

# Package ‘ContRespPP’

May 7, 2026

**Title** Predictive Probability for a Continuous Response with an ANOVA Structure

**Version** 0.4.2

**Description** A Bayesian approach to using predictive probability in an ANOVA construct with a continuous normal response, when threshold values must be obtained for the question of interest to be evaluated as successful (Sieck and Christensen (2021) <[doi:10.1002/qre.2802](https://doi.org/10.1002/qre.2802)>). The Bayesian Mission Mean (BMM) is used to evaluate a question of interest (that is, a mean that randomly selects combination of factor levels based on their probability of occurring instead of averaging over the factor levels, as in the grand mean). Under this construct, in contrast to a Gibbs sampler (or Metropolis-within-Gibbs sampler), a two-stage sampling method is required. The nested sampler determines the conditional posterior distribution of the model parameters, given Y, and the outside sampler determines the marginal posterior distribution of Y (also commonly called the predictive distribution for Y). This approach provides a sample from the joint posterior distribution of Y and the model parameters, while also accounting for the threshold value that must be obtained in order for the question of interest to be evaluated as successful.

**URL** <https://github.com/jcliff89/ContRespPP>

**BugReports** <https://github.com/jcliff89/ContRespPP/issues>

**License** CC0

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.1.1

**Imports** stats

**Suggests** rjags, coda, knitr, devtools, rmarkdown, testthat (>= 3.0.0)

**VignetteBuilder** knitr

**Depends** R (>= 2.10)

**Config/testthat/edition** 3

**NeedsCompilation** no

**Author** Victoria Sieck [aut, cre] (ORCID: <https://orcid.org/0000-0002-7169-7795>),  
 Joshua Clifford [aut] (ORCID: <https://orcid.org/0000-0002-6969-2581>),  
 Fletcher Christensen [aut] (ORCID: <https://orcid.org/0000-0002-5303-8139>)

**Maintainer** Victoria Sieck <vcarrillo314@gmail.com>

**Repository** CRAN

**Date/Publication** 2022-10-15 22:45:27 UTC

## Contents

exData . . . . .	2
gibbs.sampler . . . . .	3
gibbs.sampler.posterior . . . . .	5
gibbs.sampler.posterior.rjags . . . . .	6
gibbs.sampler.predictive . . . . .	8
gibbs.sampler.predictive.rjags . . . . .	10
print.ContRespPP . . . . .	12
prob.creator . . . . .	12
summary.ContRespPP . . . . .	13

**Index** 14

---

exData	<i>Example Continuous Response ANOVA Dataset.</i>
--------	---

---

## Description

A dataset containing an example test set result to run a predictive probability model with a continuous response variable. This data set is a result of a  $2^4$  full factorial with five replicates that included five main effects, with two-way interactions (excluding two-way interactions for the third factor). This design has a power of 80%, with an 80% confidence level, to detect a difference of 50 units with a standard deviation of 100 units.

## Usage

exData

## Format

A matrix with 80 rows and 14 columns:

**Column 1** Continuous Response Variable

**Columns 2-14** Design Matrix

**Details**

Please refer to the vignette for an example analysis of this data using ContRespPP.

**Source**

Sieck VRC, Christensen FGW. A framework for improving the efficiency of operational testing through Bayesian adaptive design. *Quality and Reliability Engineering International*. 2021; 3018-3033.

---

`gibbs.sampler`*Continuous Response Predictive Probability.*

---

**Description**

`gibbs.sampler` runs the estimation of the continuous response ANOVA predictive probability.

**Usage**

```
gibbs.sampler(  
  X,  
  Y,  
  n.seen,  
  beta.mean,  
  beta.precision,  
  shape,  
  rate,  
  n.sim,  
  y.burnin,  
  b.sim,  
  b.burnin,  
  phi.0,  
  theta.t,  
  prob,  
  factor.no.2way = NA,  
  colnames.pick = NA,  
  seed = NA,  
  verbose = TRUE  
)
```

**Arguments**

X	Design matrix for the test (matrix of indicator functions defining which model parameters are active in each test event).
Y	A vector of the responses from the test.
n.seen	Number of test events already observed (i.e., the number of rows of the design matrix, X, that have been observed).

beta.mean	Mean vector of the multivariate normal distribution (or the mean of each of the priors on the model parameters), ordered the same as the columns of the design matrix, X. It also serves as the initialization for the model parameters.
beta.precision	Precisions of the multivariate normal distribution (precision of each of the priors on the model parameters), corresponding to the beta.mean values.
shape	Hyperparameter alpha for gamma prior on the precision of the ANOVA model, tau.
rate	Hyperparameter beta for gamma prior on the precision of the ANOVA model, tau.
n.sim	Number of non-conditional posterior draws (i.e., number of draws that will be returned to the user from the function after burn-in draws for the non-conditional draws are removed).
y.burnin	Number of burn-in samples for the non-conditional posterior.
b.sim	Number of conditional posterior draws used in analysis for each non-conditional draw.
b.burnin	Number of burn-in samples for the conditional posterior.
phi.0	Threshold value the parameter of interest (BMM) must obtain (i.e., BMM > phi.0).
theta.t	Certainty threshold for the conditional posterior probability of the parameter of interest (the Bayesian mission mean, "BMM") obtaining phi.0 (i.e., BMM > phi.0) that the conditional posterior probability must obtain (the certainty threshold for conditional P(BMM > phi.0) ) must obtain for the question of interest to be evaluated as successfully passing the test.
prob	Matrix or dataframe of the "likelihood of encountering" (or probability of seeing a factor level); it is a two column matrix (or dataframe), where the first column identifies the factor numerically and the second column defines the probability of seeing each factor level.
factor.no.2way	Optional vector of model parameters (as defined by prob) that are not incorporated in the two way interactions for the model.
colnames.pick	Optional vector of model parameter names in the same order as in the design matrix to label the returned dataframe columns.
seed	Optional selection which will create a reproducible result from the function.
verbose	Allows suppression of sampler progress printing in console.

### Details

The ANOVA model includes main effects and two-way interactions. Priors on model parameters are assumed to be independent of each other; beta is then defined as the set of model parameters, which is multivariate normal.

### Value

Returns a list with three elements:

pp The predicted probability of the test ending in a successful evaluation of the question of interest

posterior The full dataframe of non-conditional posterior draws  
 indicator The vector of test success results for each posterior draw  
 Printing the result object will display the predicted probability result.

---

gibbs.sampler.posterior

*Continuous Response Posterior Probability.*

---

## Description

gibbs.sampler.posterior obtains the ContRespPP method for obtaining the posterior (i.e., all observations have been seen, and this reverts to a traditional Bayesian analysis) using base R functions, drastically increasing the computational time required to obtain predictive draws.

## Usage

```
gibbs.sampler.posterior(  
  X,  
  Y,  
  beta.mean,  
  beta.precision,  
  shape,  
  rate,  
  b.sim,  
  b.burnin,  
  phi.0,  
  prob,  
  factor.no.2way = NA,  
  colnames.pick = NA,  
  seed = NA,  
  verbose = TRUE  
)
```

## Arguments

X	Design matrix for the test (matrix of indicator functions defining which model parameters are active in each test event).
Y	A vector of the responses from the test.
beta.mean	Mean vector of the multivariate normal distribution (or the mean of each of the priors on the model parameters), ordered the same as the columns of the design matrix, X. It also serves as the initialization for the model parameters.
beta.precision	Precisions of the multivariate normal distribution (precision of each of the priors on the model parameters), corresponding to the beta.mean values.
shape	Hyperparameter alpha for gamma prior on the precision of the ANOVA model, tau.

<code>rate</code>	Hyperparameter beta for gamma prior on the precision of the ANOVA model, tau.
<code>b.sim</code>	Number of conditional posterior draws used in analysis for each non-conditional draw.
<code>b.burnin</code>	Number of burn-in samples for the conditional posterior.
<code>phi.0</code>	Threshold value the parameter of interest (BMM) must obtain (i.e., $BMM > \text{phi.0}$ ).
<code>prob</code>	Matrix or dataframe of the "likelihood of encountering" (or probability of seeing a factor level); it is a two column matrix (or dataframe), where the first column identifies the factor numerically and the second column defines the probability of seeing each factor level.
<code>factor.no.2way</code>	Optional vector of model parameters (as defined by <code>prob</code> ) that are not incorporated in the two way interactions for the model.
<code>colnames.pick</code>	Optional vector of model parameter names in the same order as in the design matrix to label the returned dataframe columns.
<code>seed</code>	Optional selection which will create a reproducible result from the function.
<code>verbose</code>	Allows suppression of sampler progress printing in console.

**Value**

Returns a list with three elements:

`pp` This value will be NA since this function only calculates the posterior

`posterior` The full dataframe of non-conditional posterior draws

`indicator` This value will be NA since this function only calculates the posterior

Printing the result object will display the predicted probability result.

---

`gibbs.sampler.posterior.rjags`

*Continuous Response Posterior Probability.*

---

**Description**

`gibbs.sampler.posterior.rjags` obtains the `ContRespPP` method for obtaining the posterior (i.e., all observations have been seen, and this reverts to a traditional Bayesian analysis) using `rjags` to generate samples instead of base R functions, drastically decreasing the computational time required to obtain predictive draws.

**Usage**

```

gibbs.sampler.posterior.rjags(
  X,
  Y,
  beta.mean,
  beta.precision,
  shape,
  rate,
  b.sim,
  b.burnin,
  phi.0,
  prob,
  factor.no.2way = NA,
  colnames.pick = NA,
  seed = NA,
  verbose = TRUE
)

```

**Arguments**

X	Design matrix for the test (matrix of indicator functions defining which model parameters are active in each test event).
Y	A vector of the responses from the test.
beta.mean	Mean vector of the multivariate normal distribution (or the mean of each of the priors on the model parameters), ordered the same as the columns of the design matrix, X. It also serves as the initialization for the model parameters.
beta.precision	Precisions of the multivariate normal distribution (precision of each of the priors on the model parameters), corresponding to the beta.mean values.
shape	Hyperparameter alpha for gamma prior on the precision of the ANOVA model, tau.
rate	Hyperparameter beta for gamma prior on the precision of the ANOVA model, tau.
b.sim	Number of conditional posterior draws used in analysis for each non-conditional draw.
b.burnin	Number of burn-in samples for the conditional posterior.
phi.0	Threshold value the parameter of interest (BMM) must obtain (i.e., $BMM > \text{phi.0}$ ).
prob	Matrix or dataframe of the "likelihood of encountering" (or probability of seeing a factor level); it is a two column matrix (or dataframe), where the first column identifies the factor numerically and the second column defines the probability of seeing each factor level.
factor.no.2way	Optional vector of model parameters (as defined by prob) that are not incorporated in the two way interactions for the model.
colnames.pick	Optional vector of model parameter names in the same order as in the design matrix to label the returned dataframe columns.

seed            Optional selection which will create a reproducible result from the function.  
verbose        Allows suppression of sampler progress printing in console.

### Value

Returns a list with three elements:

pp This value will be NA since this function only calculates the posterior

posterior The full dataframe of non-conditional posterior draws

indicator This value will be NA since this function only calculates the posterior

Printing the result object will display the predicted probability result.

---

gibbs.sampler.predictive

*Continuous Response Predictive Probability.*

---

### Description

gibbs.sampler.predictive does the predictive ContRespPP method using base R functions, which drastically increases the computational time required to obtain predictive draws.

### Usage

```
gibbs.sampler.predictive(  
  X,  
  Y,  
  n.seen,  
  beta.mean,  
  beta.precision,  
  shape,  
  rate,  
  n.sim,  
  y.burnin,  
  b.sim,  
  b.burnin,  
  phi.0,  
  theta.t,  
  prob,  
  factor.no.2way = NA,  
  colnames.pick = NA,  
  seed = NA,  
  verbose = TRUE  
)
```

**Arguments**

<code>X</code>	Design matrix for the test (matrix of indicator functions defining which model parameters are active in each test event).
<code>Y</code>	A vector of the responses from the test.
<code>n.seen</code>	Number of test events already observed (i.e., the number of rows of the design matrix, <code>X</code> , that have been observed).
<code>beta.mean</code>	Mean vector of the multivariate normal distribution (or the mean of each of the priors on the model parameters), ordered the same as the columns of the design matrix, <code>X</code> . It also serves as the initialization for the model parameters.
<code>beta.precision</code>	Precisions of the multivariate normal distribution (precision of each of the priors on the model parameters), corresponding to the <code>beta.mean</code> values.
<code>shape</code>	Hyperparameter alpha for gamma prior on the precision of the ANOVA model, tau.
<code>rate</code>	Hyperparameter beta for gamma prior on the precision of the ANOVA model, tau.
<code>n.sim</code>	Number of non-conditional posterior draws (i.e., number of draws that will be returned to the user from the function after burn-in draws for the non-conditional draws are removed).
<code>y.burnin</code>	Number of burn-in samples for the non-conditional posterior.
<code>b.sim</code>	Number of conditional posterior draws used in analysis for each non-conditional draw.
<code>b.burnin</code>	Number of burn-in samples for the conditional posterior.
<code>phi.0</code>	Threshold value the parameter of interest (BMM) must obtain (i.e., $BMM > \text{phi}.0$ ).
<code>theta.t</code>	Certainty threshold for the conditional posterior probability of the parameter of interest (the Bayesian mission mean, "BMM") obtaining <code>phi.0</code> (i.e., $BMM > \text{phi}.0$ ) that the conditional posterior probability must obtain (the certainty threshold for conditional $P(BMM > \text{phi}.0)$ ) must obtain for the question of interest to be evaluated as successfully passing the test.
<code>prob</code>	Matrix or dataframe of the "likelihood of encountering" (or probability of seeing a factor level); it is a two column matrix (or dataframe), where the first column identifies the factor numerically and the second column defines the probability of seeing each factor level.
<code>factor.no.2way</code>	Optional vector of model parameters (as defined by <code>prob</code> ) that are not incorporated in the two way interactions for the model.
<code>colnames.pick</code>	Optional vector of model parameter names in the same order as in the design matrix to label the returned dataframe columns.
<code>seed</code>	Optional selection which will create a reproducible result from the function.
<code>verbose</code>	Allows suppression of sampler progress printing in console.

**Value**

Returns a list with three elements:

pp The predicted probability of the test ending in a successful evaluation of the question of interest

posterior The full dataframe of non-conditional posterior draws

indicator The vector of test success results for each posterior draw

Printing the result object will display the predicted probability result.

---

gibbs.sampler.predictive.rjags

*Continuous Response Predictive Probability.*

---

**Description**

gibbs.sampler.predictive.jags does the predictive ContRespPP method using rjags to generate samples instead of base R functions, drastically decreasing the computational time required to obtain predictive draws.

**Usage**

```
gibbs.sampler.predictive.rjags(  
  X,  
  Y,  
  n.seen,  
  beta.mean,  
  beta.precision,  
  shape,  
  rate,  
  n.sim,  
  y.burnin,  
  b.sim,  
  b.burnin,  
  phi.0,  
  theta.t,  
  prob,  
  factor.no.2way = NA,  
  colnames.pick = NA,  
  seed = NA,  
  verbose = TRUE  
)
```

**Arguments**

<code>X</code>	Design matrix for the test (matrix of indicator functions defining which model parameters are active in each test event).
<code>Y</code>	A vector of the responses from the test.
<code>n.seen</code>	Number of test events already observed (i.e., the number of rows of the design matrix, <code>X</code> , that have been observed).
<code>beta.mean</code>	Mean vector of the multivariate normal distribution (or the mean of each of the priors on the model parameters), ordered the same as the columns of the design matrix, <code>X</code> . It also serves as the initialization for the model parameters.
<code>beta.precision</code>	Precisions of the multivariate normal distribution (precision of each of the priors on the model parameters), corresponding to the <code>beta.mean</code> values.
<code>shape</code>	Hyperparameter alpha for gamma prior on the precision of the ANOVA model, tau.
<code>rate</code>	Hyperparameter beta for gamma prior on the precision of the ANOVA model, tau.
<code>n.sim</code>	Number of non-conditional posterior draws (i.e., number of draws that will be returned to the user from the function after burn-in draws for the non-conditional draws are removed).
<code>y.burnin</code>	Number of burn-in samples for the non-conditional posterior.
<code>b.sim</code>	Number of conditional posterior draws used in analysis for each non-conditional draw.
<code>b.burnin</code>	Number of burn-in samples for the conditional posterior.
<code>phi.0</code>	Threshold value the parameter of interest (BMM) must obtain (i.e., $BMM > \text{phi}.0$ ).
<code>theta.t</code>	Certainty threshold for the conditional posterior probability of the parameter of interest (the Bayesian mission mean, "BMM") obtaining <code>phi.0</code> (i.e., $BMM > \text{phi}.0$ ) that the conditional posterior probability must obtain (the certainty threshold for conditional $P(BMM > \text{phi}.0)$ ) must obtain for the question of interest to be evaluated as successfully passing the test.
<code>prob</code>	Matrix or dataframe of the "likelihood of encountering" (or probability of seeing a factor level); it is a two column matrix (or dataframe), where the first column identifies the factor numerically and the second column defines the probability of seeing each factor level.
<code>factor.no.2way</code>	Optional vector of model parameters (as defined by <code>prob</code> ) that are not incorporated in the two way interactions for the model.
<code>colnames.pick</code>	Optional vector of model parameter names in the same order as in the design matrix to label the returned dataframe columns.
<code>seed</code>	Optional selection which will create a reproducible result from the function.
<code>verbose</code>	Allows suppression of sampler progress printing in console.

**Value**

Returns a list with three elements:

pp The predicted probability of the test ending in a successful evaluation of the question of interest

posterior The full dataframe of non-conditional posterior draws

indicator The vector of test success results for each posterior draw

Printing the result object will display the predicted probability result.

---

```
print.ContRespPP      Print method for ContRespPP class
```

---

**Description**

Print method for ContRespPP class

**Usage**

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

**Arguments**

x                    A list of class ContRespPP  
 ...                  Other arguments passed to print

**Value**

Prints the predictive or posterior probability result from ContRespPP class

---

```
prob.creator          Probability Matrix Creator.
```

---

**Description**

prob.creator creates a matrix of probabilities of encountering factor levels to support construction of input for gibbs.sampler, which uses the Bayesian Mission Mean approach to analysis.

**Usage**

```
prob.creator(
  num.factors,
  num.factor.levels,
  likelihood.encountering,
  print.result = FALSE
)
```

**Arguments**

- num.factors      Number of factors in the model (e.g., factor  $\alpha_i$  with levels i=1,2 is 1 factor). Input for num.factors should be a single number.
- num.factor.levels      Number of levels for each factor (e.g., factor  $\alpha_i$  with i=1,2 has 2 levels). Input for num.factor.levels may be a vector, a matrix, or a dataframe.
- likelihood.encountering      The probability of seeing each level of each factor (e.g., if the factor levels for  $\alpha_i$  are equally likely, then the probabilities would be c(1/2, 1/2)). The probabilities for each factor should sum to one. Input for likelihood.encountering may be a vector, a matrix, or a dataframe.
- print.result      Displays final probability matrix.

**Details**

The ANOVA model includes main effects and two-way interactions. Priors on model parameters are assumed to be independent of each other; beta is then defined as the set of model parameters, which is multivariate normal.

**Value**

Returns a matrix with two columns, one with the factor number and the other with the likelihoods of encountering.

---

summary.ContRespPP      *Summary method for ContRespPP class*

---

**Description**

Summary method for ContRespPP class

**Usage**

```
## S3 method for class 'ContRespPP'
summary(object, ...)
```

**Arguments**

- object      A list of class ContRespPP
- ...      Other arguments passed to summary

**Value**

Prints a basic summary of posterior distribution from ContRespPP class

# Index

## \* datasets

exData, [2](#)

exData, [2](#)

gibbs.sampler, [3](#)

gibbs.sampler.posterior, [5](#)

gibbs.sampler.posterior.rjags, [6](#)

gibbs.sampler.predictive, [8](#)

gibbs.sampler.predictive.rjags, [10](#)

print.ContRespPP, [12](#)

prob.creator, [12](#)

summary.ContRespPP, [13](#)