

# Package ‘lcpm’

May 8, 2026

**Type** Package

**Title** Ordinal Outcomes: Generalized Linear Models with the Log Link

**Version** 0.1.1

**Author** Gurbakhshash Singh and Gordon Hilton Fick

**Maintainer** Gurbakhshash Singh <gsingh@ccsu.edu>

**Description** An implementation of the Log Cumulative Probability Model (LCPM) and Proportional Probability Model (PPM) for which the Maximum Likelihood Estimates are determined using constrained optimization.

This implementation accounts for the implicit constraints on the parameter space. Other features such as standard errors, z tests and p-values use standard methods adapted from the results based on constrained optimization.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Depends** plyr (>= 1.8.4), numDeriv (>= 2016.8-1), Matrix (>= 1.2-11), stats (>= 3.4.2)

**RoxygenNote** 7.0.2

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2020-01-09 07:10:02 UTC

## Contents

lcpm . . . . .	2
lcpmMinusloglik . . . . .	4
ppm . . . . .	4
<b>Index</b>	<b>7</b>

## Description

`lcpm` provides the maximum likelihood estimate for ordinal outcomes ( $J > 2$  categories) and a Generalized Linear Model (GLM) with the log link without the assumption of proportionality. That is, `lcpm` determines the MLE for  $\log[P(y \leq j)] = \text{cut}_j + X \text{beta}_j$  subject to  $[\text{cut}_{j-1} + X \text{beta}_{j-1} \leq \text{cut}_j + X \text{beta}_j]$  and  $[\text{cut}_j + X \text{beta}_j \leq 0]$ . This implementation uses `constrOptim` to determine the MLE and so the results account for the restricted parameter space.

## Usage

```
lcpm(
  formula.linear,
  data,
  conf.level = 0.95,
  y.order = NULL,
  startval = NULL,
  less.than.0 = TRUE,
  control.list = NULL,
  eps.outer = NULL,
  ...
)
```

## Arguments

<code>formula.linear</code>	an object of class "formula": a symbolic description of the linear model to be fitted.
<code>data</code>	dataframe containing the data in linear model.
<code>conf.level</code>	optional confidence level (1-alpha) defaulted to 0.95.
<code>y.order</code>	optional if <code>y</code> contains ordered integer categories 1:J. If <code>y</code> is not ordered integer 1:J then this is a vector with the ordinal values for <code>y</code> ranging from the lowest to largest ordinal outcome. See Examples below.
<code>startval</code>	optional vector of the starting values.
<code>less.than.0</code>	optional logical for constraint $\text{cut}_j \leq 0$ for all $j=1:(J-1)$ . Default is TRUE.
<code>control.list</code>	optional list of controls for <code>constrOptim</code>
<code>eps.outer</code>	option for <code>constrOptim</code>
<code>...</code>	Additional arguments for built in functions

## Value

list of class "lcpm" is returned containing:

`coefficients` vector of the estimate of `cut_j` and `beta_j`

se	vector of the estimate of standard errors
vcov	matrix of the inverse of the negative Hessian
fitted.values	matrix of unique covariates and the corresponding estimate of the cumulative probabilities: $\exp(X \%*\% \text{coefficients})$
loglik	numerical value of the log-likelihood at the maximum likelihood estimate
barrier.value	value of mu in the log-barrier algorithm
outer.iterations	value of the number of outer iterations
formula	formula in the call of lcpm
startvalues	vector of the starting values for constrained optimization algorithm
proptest	Score test if a proportionality assumption is appropriate, includes test statistic (teststat), p-value (pval), df, and fitted proportional probability model (propmodel)

### Note

A warning of MLE close to the boundary must be carefully considered. Data may have some structure that requires attention. Additionally, there is no imputation. Any NA results in complete row removal.

### References

Singh, G; Fick, G.H. Ordinal outcomes: a cumulative probability model with the log link without an assumption of proportionality. Manuscript in preparation.

### See Also

[ppm](#)

### Examples

```
# Example below showing the use of y.order if outcome is not integers 1:J.
# See examples in ppm for an additional example
```

```
var_a <- c(rep(0,60),rep(1,60))
var_b <- c(rep(0,90),rep(1,30))
y1<-c(rep(2,5),rep(3,10),rep(5,5),rep(10,10),
rep(2,5),rep(3,10),rep(5,10),rep(10,5),
rep(2,10),rep(3,5),rep(5,5),rep(10,10),
rep(2,10),rep(3,5),rep(5,10),rep(10,5))
```

```
testdata<-data.frame(y=y1,var_a=var_a,var_b=var_b)
```

```
# LCPM estimates for non-proportional model
test1<-lcpm(y ~ var_a + var_b, data=testdata, y.order=c(2,3,5,10))
summary(test1)
```

```
# The proportional probability model used for the score test
summary(test1$proptest$propmodel)
```

---

lcpmMinusloglik	<i>The Negative Log-Likelihood Function for a Log Cumulative Probability Model</i>
-----------------	--

---

### Description

lcpmMinusloglik provides the negative of the log-likelihood function for a Generalized Linear Model with a log link and ordinal outcomes to be minimized in functions [lcpm](#) and [ppm](#).

### Usage

```
lcpmMinusloglik(betapar, Xa1, XaJ, Xaj1, Xaj2)
```

### Arguments

betapar	a vector of values.
Xa1	matrix of covariates for all subjects with the lowest ordinal outcome value 1.
XaJ	matrix of covariates for all subjects with the largest ordinal outcome value J.
Xaj1	matrix of covariates for all subjects with the ordinal outcomes with value $1 < j < J$ .
Xaj2	matrix of covariates for all subjects with the ordinal outcome with value $1 < j < J$ but lagged by 1.

### Value

value of the negative log-likelihood evaluated at betapar

---

ppm	<i>Fitting a Proportional Probability Model</i>
-----	---

---

### Description

ppm provides the maximum likelihood estimate for ordinal outcomes ( $J > 2$  categories) and a Generalized Linear Model with the log link with the assumption of proportionality. That is, ppm determines the MLE for  $\log[P(y \leq j)] = \text{cut}_j + X \beta$  subject to  $[\text{cut}_{j-1} \leq \text{cut}_j]$  and  $[\text{cut}_j + X \beta \leq 0]$ . This implementation uses [constrOptim](#) to determine the MLE and so the results should correctly account for the restricted parameter space. A proposed test for proportionality is included in [lcpm](#).

**Usage**

```
ppm(
  formula.linear,
  data,
  conf.level = 0.95,
  y.order = NULL,
  startval = NULL,
  less.than.0 = TRUE,
  control.list = NULL,
  eps.outer = NULL,
  ...
)
```

**Arguments**

<code>formula.linear</code>	an object of class "formula": a symbolic description of the linear model to be fitted.
<code>data</code>	dataframe containing the data in linear model.
<code>conf.level</code>	optional confidence level (1-alpha) defaulted to 0.95.
<code>y.order</code>	optional if y contains ordered integer categories 1:J. If y is not ordered integer 1:J then this is a vector with the ordinal values for y ranging from the lowest to largest ordinal outcome. See Examples below.
<code>startval</code>	optional vector of the starting values.
<code>less.than.0</code>	optional logical for constraint $\text{cut}_j \leq 0$ for all $j=1:(J-1)$ . Default is TRUE.
<code>control.list</code>	optional list of controls for <code>constrOptim</code> .
<code>eps.outer</code>	option for <code>constrOptim</code> .
<code>...</code>	Additional arguments for built in functions.

**Value**

list of class "ppm" is returned containing:

<code>coefficients</code>	vector of the estimate of $\text{cut}_j$ and beta
<code>se</code>	vector of the estimate of standard errors
<code>vcov</code>	matrix of the inverse of the negative Hessian
<code>fitted.values</code>	matrix of unique covariates and the corresponding estimate of the cumulative probabilities: $\exp(X \%*\% \text{coefficients})$
<code>loglik</code>	numerical value of the log-likelihood at the maximum likelihood estimate
<code>barrier.value</code>	value of mu in the log-barrier algorithm
<code>outer.iterations</code>	value of the number of outer iterations
<code>formula</code>	formula in the call of ppm
<code>startvalues</code>	vector of the starting values for constrained optimization algorithm

**Note**

A warning of MLE close to the boundary must be carefully considered. Data may have some structure that requires attention. Additionally, there is no imputation. Any NA results in complete row removal.

**References**

Singh, G; Fick, G.H. (accepted) Ordinal outcomes: a cumulative probability model with the log link and an assumption of proportionality. *Statistics in Medicine*.

**See Also**

[lcpm](#)

**Examples**

```
# 2 examples below showing the use of y.order if outcome are not integers 1:J.
```

```
# Example 1:
```

```
var_a <- c(rep(0,60),rep(1,60))
var_b <- c(rep(0,90),rep(1,30))
y1<-c(rep(2,5),rep(3,10),rep(5,5),rep(10,10),
rep(2,5),rep(3,10),rep(5,10),rep(10,5),
rep(2,10),rep(3,5),rep(5,5),rep(10,10),
rep(2,10),rep(3,5),rep(5,10),rep(10,5))
```

```
testdata<-data.frame(y=y1,var_a=var_a,var_b=var_b)
```

```
# PPM estimates for proportional model
```

```
test1<-ppm( y ~ var_a + var_b, data=testdata, y.order=c(2,3,5,10))
summary(test1)
```

```
# Example 2:
```

```
y2<-c(rep("a",5),rep("b",10),rep("c",5),rep("d",10),
rep("a",5),rep("b",10),rep("c",10),rep("d",5),
rep("a",10),rep("b",5),rep("c",5),rep("d",10),
rep("a",10),rep("b",5),rep("c",10),rep("d",5))
testdata2<-data.frame(y=y2,var_a=var_a,var_b=var_b)
test2<-ppm(y~var_a + var_b , data=testdata2, y.order=c("a","b","c","d"))
summary(test2)
```

# Index

constrOptim, 2, 4

lcpm, 2, 4, 6

lcpmMinusloglik, 4

ppm, 3, 4, 4