

# Package: UPG (via r-universe)

August 30, 2024

**Type** Package

**Title** Efficient Bayesian Algorithms for Binary and Categorical Data Regression Models

**Version** 0.3.4

**Author** Gregor Zens [aut, cre], Sylvia Frühwirth-Schnatter [aut], Helga Wagner [aut]

**Maintainer** Gregor Zens <zens@iiasa.ac.at>

**Description** Efficient Bayesian implementations of probit, logit, multinomial logit and binomial logit models. Functions for plotting and tabulating the estimation output are available as well. Estimation is based on Gibbs sampling where the Markov chain Monte Carlo algorithms are based on the latent variable representations and marginal data augmentation algorithms described in ``Gregor Zens, Sylvia Frühwirth-Schnatter & Helga Wagner (2023). Ultimate Pólya Gamma Samplers – Efficient MCMC for possibly imbalanced binary and categorical data, Journal of the American Statistical Association <doi:10.1080/01621459.2023.2259030>".

**Encoding** UTF-8

**License** GPL-3

**Language** en-US

**Depends** R (>= 3.5.0)

**Imports** ggplot2, knitr, matrixStats, mnormt, pgdraw, reshape2, coda, truncnorm

**LazyData** true

**RoxygenNote** 7.2.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2023-11-04 14:30:05 UTC

## Contents

coef.UPG.Binomial . . . . .	2
coef.UPG.Logit . . . . .	3
coef.UPG.MNL . . . . .	4
coef.UPG.Probit . . . . .	5
lfp . . . . .	6
logLik.UPG.Binomial . . . . .	7
logLik.UPG.Logit . . . . .	8
logLik.UPG.MNL . . . . .	9
logLik.UPG.Probit . . . . .	10
plot.UPG.Binomial . . . . .	11
plot.UPG.Logit . . . . .	12
plot.UPG.MNL . . . . .	14
plot.UPG.Probit . . . . .	15
predict.UPG.Binomial . . . . .	17
predict.UPG.Logit . . . . .	18
predict.UPG.MNL . . . . .	19
predict.UPG.Probit . . . . .	20
print.UPG.Binomial . . . . .	21
print.UPG.Logit . . . . .	22
print.UPG.MNL . . . . .	23
print.UPG.Probit . . . . .	24
program . . . . .	24
summary.UPG.Binomial . . . . .	25
summary.UPG.Logit . . . . .	27
summary.UPG.MNL . . . . .	28
summary.UPG.Probit . . . . .	30
titanic . . . . .	31
UPG . . . . .	32
UPG.Diag . . . . .	34
UPG.Diag.Binomial . . . . .	35
UPG.Diag.Logit . . . . .	36
UPG.Diag.MNL . . . . .	36
UPG.Diag.Probit . . . . .	37
<b>Index</b>	<b>38</b>

---

coef.UPG.Binomial	<i>Extract coefficients from UPG.Binomial objects</i>
-------------------	---

---

### Description

coef can be used to extract posterior means and credible intervals based on posterior quantiles from UPG.Binomial objects.

**Usage**

```
## S3 method for class 'UPG.Binomial'  
coef(object, ..., q = c(0.025, 0.975))
```

**Arguments**

object            an object of class UPG.Binomial.  
...                other coef parameters.  
q                  a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.

**Value**

Returns a matrix containing posterior means and the desired credible interval.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Binomial](#) to summarize a UPG.Binomial object and create tables. [predict.UPG.Binomial](#) to predict probabilities using a UPG.Binomial object. [plot.UPG.Binomial](#) to plot a UPG.Binomial object.

**Examples**

```
# estimate a binomial logit model using example data  
library(UPG)  
data(titanic)  
y = titanic[,1]  
Ni = titanic[,2]  
X = titanic[,-c(1,2)]  
results.binomial = UPG(y = y, X = X, Ni = Ni, model = "binomial")  
  
# extract posterior means and credible interval based on 0.025 and 0.975 quantiles  
coef(results.binomial, q = c(0.025, 0.975))
```

---

coef.UPG.Logit

*Extract coefficients from UPG.Logit objects*

---

**Description**

coef can be used to extract posterior means and credible intervals based on posterior quantiles from UPG.Logit objects.

**Usage**

```
## S3 method for class 'UPG.Logit'
coef(object, ..., q = c(0.025, 0.975))
```

**Arguments**

object	an object of class UPG.Logit.
...	other coef parameters.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.

**Value**

Returns a matrix containing posterior means and the desired credible interval.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Logit](#) to summarize a UPG.Logit object and create tables. [predict.UPG.Logit](#) to predict probabilities using a UPG.Logit object. [plot.UPG.Logit](#) to plot a UPG.Logit object.

**Examples**

```
# estimate a logit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.logit = UPG(y = y, X = X, model = "logit")

# extract posterior means and credible interval based on 0.025 and 0.975 quantiles
coef(results.logit, q = c(0.025, 0.975))
```

---

coef.UPG.MNL

*Extract coefficients from UPG.MNL objects*


---

**Description**

coef can be used to extract posterior means and credible intervals based on posterior quantiles from UPG.MNL objects.

**Usage**

```
## S3 method for class 'UPG.MNL'
coef(object, ..., q = c(0.025, 0.975))
```

**Arguments**

object            an object of class UPG.MNL.  
 ...                other coef parameters.  
 q                  a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.

**Value**

Returns a list containing posterior means and the desired credible interval.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.MNL](#) to summarize a UPG.MNL object and create tables. [predict.UPG.MNL](#) to predict probabilities using a UPG.MNL object. [plot.UPG.MNL](#) to plot a UPG.MNL object.

**Examples**

```
# estimate a multinomial logit model using example data
library(UPG)
data(program)
y = program[,1]
X = program[,-1]
results.mnl = UPG(y = y, X = X, model = "mnl")

# extract posterior means and credible interval based on 0.025 and 0.975 quantiles
coef(results.mnl, q = c(0.025, 0.975))
```

---

coef.UPG.Probit            *Extract coefficients from UPG.Probit objects*

---

**Description**

coef can be used to extract posterior means and credible intervals based on posterior quantiles from UPG.Probit objects.

**Usage**

```
## S3 method for class 'UPG.Probit'
coef(object, ..., q = c(0.025, 0.975))
```

**Arguments**

object	an object of class <code>UPG.Probit</code> .
...	other coef parameters.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.

**Value**

Returns a matrix containing posterior means and the desired credible interval.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Probit](#) to summarize a `UPG.Probit` object and create tables. [predict.UPG.Probit](#) to predict probabilities using a `UPG.Probit` object. [plot.UPG.Probit](#) to plot a `UPG.Probit` object.

**Examples**

```
# estimate a probit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.probit = UPG(y = y, X = X, model = "probit")

# extract posterior means and credible interval based on 0.025 and 0.975 quantiles
coef(results.probit, q = c(0.025, 0.975))
```

---

lfp

*Female labor force participation data.*


---

**Description**

A dataset containing socio-economic characteristics as well as a labor force participation dummy for 753 married women from the panel study of income dynamics.

**Usage**

```
lfp
```

**Format**

A data frame with 753 rows and 9 variables:

**lfp** binary indicator for participating in the labor force (=1) or not (=0)

**intercept** intercept

**k5** number of children 5 years old or younger

**k618** number of children 6 to 18 years old

**age** age in years

**wc** binary indicator for college education of the wife

**hc** binary indicator for college education of the husband

**lwg** log expected wage rate; for women in the labor force, the actual wage rate; for women not in the labor force, an imputed value based on a regression of lwg on the other variables

**inc** family income exclusive of wife's income

**Source**

Data taken from 'carData' package. Also known as the 'Mroz' dataset. Mroz, T. A. (1987) The sensitivity of an empirical model of married women's hours of work to economic and statistical assumptions. *Econometrica* 55, 765-799.

---

logLik.UPG.Binomial    *Compute log-likelihoods from UPG.Binomial objects*

---

**Description**

logLik can be used to compute log-likelihoods from UPG.Binomial objects. The log-likelihood is based on the posterior mean of the coefficients.

**Usage**

```
## S3 method for class 'UPG.Binomial'
logLik(object = NULL, ...)
```

**Arguments**

**object**            an object of class UPG.Binomial.  
**...**                other logLik parameters.

**Value**

Returns a numeric of class logLik with attributes containing the number of estimated parameters and the number of observations.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Binomial](#) to summarize a UPG.Binomial object and create tables. [plot.UPG.Binomial](#) to plot a UPG.Binomial object. [coef.UPG.Binomial](#) to extract coefficients from a UPG.Binomial object.

**Examples**

```
# estimate a binomial logit model using example data
library(UPG)
data(titanic)
y = titanic[,1]
Ni = titanic[,2]
X = titanic[,-c(1,2)]
results.binomial = UPG(y = y, X = X, Ni = Ni, model = "binomial")

# extract log-likelihood
ll.binomial = logLik(results.binomial)
```

---

logLik.UPG.Logit

*Compute log-likelihoods from UPG.Logit objects*


---

**Description**

logLik can be used to compute log-likelihoods from UPG.Logit objects. The log-likelihood is based on the posterior mean of the coefficients.

**Usage**

```
## S3 method for class 'UPG.Logit'
logLik(object = NULL, ...)
```

**Arguments**

object            an object of class UPG.Logit.  
...                other logLik parameters.

**Value**

Returns a numeric of class logLik with attributes containing the number of estimated parameters and the number of observations.

**Author(s)**

Gregor Zens



**See Also**

[summary.UPG.Logit](#) to summarize a UPG.Logit object and create tables. [plot.UPG.Logit](#) to plot a UPG.Logit object. [coef.UPG.Logit](#) to extract coefficients from a UPG.Logit object.

**Examples**

```
# estimate a logit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.logit = UPG(y = y, X = X, model = "logit")

# extract log-likelihood
ll.logit = logLik(results.logit)
```

---

logLik.UPG.MNL	<i>Compute log-likelihoods from UPG.MNL objects</i>
----------------	---

---

**Description**

logLik can be used to compute log-likelihoods from UPG.MNL objects. The log-likelihood is based on the posterior mean of the coefficients.

**Usage**

```
## S3 method for class 'UPG.MNL'
logLik(object = NULL, ...)
```

**Arguments**

object            an object of class UPG.MNL.  
...                other logLik parameters.

**Value**

Returns a numeric of class logLik with attributes containing the number of estimated parameters and the number of observations.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.MNL](#) to summarize a UPG.MNL object and create tables. [plot.UPG.MNL](#) to plot a UPG.MNL object. [coef.UPG.MNL](#) to extract coefficients from a UPG.MNL object.

## Examples

```
# estimate a multinomial logit model using example data
library(UPG)
data(program)
y = program[,1]
X = program[,-1]
results.mnl = UPG(y = y, X = X, model = "mnl")

# extract log-likelihood
ll.mnl = logLik(results.mnl)
```

---

logLik.UPG.Probit	<i>Compute log-likelihoods from UPG.Probit objects</i>
-------------------	--

---

## Description

logLik can be used to compute log-likelihoods from UPG.Probit objects. The log-likelihood is based on the posterior mean of the coefficients.

## Usage

```
## S3 method for class 'UPG.Probit'
logLik(object = NULL, ...)
```

## Arguments

object	an object of class UPG.Probit.
...	other logLik parameters.

## Value

Returns a numeric of class logLik with attributes containing the number of estimated parameters and the number of observations.

## Author(s)

Gregor Zens

## See Also

[summary.UPG.Probit](#) to summarize a UPG.Probit object and create tables. [plot.UPG.Probit](#) to plot a UPG.Probit object. [coef.UPG.Probit](#) to extract coefficients from a UPG.Probit object.

**Examples**

```
# estimate a probit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.probit = UPG(y = y, X = X, model = "probit")

# extract log-likelihood
ll.probit = logLik(results.probit)
```

---

plot.UPG.Binomial      *Coefficient plots for UPG.Binomial objects*

---

**Description**

plot generates plots from UPG.Binomial objects using ggplot2. Coefficient plots show point estimates for all coefficients as well as their credible intervals.

**Usage**

```
## S3 method for class 'UPG.Binomial'
plot(
  x = NULL,
  ...,
  sort = FALSE,
  names = NULL,
  xlab = NULL,
  ylab = NULL,
  q = c(0.025, 0.975),
  include = NULL
)
```

**Arguments**

x	an object of class UPG.Binomial.
...	other plot parameters.
sort	a logical variable indicating whether the plotted coefficients should be sorted according to effect sizes. Default is FALSE.
names	a character vector indicating names for the variables used in the plots.
xlab	a character vector of length 1 indicating a title for the x-axis.
ylab	a character vector of length 1 indicating a title for the y-axis.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.
include	can be used to plot only a subset of variables. Specify the columns of X that should be kept in the plot. See examples for further information.

**Value**

Returns a ggplot2 object.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Binomial](#) to summarize a UPG.Binomial object and create tables. [predict.UPG.Binomial](#) to predict probabilities using a UPG.Binomial object. [coef.UPG.Binomial](#) to extract coefficients from a UPG.Binomial object.

**Examples**

```
# estimate a binomial logit model using example data
library(UPG)
data(titanic)
y = titanic[,1]
Ni = titanic[,2]
X = titanic[,-c(1,2)]
results.binomial = UPG(y = y, X = X, Ni = Ni, model = "binomial")

# plot the results and sort coefficients by effect size
plot(results.binomial, sort = TRUE)

# plot only variables 1 and 3 with custom names, credible intervals and axis labels
plot(results.binomial,
      include = c(1,3),
      names = c("Custom 1", "Custom 2"),
      q = c(0.1, 0.9),
      xlab = c("Custom X"),
      ylab = c("Custom Y"))
```

---

plot.UPG.Logit

*Coefficient plots for UPG.Logit objects*

---

**Description**

plot generates plots from UPG.Logit objects using ggplot2. Coefficient plots show point estimates for all coefficients as well as their credible intervals.

**Usage**

```
## S3 method for class 'UPG.Logit'
plot(
  x = NULL,
  ...,
```

```

    sort = FALSE,
    names = NULL,
    xlab = NULL,
    ylab = NULL,
    q = c(0.025, 0.975),
    include = NULL
  )

```

### Arguments

x	an object of class UPG.Logit.
...	other plot parameters.
sort	a logical variable indicating whether the plotted coefficients should be sorted according to effect sizes. Default is FALSE.
names	a character vector indicating names for the variables used in the plots.
xlab	a character vector of length 1 indicating a title for the x-axis.
ylab	a character vector of length 1 indicating a title for the y-axis.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.
include	can be used to plot only a subset of variables. Specify the columns of X that should be kept in the plot. See examples for further information.

### Value

Returns a ggplot2 object.

### Author(s)

Gregor Zens

### See Also

[summary.UPG.Logit](#) to summarize a UPG.Logit object and create tables. [predict.UPG.Logit](#) to predict probabilities using a UPG.Logit object. [coef.UPG.Logit](#) to extract coefficients from a UPG.Logit object.

### Examples

```

# estimate a logit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.logit = UPG(y = y, X = X, model = "logit")

# plot the results and sort coefficients by effect size
plot(results.logit, sort = TRUE)

```

```
# plot only variables 1 and 3 with custom names, credible intervals and axis labels
plot(results.logit,
      include = c(1,3),
      names   = c("Custom 1", "Custom 2"),
      q       = c(0.1, 0.9),
      xlab    = c("Custom X"),
      ylab    = c("Custom Y"))
```

---

plot.UPG.MNL

*Coefficient plots for UPG.MNL objects*


---

### Description

plot generates plots from UPG.MNL objects using ggplot2. Coefficient plots show point estimates for all coefficients in all groups except the baseline as well as their credible intervals.

### Usage

```
## S3 method for class 'UPG.MNL'
plot(
  x = NULL,
  ...,
  sort = FALSE,
  names = NULL,
  groups = NULL,
  xlab = NULL,
  ylab = NULL,
  q = c(0.025, 0.975),
  include = NULL
)
```

### Arguments

x	an object of class UPG.MNL.
...	other plot parameters.
sort	a logical variable indicating whether the plotted coefficients should be sorted according to average effect sizes across groups. Default is FALSE.
names	a character vector indicating names for the variables used in the plots.
groups	a character vector indicating names for the groups excluding the baseline. The group names must correspond to the ordering in the dependent variable used for estimation.
xlab	a character vector of length 1 indicating a title for the x-axis.
ylab	a character vector of length 1 indicating a title for the y-axis.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.
include	can be used to plot only a subset of variables. Specify the columns of X that should be kept in the plot. See examples for further information.

**Value**

Returns a ggplot2 object.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.MNL](#) to summarize a UPG.MNL object and create tables. [predict.UPG.MNL](#) to predict probabilities using a UPG.MNL object. [coef.UPG.MNL](#) to extract coefficients from a UPG.MNL object.

**Examples**

```
# estimate a multinomial logit model using example data
library(UPG)
data(program)
y = program[,1]
X = program[,-1]
results.mnl = UPG(y = y, X = X, model = "mnl")

# plot the results and sort coefficients by average effect size
plot(results.mnl, sort = TRUE)

# plot only variables 1 and 3 with custom group and variable names
# also, customize credible intervals and axis labels
plot(results.mnl,
      include = c(1,3),
      names = c("Custom 1", "Custom 2"),
      groups = c("Alpha", "Beta"),
      q = c(0.1, 0.9),
      xlab = c("Custom X"),
      ylab = c("Custom Y"))
```

---

plot.UPG.Probit

*Coefficient plots for UPG.Probit objects*


---

**Description**

plot generates plots from UPG.Probit objects using ggplot2. Coefficient plots show point estimates for all coefficients as well as their credible intervals.

**Usage**

```
## S3 method for class 'UPG.Probit'
plot(
  x = NULL,
  ...,
```

```

    sort = FALSE,
    names = NULL,
    xlab = NULL,
    ylab = NULL,
    q = c(0.025, 0.975),
    include = NULL
  )

```

### Arguments

x	an object of class UPG.Probit.
...	other plot parameters.
sort	a logical variable indicating whether the plotted coefficients should be sorted according to effect sizes. Default is FALSE.
names	a character vector indicating names for the variables used in the plots.
xlab	a character vector of length 1 indicating a title for the x-axis.
ylab	a character vector of length 1 indicating a title for the y-axis.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.
include	can be used to plot only a subset of variables. Specify the columns of X that should be kept in the plot. See examples for further information.

### Value

Returns a ggplot2 object.

### Author(s)

Gregor Zens

### See Also

[summary.UPG.Probit](#) to summarize a UPG.Probit object and create tables. [predict.UPG.Probit](#) to predict probabilities using a UPG.Probit object. [coef.UPG.Probit](#) to extract coefficients from a UPG.Probit object.

### Examples

```

# estimate a probit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.probit = UPG(y = y, X = X, model = "probit")

# plot the results and sort coefficients by effect size
plot(results.probit, sort = TRUE)

```



```
# plot only variables 1 and 3 with custom names, credible intervals and axis labels
plot(results.probit,
      include = c(1, 3),
      names   = c("Custom 1", "Custom 2"),
      q       = c(0.1, 0.9),
      xlab    = c("Custom X"),
      ylab    = c("Custom Y"))
```

---

predict.UPG.Binomial *Predicted probabilities from UPG.Binomial objects*

---

## Description

predict generates predicted probabilities from a UPG.Binomial object. In addition, credible intervals for these probabilities are computed. Probabilities can be predicted from the data used for estimating the model or for a new data set with the same structure.

## Usage

```
## S3 method for class 'UPG.Binomial'
predict(object = NULL, ..., newdata = NULL, q = c(0.025, 0.975))
```

## Arguments

object	an object of class UPG.Binomial.
...	other predict parameters.
newdata	a matrix or a data.frame containing new explanatory data. The number of columns and the variable ordering must be the same as in the explanatory data used for estimation to generate valid predictions. If no new data is provided, predict will return predicted probabilities for the data used for estimating the model.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.

## Value

Returns a list containing posterior means of predicted probabilities as well as the desired credible interval.

## Author(s)

Gregor Zens

## See Also

[summary.UPG.Binomial](#) to summarize a UPG.Binomial object and create tables. [plot.UPG.Binomial](#) to plot a UPG.Binomial object. [coef.UPG.Binomial](#) to extract coefficients from a UPG.Binomial object.

**Examples**

```
# estimate a binomial logit model using example data
library(UPG)
data(titanic)
y = titanic[,1]
Ni = titanic[,2]
X = titanic[,-c(1,2)]
results.binomial = UPG(y = y, X = X, Ni = Ni, model = "binomial")

# extract predicted probabilities
predict(results.binomial)
```

---

predict.UPG.Logit      *Predicted probabilities from UPG.Logit objects*

---

**Description**

predict generates predicted probabilities from a UPG.Logit object. In addition, credible intervals for these probabilities are computed. Probabilities can be predicted from the data used for estimating the model or for a new data set with the same structure.

**Usage**

```
## S3 method for class 'UPG.Logit'
predict(object = NULL, ..., newdata = NULL, q = c(0.025, 0.975))
```

**Arguments**

object	an object of class UPG.Logit.
...	other predict parameters.
newdata	a matrix or a data.frame containing new explanatory data. The number of columns and the variable ordering must be the same as in the explanatory data used for estimation to generate valid predictions. If no new data is provided, predict will return predicted probabilities for the data used for estimating the model.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.

**Value**

Returns a list containing posterior means of predicted probabilities as well as the desired credible interval.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Logit](#) to summarize a UPG.Logit object and create tables. [plot.UPG.Logit](#) to plot a UPG.Logit object. [coef.UPG.Logit](#) to extract coefficients from a UPG.Logit object.

**Examples**

```
# estimate a logit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.logit = UPG(y = y, X = X, model = "logit")

# extract predicted probabilities
predict(results.logit)
```

---

predict.UPG.MNL	<i>Predicted probabilities from UPG.MNL objects</i>
-----------------	---

---

**Description**

predict generates predicted probabilities from a UPG.MNL object. In addition, credible intervals for these probabilities are computed. Probabilities can be predicted from the data used for estimating the model or for a new data set with the same structure.

**Usage**

```
## S3 method for class 'UPG.MNL'
predict(object = NULL, ..., newdata = NULL, q = c(0.025, 0.975))
```

**Arguments**

object	an object of class UPG.MNL.
...	other predict parameters.
newdata	a matrix or a data.frame containing new explanatory data. The number of columns and the variable ordering must be the same as in the explanatory data used for estimation to generate valid predictions. If no new data is provided, predict will return predicted probabilities for the data used for estimating the model.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.

**Value**

Returns a list containing posterior means of predicted probabilities as well as the desired credible interval.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.MNL](#) to summarize a UPG.MNL object and create tables. [plot.UPG.MNL](#) to plot a UPG.MNL object. [coef.UPG.MNL](#) to extract coefficients from a UPG.MNL object.

**Examples**

```
# estimate a multinomial logit model using example data
library(UPG)
data(program)
y = program[,1]
X = program[,-1]
results.mnl = UPG(y = y, X = X, model = "mnl")

# extract predicted probabilities
predict(results.mnl)
```

---

predict.UPG.Probit      *Predicted probabilities from UPG.Probit objects*

---

**Description**

predict generates predicted probabilities from a UPG.Probit object. In addition, credible intervals for these probabilities are computed. Probabilities can be predicted from the data used for estimating the model or for a new data set with the same structure.

**Usage**

```
## S3 method for class 'UPG.Probit'
predict(object = NULL, ..., newdata = NULL, q = c(0.025, 0.975))
```

**Arguments**

object	an object of class UPG.Probit.
...	other predict parameters.
newdata	a matrix or a data.frame containing new explanatory data. The number of columns and the variable ordering must be the same as in the explanatory data used for estimation to generate valid predictions. If no new data is provided, predict will return predicted probabilities for the data used for estimating the model.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.

**Value**

Returns a list containing posterior means of predicted probabilities as well as the desired credible interval.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Probit](#) to summarize a UPG.Probit object and create tables. [plot.UPG.Probit](#) to plot a UPG.Probit object. [coef.UPG.Probit](#) to extract coefficients from a UPG.Probit object.

**Examples**

```
# estimate a probit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.probit = UPG(y = y, X = X, model = "probit")

# extract predicted probabilities
predict(results.probit)
```

---

`print.UPG.Binomial`      *Print information for UPG.Binomial objects*

---

**Description**

`print` provides some basic information about a UPG.Binomial object.

**Usage**

```
## S3 method for class 'UPG.Binomial'
print(x, ...)
```

**Arguments**

- `x`                    an object of class UPG.Binomial.
- `...`                other print parameters.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Binomial](#) to summarize a UPG.Binomial object and create tables. [predict.UPG.Binomial](#) to predict probabilities using a UPG.Binomial object. [plot.UPG.Binomial](#) to plot a UPG.Binomial object.

**Examples**

```
# estimate a binomial logit model using example data
library(UPG)
data(titanic)
y = titanic[,1]
Ni = titanic[,2]
X = titanic[,-c(1,2)]
results.binomial = UPG(y = y, X = X, Ni = Ni, model = "binomial")
print(results.binomial)
```

---

print.UPG.Logit	<i>Print information for UPG.Logit objects</i>
-----------------	--

---

**Description**

print provides some basic information about a UPG.Logit object.

**Usage**

```
## S3 method for class 'UPG.Logit'
print(x, ...)
```

**Arguments**

x	an object of class UPG.Logit.
...	other print parameters.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Logit](#) to summarize a UPG.Logit object and create tables. [predict.UPG.Logit](#) to predict probabilities using a UPG.Logit object. [plot.UPG.Logit](#) to plot a UPG.Logit object.

## Examples

```
# estimate a logit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.logit = UPG(y = y, X = X, model = "logit")
print(results.logit)
```

---

print.UPG.MNL	<i>Print information for UPG.MNL objects</i>
---------------	--

---

## Description

print provides some basic information about a UPG.MNL object.

## Usage

```
## S3 method for class 'UPG.MNL'
print(x, ...)
```

## Arguments

x	an object of class UPG.MNL.
...	other print parameters.

## Author(s)

Gregor Zens

## See Also

[summary.UPG.MNL](#) to summarize a UPG.MNL object and create tables. [predict.UPG.MNL](#) to predict probabilities using a UPG.MNL object. [plot.UPG.MNL](#) to plot a UPG.MNL object.

## Examples

```
# estimate a multinomial logit model using example data
library(UPG)
data(program)
y = program[,1]
X = program[,-1]
results.mnl = UPG(y = y, X = X, model = "mnl")
print(results.mnl)
```

---

```
print.UPG.Probit      Print information for UPG.Probit objects
```

---

### Description

`print` provides some basic information about a `UPG.Probit` object.

### Usage

```
## S3 method for class 'UPG.Probit'  
print(x, ...)
```

### Arguments

`x` an object of class `UPG.Probit`.  
`...` other print parameters.

### Author(s)

Gregor Zens

### See Also

[summary.UPG.Probit](#) to summarize a `UPG.Probit` object and create tables. [predict.UPG.Probit](#) to predict probabilities using a `UPG.Probit` object. [plot.UPG.Probit](#) to plot a `UPG.Probit` object.

### Examples

```
# estimate a probit model using example data  
library(UPG)  
data(lfp)  
y = lfp[,1]  
X = lfp[,-1]  
results.probit = UPG(y = y, X = X, model = "probit")  
print(results.probit)
```

---

```
program      Students program choices.
```

---

### Description

A dataset containing the choice among general program, vocational program and academic program for 200 high school students as well as some explanatory variables.



**Usage**

program

**Format**

A data frame with 200 rows and 5 variables:

**program** a vector of program choices

**intercept** an intercept term

**female** binary indicator for female students

**ses** socioeconomic status, 1 is lowest

**write** writing score of student

**Source**

Original dataset is known as 'hsbdemo' and has been sourced from <https://stats.oarc.ucla.edu/stat/data/hsbdemo.dta>.

---

summary.UPG.Binomial *Estimation results and tables for UPG.Binomial objects*

---

**Description**

summary generates a summary of estimation results for UPG.Binomial objects. Point estimates, estimated standard deviation as well as credible intervals for each variable are tabulated. In addition, an indicator quickly shows whether the credible interval includes zero or not. Moreover, LaTeX, HTML and pandoc tables can be quickly generated via knitr.

**Usage**

```
## S3 method for class 'UPG.Binomial'
summary(
  object = NULL,
  ...,
  q = c(0.025, 0.975),
  names = NULL,
  digits = 2,
  include = NULL,
  table = NULL,
  cap = NULL
)
```

**Arguments**

object	an object of class UPG.Binomial.
...	other summary parameters.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.
names	a character vector indicating names for the variables used in the output.
digits	number of digits to be included in output. Last digit will be rounded using round.
include	can be used to summarize and tabulate only a subset of variables. Specify the columns of X that should be kept in the plot. See examples for further information.
table	can be used to return a LaTeX table ('latex'), a Word table ('pandoc') and HTML tables ('html') via knitr. Include package "booktabs" in LaTeX preamble for LaTeX tables.
cap	character vector that can be used to specify the table caption.

**Value**

Returns a knitr\_kable object containing the summary table.

**Author(s)**

Gregor Zens

**See Also**

[plot.UPG.Binomial](#) to plot a UPG.Binomial object. [predict.UPG.Binomial](#) to predict probabilities using a UPG.Binomial object. [coef.UPG.Binomial](#) to extract coefficients from a UPG.Binomial object.

**Examples**

```
# estimate a binomial logit model using example data
library(UPG)
data(titanic)
y = titanic[,1]
Ni = titanic[,2]
X = titanic[,-c(1,2)]
results.binomial = UPG(y = y, X = X, Ni = Ni, model = "binomial")

# basic summary of regression results
summary(results.binomial)

# generate a LaTeX table with subset of variables and custom names
summary(results.binomial,
  include=c(1,3),
  names=c("V. kept 1", "V. kept 3"),
  table="latex")
```

---

summary.UPG.Logit      *Estimation results and tables for UPG.Logit objects*

---

## Description

summary generates a summary of estimation results for UPG.Logit objects. Point estimates, estimated standard deviation as well as credible intervals for each variable are tabulated. In addition, an indicator quickly shows whether the credible interval includes zero or not. Moreover, LaTeX, HTML and pandoc tables can be quickly generated via knitr.

## Usage

```
## S3 method for class 'UPG.Logit'
summary(
  object = NULL,
  ...,
  q = c(0.025, 0.975),
  names = NULL,
  digits = 2,
  include = NULL,
  table = NULL,
  cap = NULL
)
```

## Arguments

object	an object of class UPG.Logit.
...	other summary parameters.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.
names	a character vector indicating names for the variables used in the output.
digits	number of digits to be included in output. Last digit will be rounded using round.
include	can be used to summarize and tabulate only a subset of variables. Specify the columns of X that should be kept in the plot. See examples for further information.
table	can be used to return a LaTeX table ('latex'), a Word table ('pandoc') and HTML tables ('html') via knitr. Include package "booktabs" in LaTeX preamble for LaTeX tables.
cap	character vector that can be used to specify the table caption.

## Value

Returns a knitr\_kable object containing the summary table.

**Author(s)**

Gregor Zens

**See Also**

[plot.UPG.Logit](#) to plot a UPG.Logit object. [predict.UPG.Logit](#) to predict probabilities using a UPG.Logit object. [coef.UPG.Logit](#) to extract coefficients from a UPG.Logit object.

**Examples**

```
# estimate a logit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.logit = UPG(y = y, X = X, model = "logit")

# basic summary of regression results
summary(results.logit)

# generate a LaTeX table with subset of variables and custom names
summary(results.logit,
         include=c(1,3),
         names=c("V. kept 1", "V. kept 3"),
         table="latex")
```

summary.UPG.MNL

*Estimation results and tables for UPG.MNL objects***Description**

summary generates a summary of estimation results for UPG.MNL objects. Point estimates, estimated standard deviation as well as credible intervals for each variable are tabulated. In addition, an indicator quickly shows whether the credible interval includes zero or not. Moreover, LaTeX, HTML and pandoc tables can be quickly generated via knitr.

**Usage**

```
## S3 method for class 'UPG.MNL'
summary(
  object = NULL,
  ...,
  q = c(0.025, 0.975),
  groups = NULL,
  names = NULL,
  digits = 2,
  include = NULL,
```

```

    table = NULL,
    cap = NULL
  )

```

### Arguments

object	an object of class UPG.MNL.
...	other summary parameters.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.
groups	a character vector indicating names for the groups, excluding the baseline. The group names must correspond to the ordering in the dependent variable used for estimation.
names	a character vector indicating names for the variables used in the output.
digits	number of digits to be included in output. Last digit will be rounded using round.
include	can be used to summarize and tabulate only a subset of variables. Specify the columns of X that should be kept in the plot. See examples for further information.
table	can be used to return a LaTeX table ('latex'), a Word table ('pandoc') and HTML tables ('html') via knitr. Include package "booktabs" in LaTeX preamble for LaTeX tables.
cap	character vector that can be used to specify the table caption.

### Value

Returns a knitr\_kable object containing the summary table.

### Author(s)

Gregor Zens

### See Also

[plot.UPG.MNL](#) to plot a UPG.MNL object. [predict.UPG.MNL](#) to predict probabilities using a UPG.MNL object. [coef.UPG.MNL](#) to extract coefficients from a UPG.MNL object.

### Examples

```

# estimate a multinomial logit model using example data
library(UPG)
data(program)
y = program[,1]
X = program[,-1]
results.mnl = UPG(y = y, X = X, model = "mnl")

# basic summary of regression results
summary(results.mnl)

```

```
# generate a LaTeX table with subset of variables and custom names
summary(results.mnl,
        include=c(1,3),
        groups=c("Alpha","Beta"),
        names=c("V. kept 1", "V. kept 3"),
        table="latex")
```

---

summary.UPG.Probit      *Estimation result summary and tables for UPG.Probit objects*

---

## Description

summary generates a summary of estimation results for UPG.Probit objects. Point estimates, estimated standard deviation as well as credible intervals for each variable are tabulated. In addition, an indicator quickly shows whether the credible interval includes zero or not. Moreover, LaTeX, HTML and pandoc tables can be quickly generated via knitr.

## Usage

```
## S3 method for class 'UPG.Probit'
summary(
  object = NULL,
  ...,
  q = c(0.025, 0.975),
  names = NULL,
  digits = 2,
  include = NULL,
  table = NULL,
  cap = NULL
)
```

## Arguments

object	an object of class UPG.Probit.
...	other summary parameters.
q	a numerical vector of length two providing the posterior quantiles to be extracted. Default are 0.025 and 0.975 quantiles.
names	a character vector indicating names for the variables used in the output.
digits	number of digits to be included in output. Last digit will be rounded using round.
include	can be used to summarize and tabulate only a subset of variables. Specify the columns of X that should be kept in the plot. See examples for further information.

table	can be used to return a LaTeX table ('latex'), a Word table ('pandoc') and HTML tables ('html') via knitr. Include package "booktabs" in LaTeX preamble for LaTeX tables.
cap	character vector that can be used to specify the table caption.

**Value**

Returns a `knitr_kable` object containing the summary table.

**Author(s)**

Gregor Zens

**See Also**

[plot.UPG.Probit](#) to plot a UPG.Probit object. [predict.UPG.Probit](#) to predict probabilities using a UPG.Probit object. [coef.UPG.Probit](#) to extract coefficients from a UPG.Probit object.

**Examples**

```
# estimate a probit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.probit = UPG(y = y, X = X, model = "probit")

# basic summary of regression results
summary(results.probit)

# generate a LaTeX table with subset of variables and custom names
summary(results.probit,
         include=c(1,3),
         names=c("V. kept 1", "V. kept 3"),
         table="latex")
```

---

titanic

*Grouped Titanic survival data.*


---

**Description**

A dataset containing the number of survivals and the total number of persons by passenger class, age group and gender.

**Usage**

```
titanic
```

**Format**

A data frame with 78 rows and 6 variables:

**survived** number of passengers that survived

**total** number of total passengers

**intercept** an intercept term

**pclass** passenger class (3 is lowest)

**female** binary indicator for female passenger groups

**age.group** age group indicator (0-5yrs, 5-10yrs, ...)

**Source**

Data originally sourced from <https://web.stanford.edu/class/archive/cs/cs109/cs109.1166/stuff/titanic.csv>. See also <https://towardsdatascience.com/the-binomial-regression-model-everything-you-need-to-know>

---

UPG

*Efficient MCMC Samplers for Bayesian probit regression and various logistic regression models*

---

**Description**

UPG estimates Bayesian regression models for binary or categorical outcomes using samplers based on marginal data augmentation.

**Usage**

```
UPG(y,
    X,
    model,
    Ni      = NULL,
    baseline = NULL,
    draws   = 1000,
    burnin  = 1000,
    A0      = 4,
    B0      = 4,
    d0      = 2.5,
    D0      = 1.5,
    G0      = 100,
    verbose  = TRUE,
    gamma.boost = TRUE,
    delta.boost = TRUE,
    beta.start = NULL)
```



**Arguments**

<code>y</code>	a binary vector for probit and logit models. A character, factor or numeric vector for multinomial logit models. A numerical vector of the number of successes for the binomial model.
<code>X</code>	a matrix of explanatory variables including an intercept in the first column. Rows are individuals, columns are variables.
<code>model</code>	indicates the model to be estimated. 'probit' for the probit model, 'logit' for the logit model, 'mnl' for the multinomial logit model or 'binomial' for the binomial logit model.
<code>Ni</code>	a vector containing the number of trials when estimating a binomial logit model.
<code>baseline</code>	a string that can be used to change the baseline category in MNL models. Default baseline is the most commonly observed category.
<code>draws</code>	number of saved Gibbs sampler iterations. Default is 1000 for illustration purposes, you should use more when estimating a model (e.g. 10,000).
<code>burnin</code>	number of burned Gibbs sampler iterations. Default is 1000 for illustration purposes, you should use more when estimating a model (e.g. 2,000).
<code>A0</code>	prior variance for the intercept, 4 is the default.
<code>B0</code>	prior variance for the coefficients, 4 is the default.
<code>d0</code>	prior shape for working parameter delta, 2.5 is the default.
<code>D0</code>	prior rate for working parameter delta, 1.5 is the default.
<code>G0</code>	prior variance for working parameter gamma, 100 is the default.
<code>verbose</code>	logical variable indicating whether progress should be printed during estimation.
<code>gamma.boost</code>	logical variable indicating whether location-based parameter expansion boosting should be used.
<code>delta.boost</code>	logical variable indicating whether scale-based parameter expansion boosting should be used.
<code>beta.start</code>	provides starting values for beta (e.g. for use within Gibbs sampler). Baseline coefficients need to be zero for multinomial model.

**Value**

Depending on the estimated model, `UPG()` returns a `UPG.Probit`, `UPG.Logit`, `UPG.MNL` or `UPG.Binomial` object.

**Author(s)**

Gregor Zens

**See Also**

[summary.UPG.Probit](#) to summarize a `UPG.Probit` object and to create tables. [predict.UPG.Logit](#) to predict probabilities using a `UPG.Logit` object. [plot.UPG.MNL](#) to plot a `UPG.MNL` object.

**Examples**

```

# load package
library(UPG)

# estimate a probit model using example data
# warning: use more burn-ins, burnin = 100 is just used for demonstration purposes
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.probit = UPG(y = y, X = X, model = "probit", burnin = 100)

# estimate a logit model using example data
# warning: use more burn-ins, burnin = 100 is just used for demonstration purposes
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.logit = UPG(y = y, X = X, model = "logit", burnin = 100)

# estimate a MNL model using example data
# warning: use more burn-ins, burnin = 100 is just used for demonstration purposes
data(program)
y = program[,1]
X = program[,-1]
results.mnl = UPG(y = y, X = X, model = "mnl", burnin = 100)

# estimate a binomial logit model using example data
# warning: use more burn-ins, burnin = 100 is just used for demonstration purposes
data(titanic)
y = titanic[,1]
Ni = titanic[,2]
X = titanic[,-c(1,2)]
results.binomial = UPG(y = y, X = X, Ni = Ni, model = "binomial", burnin = 100)

```

---

UPG.Diag

*MCMC Diagnostics for UPG.Probit, UPG.Logit, UPG.MNL and  
UPG.Binomial objects using coda*


---

**Description**

UPG.Diag computes a number of MCMC diagnostics based on the Markov chains that are contained in the model output returned by UPG.

**Usage**

```
UPG.Diag(object = NULL)
```

**Arguments**

object            an object of class UPG.Probit, UPG.Logit, UPG.MNL or UPG.Binomial.

**Value**

Returns a list containing effective sample size, effective sampling rate and inefficiency factors for each coefficient. In addition, maximum, minimum and median of these measures are returned.

**Author(s)**

Gregor Zens

**Examples**

```
# estimate a probit model using example data
library(UPG)
data(lfp)
y = lfp[,1]
X = lfp[,-1]
results.probit = UPG(y = y, X = X, model = "probit")

# compute MCMC diagnostics
UPG.Diag(results.probit)
```

---

UPG.Diag.Binomial            *MCMC Diagnostics for UPG.Binomial objects*

---

**Description**

UPG.Diag.Binomial computes inefficiency factors, effective sample size and effective sampling rate based on the posterior distributions in a UPG.Binomial object.

**Usage**

```
UPG.Diag.Binomial(object = NULL)
```

**Arguments**

object            an object of class UPG.Binomial.

**Value**

Returns a list containing effective sample size, effective sampling rate and inefficiency factors for each coefficient.

**Author(s)**

Gregor Zens

---

UPG.Diag.Logit            *MCMC Diagnostics for UPG.Logit objects*

---

**Description**

UPG.Diag.Logit computes inefficiency factors, effective sample size and effective sampling rate based on the posterior distributions in a UPG.Logit object.

**Usage**

```
UPG.Diag.Logit(object = NULL)
```

**Arguments**

object            an object of class UPG.Logit.

**Value**

Returns a list containing effective sample size, effective sampling rate and inefficiency factors for each coefficient.

**Author(s)**

Gregor Zens

---

UPG.Diag.MNL            *MCMC Diagnostics for UPG.MNL objects*

---

**Description**

UPG.Diag.MNL computes inefficiency factors, effective sample size and effective sampling rate based on the posterior distributions in a UPG.MNL object.

**Usage**

```
UPG.Diag.MNL(object = NULL)
```

**Arguments**

object            an object of class UPG.MNL.

**Value**

Returns a list containing effective sample size, effective sampling rate and inefficiency factors for each coefficient.

**Author(s)**

Gregor Zens

---

`UPG.Diag.Probit`*MCMC Diagnostics for UPG.Probit objects*

---

**Description**

UPG.Diag.Probit computes inefficiency factors, effective sample size and effective sampling rate based on the posterior distributions in a UPG.Probit object.

**Usage**

```
UPG.Diag.Probit(object = NULL)
```

**Arguments**

`object` an object of class UPG.Probit.

**Value**

Returns a list containing effective sample size, effective sampling rate and inefficiency factors for each coefficient.

**Author(s)**

Gregor Zens

# Index

## \* datasets

lfp, [6](#)  
program, [24](#)  
titanic, [31](#)

coef.UPG.Binomial, [2](#), [8](#), [12](#), [17](#), [26](#)  
coef.UPG.Logit, [3](#), [9](#), [13](#), [19](#), [28](#)  
coef.UPG.MNL, [4](#), [9](#), [15](#), [20](#), [29](#)  
coef.UPG.Probit, [5](#), [10](#), [16](#), [21](#), [31](#)

lfp, [6](#)  
logLik.UPG.Binomial, [7](#)  
logLik.UPG.Logit, [8](#)  
logLik.UPG.MNL, [9](#)  
logLik.UPG.Probit, [10](#)

plot.UPG.Binomial, [3](#), [8](#), [11](#), [17](#), [22](#), [26](#)  
plot.UPG.Logit, [4](#), [9](#), [12](#), [19](#), [22](#), [28](#)  
plot.UPG.MNL, [5](#), [9](#), [14](#), [20](#), [23](#), [29](#), [33](#)  
plot.UPG.Probit, [6](#), [10](#), [15](#), [21](#), [24](#), [31](#)  
predict.UPG.Binomial, [3](#), [12](#), [17](#), [22](#), [26](#)  
predict.UPG.Logit, [4](#), [13](#), [18](#), [22](#), [28](#), [33](#)  
predict.UPG.MNL, [5](#), [15](#), [19](#), [23](#), [29](#)  
predict.UPG.Probit, [6](#), [16](#), [20](#), [24](#), [31](#)  
print.UPG.Binomial, [21](#)  
print.UPG.Logit, [22](#)  
print.UPG.MNL, [23](#)  
print.UPG.Probit, [24](#)  
program, [24](#)

summary.UPG.Binomial, [3](#), [8](#), [12](#), [17](#), [22](#), [25](#)  
summary.UPG.Logit, [4](#), [9](#), [13](#), [19](#), [22](#), [27](#)  
summary.UPG.MNL, [5](#), [9](#), [15](#), [20](#), [23](#), [28](#)  
summary.UPG.Probit, [6](#), [10](#), [16](#), [21](#), [24](#), [30](#), [33](#)

titanic, [31](#)

UPG, [32](#)  
UPG.Diag, [34](#)  
UPG.Diag.Binomial, [35](#)  
UPG.Diag.Logit, [36](#)

UPG.Diag.MNL, [36](#)  
UPG.Diag.Probit, [37](#)