*

Description

This dataset contains the number of cases from 2020 Summer Olympics and 2020 Summer Paralympics in Tokyo.

Usage

```
data(covid)
```

Format

This data frame consists of 5 variables on 70 dates:

- `date` Date of recording the number of cases of COVID-19.
- `athlete` Number of cases for athletes.
- `staff` Number of cases for staff.
- `volunteer` Number of cases for volunteers.
- `total` Total number of cases.

Source

Wikipedia contributors. (2022, December 5). COVID-19 cases at the 2020 Summer Olympics and 2020 Summer Paralympics. In Wikipedia, The Free Encyclopedia. Retrieved 13:01, December 20, 2022, https://en.wikipedia.org/wiki/COVID-19_cases_at_the_2020_Summer_Olympics_and_2020_Summer_Paralympics#cite_note-Tokyo2020CovidList-1

`hdw` *Hurdle Discrete Weibull Distribution*

Description

Density, distribution function, quantile function and random generation for the hurdle Discrete Weibull distribution with parameter q_{par}, β and hurdle crossing probability λ (i.e., $1 - \lambda$ is the probability for observed zeros).

Usage

```
dhdw(x, q_par, beta, lam, log = FALSE)
phdw(q, q_par, beta, lam, lower.tail = TRUE, log.p = FALSE)
qhdw(p, q_par, beta, lam, lower.tail = TRUE, log.p = FALSE)
rhdw(n, q_par, beta, lam)
```

Arguments

`x, q`	Vector of quantiles.
`p`	Vector of probabilities.
`n`	Number of observation.
`q_par`	Shape parameter.
`beta`	Shape parameter.
`lam`	Zero-infation parameter.
`log, log.p`	Logical; if `TRUE`, probabilities are returned on log-scale.
`lower.tail`	Logical; if `TRUE` (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.

Details

The hurdle discrete Weibull has the mass function

$f(y) = \lambda + (1-\lambda)\frac{q^{y^\beta} - q^{(y+1)^\beta}}{q}$,

for $y = 0, 1, 2, ...$ where $\lambda \in (0, 1)$ is the zero-inflation parameter, $q \in (0, 1)$, and $\beta > 0$.

Value

`dhdw`	Gives the density.
`phdw`	Gives the cumulative probability.
`qhdw`	Gives the quantile value.
`rhdw`	Generates random numbers.

References

Kalktawi, H. S. (2017), *Discrete Weibull Regression Model for Count Data*. Ph.D. Thesis, Brunel University London.

Yeh, P. and Young, D. S. (2025), Some Estimation and Inference Considerations for the Zero-Inflated Discrete Weibull Distribution. *Communications in Statistics - Simulation and Computation (in press)*, 1–22.

Examples

```
dhdw(.5, .6, 1, .4)
phdw(.5, .6, 1, .4)
qhdw(.8, .6, 1, .4)
rhdw(100, .6, 1, .4)
```

`logLik.zidw` *Extract Log-Likelihood for ZIDW Model Fits*

Description

Extract the log-likelihood of a ZIDW model fit.

Usage

```
## S3 method for class 'zidw'
logLik(object, ...)
```

Arguments

`object`	A fitted ZIDW model object for which there exists a `logLik` method to extract the corresponding log-likelihood, or an object inheriting from class `logLik`.
`...`	Some methods for this generic function require additional arguments.

Value

A numeric value with the corresponding log-likelihood.

See Also

logLik AIC

Examples

```
## data
data("rwm1984", package = 'COUNT')

zidw_out <- zidw_reg(docvis ~ outwork + female + age,
                     ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)

logLik(zidw_out)
```

---

predict.zidw   *Prediction for ZIDW Model Fits*

Description

Obtains predictions from the fitted ZIDW model object.

Usage

```
## S3 method for class 'zidw'
predict(object, newdata,
                         type = c("response", "prob", "count", "zero"),
                         at = NULL, ...)
```

Arguments

object	A fitted object of class inheriting from "zidw".
newdata	Optionally, a data frame in which to look for variables with which to predict. If omitted, the fitted predictors are used.
type	The type of prediction required. For details see below.
at	Optionally, if type = "prob", a numeric vector at which the probabilities are evaluated. By default 0:max(y) is used where y is the original observed response.
...	Currently not used.

Details

The default is `type = "response"`, which is on the scale of the response variable. `type = "prob"` provides the predicted density (i.e., probabilities for the observed counts). `type = "count"` offers the predicted mean from the count component without zero-inflation and `type = "zero"` predicts the probability for the zero-component.

Value

If `type = "response"`, a vector of estimated conditional mean values from the ZIDW model is returned. If `type = "prob"`, a matrix of predicted probabilities is returned, where each row is a vector of predicted probabilities over the range of responses seen in the data (i.e., `min(y):max(y)`). If `type = "count"`, a vector of the predicted means from the count component (without zero-inflation) is returned. If `type = "zero"`, the predicted probability for the zero-component is returned.

Examples

```
## data
data("rwm1984", package = 'COUNT')

## model with covariates
zidw_out <- zidw_reg(docvis ~ outwork + female + age,
                     ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)

predict.zidw(zidw_out)
```

`print.zidw` *Print Values from a ZIDW Object*

Description

Print the output of a ZIDW object.

Usage

```
## S3 method for class 'zidw'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

`x`	A ZIDW object used to select a method.
`digits`	Minimal number of significant digits.
`...`	Further arguments passed to or from other methods.

Value

Returns an object of class zidw based on the object x with mostly the same output when printing output from an object of class lm. Whereas an lm object has one model.matrix in its output, use of print.zidw returns three model matrices, one each for the model parameters q and $beta$ as well as for the zero-inflation component.

See Also

print.lm

Examples

```
## data
data("rwm1984", package = 'COUNT')

zidw_out <- zidw_reg(docvis ~ outwork + female + age,
                     ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)

print(zidw_out)
```

---

rootogram_zidw *Rootogram for a ZIDW fit*

Description

A rootogram is a model diagnostic tool that assesses the goodness-of-fit of a statistical model. The rootogram is drawn using ggplot2::ggplot() graphics.

Usage

```
rootogram_zidw(object, type = c("hanging", "standing", "suspended"),
               sqrt = TRUE, ref_line = TRUE, warn_limits = TRUE,
               fitted_colour = "steelblue", bar_colour = NA,
               bar_fill = "grey", ref_line_colour = "black",
               warn_line_colour = "black", ylab = NULL, xlab = NULL, ...)
```

Arguments

object	A ZIDW model object.
type	Type of rootogram; default is type = "hanging".
sqrt	Logical; show the observed and fitted frequencies (default is sqrt = TRUE).
ref_line	Logical; draw a reference line at zero (default is ref_line = TRUE).
warn_limits	Logical; draw Tukey's warning limit lines at +/- 1 (default is warn_limits = TRUE).
fitted_colour, bar_colour, bar_fill, ref_line_colour, warn_line_colour	Colors used to draw the respective element of the rootogram.

xlab, ylab — Character; labels for the x and y axis of the rootogram. May be missing (NULL), in which case suitable labels will be used.

... — Arguments passed to other methods.

Value

A ggplot object.

References

Kleiber, C. and Zeileis, A. (2016). Visualizing Count Data Regressions Using Rootograms. *The American Statistician*, **70**:3, 296–303.

See Also

rootogram

Examples

```
## data
data("rwm1984", package = 'COUNT')

zidw_out <- zidw_reg(docvis ~ outwork + female + age,
                     ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)
rootogram_zidw(zidw_out)
```

---

rqres_zidw_reg — *Randomized Quantile Residuals for a ZIDW Regression Fit*

Description

Calculate randomized quantile residuals from a ZIDW regression fit.

Usage

```
rqres_zidw_reg(test, plot = FALSE)
```

Arguments

test — Zero-inflated discrete Weibull regresssion output.

plot — If plot = TRUE, produce the randomized quantile residuals plot.

Value

Return a vector of randomized quantile resuduals or a randomized quantile residuals plot:

rqr — Randomized quantile resuduals.

References

Dunn, P. K. and Smyth, G. K. (1996), Randomzied Quantile Residuals. *Journal of Computational and Graphical Statistics*, **5**:3, 236–244.

Yeh, P. and Young, D. S. (2025), Some Estimation and Inference Considerations for the Zero-Inflated Discrete Weibull Distribution. *Communications in Statistics - Simulation and Computation (in press)*, 1–22.

See Also

`glm`, `lm`

Examples

```
## data
data("rwm1984", package = 'COUNT')

zidw_out <- zidw_reg(docvis ~ outwork + female + age,
                     ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)
rqr <- rqres_zidw_reg(zidw_out)
```

`summary.zidw` *Object Summaries*

Description

`summary` method to produce results for objects of class `"zidw"`.

Usage

```
## S3 method for class 'zidw'
summary(object, ...)
## S3 method for class 'summary.zidw'
print(x, digits = max(3, getOption("digits") - 3), ...)
```

Arguments

`object`	An object of class `"zidw"` for which a summary is desired.
`...`	Additional arguments affecting the summary produced.
`x`	A `summary.zidw` object.
`digits`	The number of digits in the output.

Details

Additional information about the ZIDW fit represented by `object` is extracted and included as components of `object`. The returned object has a `print` method.

Value

Currently, limited results from an object of class "zidw" is returned. The output is of class `summary.zidw`. Future versions of this code will attempt to mimic `summary.glm`.

See Also

`summary.glm`

Examples

```
## data
data("rwm1984", package = 'COUNT')

## model with covariates
zidw_out <- zidw_reg(docvis ~ outwork + female + age,
                     ziformula = ~ age + female, betaformula = ~ 1,
                     data = rwm1984[1:100, ], beta = .1)

summary(zidw_out)
```

---

zidw *Zero-Inflated Discrete Weibull Distribution*

Description

Density, distribution function, quantile function and random generation for the zero-inflated Discrete Weibull distribution with parameters q_{par}, β, λ.

Usage

```
dzidw(x, q_par, beta, lam, log = FALSE)
pzidw(q, q_par, beta, lam, lower.tail = TRUE, log.p = FALSE)
qzidw(p, q_par, beta, lam, lower.tail = TRUE, log.p = FALSE)
rzidw(n, q_par, beta, lam)
```

Arguments

`x, q`	Vector of quantiles.
`p`	Vector of probabilities.
`n`	Number of observation.
`q_par`	Shape parameter.
`beta`	Shape parameter.
`lam`	Zero-infation parameter.
`log, log.p`	Logical; if `TRUE`, probabilities are returned on log-scale.
`lower.tail`	Logical; if `TRUE` (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.

Details

The zero-inflated discrete Weibull has the mass function

$f(y) = \lambda + (1 - \lambda)(q^{y^\beta} - q^{(y+1)^\beta}),$

for $y = 0, 1, 2, ...$ where $\lambda \in (0, 1)$ is zero-inflation parameter, $q \in (0, 1)$, and $\beta > 0$.

Value

`dzidw`	Gives the density.
`pzidw`	Gives the cumulative probability.
`qzidw`	Gives the quantile value.
`rzidw`	Generates random numbers.

References

Kalktawi, H. S. (2017), *Discrete Weibull Regression Model for Count Data*. Ph.D. Thesis, Brunel University London.

Yeh, P. and Young, D. S. (2025), Some Estimation and Inference Considerations for the Zero-Inflated Discrete Weibull Distribution. *Communications in Statistics - Simulation and Computation (in press)*, 1–22.

Examples

```
dzidw(.5, .6, 1, .4)
pzidw(.5, .6, 1, .4)
qzidw(.8, .6, 1, .4)
rzidw(100, .6, 1, .4)
```

`zidw_reg` *Fitting of Zero-Inflated Discrete Weibull Regression Models*

Description

Fitting of the zero-inflated discrete Weibull regression model is done via `constrOptim`. Fitting of the univariate model can be accomplished via maximum likelihood or minimum distance estimation.

Usage

```
zidw_reg(qformula, betaformula = ~ 1, ziformula = ~ 1, data, lam = NULL,
         beta = NULL, q = NULL, k = 1000, uni_method =  c('MLE', 'mde'),
         max_method = NULL, constraint = TRUE, B = NULL)
```

Arguments

qformula	A symbolic description of the model to be fitted for the parameter q, including the response variable.
betaformula	A symbolic description of the model to be fitted for the parameter β.
ziformula	a symbolic description of the model to be fitted for the parameter λ.
data	A data frame containg model variables.
lam	The starting value for λ.
beta	The starting value for parameter β.
q	The starting value for parameter q.
k	A cutofff value used to calculate residuals.
uni_method	The univariate fitting method. Methods include uni_method = 'MLE' (default) for maximum likelihood estimation and uni_method = 'MDE' for minimum distance estimation.
max_method	Maximum likelihood estimation method for the univariate setting. See maxLik for the different methods.
constraint	Set constraint = FALSE for unconstrained maximization. constraint = TRUE is the default.
B	Number of bootstrap replciates for standard error estimation when modeling univariate data using uni_method = 'MDE'.

Value

zidw_reg An object of class "zidw", i.e., a list with components including the following:

call	The original function call.
coefficients	A list with elements "zero", "beta", and "q" containing the coefficients from the respective models.
loglik	The log-likehood of the fitted model.
SE	Estimated standard errors.
convergence	The convergence code of optim.
nall	The number of observations.
res	A vector of raw residuals (observed - fitted).
fitted_values	The fitted values.
model_matrix_q	The model matrix of parameter q.
model_matrix_beta	The model matrix of parameter β.
model_matrix_zi	The model matrix of parameter λ.
response	Vector of response values.
model	The dataset.
formula	A list containing three formulas for q, β, and λ.

References

Taconeli, C. A. and Rodrigues de Lara, I. A. (2022), Discrete Weibull Distribution: Different Estimation Methods Under Ranked Set Sampling and Simple Random Sampling. *Journal of Statistical Computation and Simulation*, **92**:8, 1740–1762.

Yeh, P. and Young, D. S. (2025), Some Estimation and Inference Considerations for the Zero-Inflated Discrete Weibull Distribution. *Communications in Statistics - Simulation and Computation (in press)*, 1–22.

See Also

glm, lm

Examples

```
## data
data("rwm1984", package = 'COUNT')

## model with covariates
zidw_out <- zidw_reg(docvis ~ outwork + female + age, ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)

## model without covariates
zidw_out2 <- zidw_reg(docvis ~ 1, data = rwm1984)
```

---

zidw_r_squared *Pseudo-R^2 for ZIDW Regression*

Description

Calculates the pseudo-R^2 for ZIDW regression fits. Modeling q and λ.

Usage

```
zidw_r_squared(object, adj = TRUE)
```

Arguments

`object`	An object from a ZIDW regression fit of class `"zidw"`.
`adj`	An adjustment for calculating the pseudo-R^2. Default is `adj = TRUE`.

Value

A numerical value with the corresponding pseudo-R^2.

References

Martin, J. and Hall, D. B. (2016), R^2 Measures for Zero-Inflated Regression Models for Count Data with Excess Zeros. *Journal of Statistical Computation and Simulation*, **84**:18, 3777–3790.

Examples

```
## data
data("rwm1984", package = 'COUNT')

## model with covariates
zidw_out <- zidw_reg(docvis ~ outwork + female + age, ziformula = ~ age + female,
                     betaformula = ~ 1, data = rwm1984[1:100, ], beta = .1)

zidw_r_squared(zidw_out)
```

---

zidw_uni *Parameter Estimation for the Zero-Inflated Discrete Weibull*

Description

Parameter estimation for the univariate zero-inflated discrete Weibull distribution using maximum likelihood or minimum distance estimation.

Usage

```
zidw_uni(y, par = NULL, method = c('MLE', 'mde'), B = NULL, max_method = NULL,
         constraint = TRUE)
```

Arguments

y	Vector of univariate counts.
par	Vector of starting values in the order of λ, β, and q.
method	Parameter estimation method. Methods include `uni_method = 'MLE'` (default) for maximum likelihood estimation and `uni_method = 'MDE'` for minimum distance estimation.
B	Number of bootstrap replciates for standard error estimation when modeling univariate data using `uni_method = 'MDE'`.
max_method	Maximum likelihood estimation method for the univariate setting. See maxLik for the different methods.
constraint	Logical value to constrain the parameter to be within the parameter space. Default is `constraint = TRUE`.

Value

Output for for maximum likelihood estimation (`uni_method = 'MLE'`):

MLE	Maximum likelihood estimates.
coefficients	Coefficients for q, β, and λ.
convergence	Convergence code of `maxLik`.
iteration	Number of iterations.

SE Estimated standard errors.

Output for for minimum distance estimation (uni_method = 'MDE'):

lam Parameter estimate for λ.

Beta Parameter estimate for β.

q Parameter estimate for q.

distance The final distance value.

SE Bootstrap standard error estimates for λ, β, and q.

References

Taconeli, C. A. and Rodrigues de Lara, I. A. (2022), Discrete Weibull Distribution: Different Estimation Methods Under Ranked Set Sampling and Simple Random Sampling. *Journal of Statistical Computation and Simulation*, **92**:8, 1740–1762.

Yeh, P. and Young, D. S. (2025), Some Estimation and Inference Considerations for the Zero-Inflated Discrete Weibull Distribution. *Communications in Statistics - Simulation and Computation (in press)*, 1–22.

See Also

glm, lm

Examples

```
## data
data("rwm1984", package = 'COUNT')

## MLE method
zidw_out <- zidw_uni(rwm1984$docvis)


## mde without standard error estimation
set.seed(1)
zidw_out2 <- zidw_uni(rwm1984$docvis, method = 'mde')

## mde with bootstrap standard error estimation
set.seed(1)
zidw_out2 <- zidw_uni(rwm1984$docvis, method = 'mde', B = 5)
```

ztdw *Zero-Truncated Discrete Weibull Distribution*

Description

Density, distribution function, quantile function and random generation for the zero-truncated Discrete Weibull distribution with parameter q_{par}, β.

Usage

```
dztdw(x, q_par, beta, log = FALSE)
pztdw(q, q_par, beta, lower.tail = TRUE, log.p = FALSE)
qztdw(p, q_par, beta, lower.tail = TRUE, log.p = FALSE)
rztdw(n, q_par, beta)
```

Arguments

x, q	Vector of quantiles.
p	Vector of probabilities.
n	Number of observation.
q_par	Shape parameter.
beta	Shape parameter.
log, log.p	Logical; if TRUE, probabilities are returned on log-scale.
lower.tail	Logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.

Details

The zero-truncated discrete Weibull has the mass function
$f(y) = \frac{q^{y^\beta} - q^{(y+1)^\beta}}{q}$,
for $y = 1, 2, 3, ...$ where $q \in (0, 1)$, and $\beta > 0$.

Value

dztdw	Gives the density.
pztdw	Gives the cumulative probability.
qztdw	Gives the quantile value.
rztdw	Generates random numbers.

References

Kalktawi, H. S. (2017), *Discrete Weibull Regression Model for Count Data.* Ph.D. Thesis, Brunel University London.

Yeh, P. and Young, D. S. (2025), Some Estimation and Inference Considerations for the Zero-Inflated Discrete Weibull Distribution. *Communications in Statistics - Simulation and Computation (in press)*, 1–22.

Examples

```
dztdw(1, .6, 1)
pztdw(1, .6, 1)
qztdw(.8, .6, 1)
rztdw(100, .6, 1)
```

Index