

# Package: fabPrediction (via r-universe)

August 24, 2024

**Title** Compute FAB (Frequentist and Bayes) Conformal Prediction Intervals

**Version** 1.0.4

**Description** Computes and plots prediction intervals for numerical data or prediction sets for categorical data using prior information. Empirical Bayes procedures to estimate the prior information from multi-group data are included. See, e.g., Bersson and Hoff (2022) <[arXiv:2204.08122](https://arxiv.org/abs/2204.08122)> "Optimal Conformal Prediction for Small Areas".

**License** GPL-3

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Imports** sae (>= 1.3), parallel, stats, graphics

**Suggests** knitr, devtools

**NeedsCompilation** yes

**ByteCompile** yes

**VignetteBuilder** knitr

**URL** <https://github.com/betsybersson/fabPrediction>

**BugReports** <https://github.com/betsybersson/fabPrediction/issues>

**Repository** <https://betsybersson.r-universe.dev>

**RemoteUrl** <https://github.com/betsybersson/fabprediction>

**RemoteRef** HEAD

**RemoteSha** 31cbcd68f34fa6ab7102c343f72bae686da94412

## Contents

bayesMultinomialPrediction . . . . .	2
bayesNormalPrediction . . . . .	3
dtaPrediction . . . . .	3
eye . . . . .	4

fabCategoricalPrediction . . . . .	4
fabContinuousPrediction . . . . .	5
fabPrediction . . . . .	5
fayHerriotEB . . . . .	6
initMoM . . . . .	6
normalPrediction . . . . .	7
plot.pred . . . . .	7
pluginValues . . . . .	8
polyaGradient . . . . .	8
polyaHessian . . . . .	9
polyaMLE . . . . .	9
predictionInterval . . . . .	10
predictionSet . . . . .	11
radon . . . . .	12
rdirichlet . . . . .	12
row_standardize . . . . .	13
W . . . . .	13

## Index 14

---

bayesMultinomialPrediction

*Obtain a Bayesian prediction interval for categorical data*

---

### Description

This function computes the Bayesian prediction set for a multinomial conjugate family.

### Usage

```
bayesMultinomialPrediction(
  Y,
  alpha = 0.15,
  gamma = rep(1, length(Y)),
  category_names = 1:length(Y)
)
```

### Arguments

Y	Observed data vector of length K containing counts of observations from each of the K categories
alpha	Prediction mis-coverage rate
gamma	Dirichlet prior concentration for the K categories
category_names	Category names (optional)

### Value

pred object

---

bayesNormalPrediction *Obtain a Bayesian prediction interval*

---

### Description

This function computes a Bayesian prediction interval based on a normal model.

### Usage

```
bayesNormalPrediction(Y, alpha = 0.15, mu = 0, tau2 = 1)
```

### Arguments

Y	Observed data vector
alpha	Prediction error rate
mu	Prior expected mean of the population mean
tau2	Prior expected variance of the population mean

### Value

pred object

---

dtaPrediction *Obtain a distance-to-average conformal prediction interval*

---

### Description

This function computes a conformal prediction region under the distance-from-average non-conformity measure. That is,  $|a + bz^*| \leq |c_i + d_i z^*|$  where  $i$  indexes training data.

### Usage

```
dtaPrediction(Y, alpha = 0.15)
```

### Arguments

Y	Observed data vector
alpha	Prediction error rate

### Value

pred object

---

eye *Create Identity Matrix*

---

**Description**

This function returns an NxN identity matrix.

**Usage**

```
eye(N)
```

**Arguments**

N                    dimension of square matrix

**Value**

NxN identity matrix

---

fabCategoricalPrediction

*Obtain a FAB conformal prediction interval for categorical data*

---

**Description**

This function computes a FAB conformal prediction set as described in Bersson and Hoff 2023.

**Usage**

```
fabCategoricalPrediction(  
  Y,  
  alpha = 0.15,  
  gamma = rep(1, length(Y)),  
  category_names = 1:length(Y)  
)
```

**Arguments**

Y                    Observed data vector of length K containing counts of observations from each of the K categories

alpha                Prediction mis-coverage rate

gamma                Dirichlet prior concentration for the K categories

category\_names      Category names (optional)

**Value**

pred object

---

`fabContinuousPrediction`*Obtain a FAB conformal prediction interval*

---

**Description**

This function computes a FAB conformal prediction region as described in Bersson and Hoff 2022.

**Usage**

```
fabContinuousPrediction(Y, alpha = 0.15, mu = 0, tau2 = 1)
```

**Arguments**

Y	Observed data vector
alpha	Prediction error rate
mu	Prior expected mean of the population mean
tau2	Prior expected variance of the population mean

**Value**

pred object

---

`fabPrediction`*fabPrediction: Compute FAB Conformal Prediction Intervals*

---

**Description**

A package for computing and plotting prediction intervals for numerical data or prediction sets for categorical data using prior information. Empirical Bayes procedures to estimate the prior information from multi-group data are included.

**Author(s)**

Elizabeth Bersson,  
Maintainer: Elizabeth Bersson <elb75@duke.edu>

**References**

E. Bersson and P.D. Hoff. (2023) Frequentist Prediction Sets for Species Abundance using Indirect Information. Preprint.

E. Bersson and P.D. Hoff. (2023) Optimal Conformal Prediction for Small Areas. Journal of Survey Statistics and Methodology, forthcoming.

---

fayHerriotEB	<i>Obtain empirical Bayesian estimates for group j</i>
--------------	--

---

**Description**

This function returns empirical Bayesian estimates for a specified group from the conjugate normal spatial Fay-Herriot model.

**Usage**

```
fayHerriotEB(j, Y, group, W = NA, X = NA)
```

**Arguments**

j	Obtain EB values for group in index j- numeric value in group
Y	Data vector
group	index vector of the same length as Y
W	Non-standardized adjacency matrix
X	Group-level covariates

**Value**

empirical Bayesian estimates of population mean and it's variance

---

initMoM	<i>Obtain initial guess of MLE of the marginal Dirichlet-multinomial likelihood</i>
---------	---

---

**Description**

Method of moment matching to obtain an initial guess of the MLE, as in Minka (2000).

**Usage**

```
initMoM(D)
```

**Arguments**

D	matrix (JxK) of counts; each row is a sample from a MN distribution with K categories
---	---

**Value**

Hessian

---

normalPrediction	<i>Obtain a pivot prediction interval</i>
------------------	---

---

**Description**

This function computes a prediction interval under assumed normality.

**Usage**

```
normalPrediction(Y, alpha = 0.15)
```

**Arguments**

Y	Observed data vector
alpha	Prediction error rate

**Value**

pred object

---

plot.pred	<i>Plot a 'pred' object constructed for a categorical response</i>
-----------	--

---

**Description**

Plot a 'pred' object constructed for a categorical response

**Usage**

```
## S3 method for class 'pred'
plot(x, ...)
```

**Arguments**

x	pred object- a list classified as pred containing objects data and bound
...	additional parameters passed to the default plot method

**Value**

capability to plot pred object. More details: the command 'plot(obj)' plots the empirical densities of each category. Mass denoted in red indicates inclusion in the prediction set

---

pluginValues	<i>Obtain empirical Bayesian estimates for conjugate normal spatial Fay-Herriot model</i>
--------------	---

---

**Description**

This function returns plug-in values for a conjugate normal spatial Fay-Herriot model.

**Usage**

```
pluginValues(Y, group, W = NA, X = NA)
```

**Arguments**

Y	Data vector
group	Group membership of each entry in Y
W	Adjacency matrix
X	Group-level covariates

**Value**

plug-in values of spatial Fay-Herriot model

---

polyaGradient	<i>Obtain gradient of the marginal Dirichlet-multinomial likelihood</i>
---------------	---

---

**Description**

Obtain gradient of the marginal Dirichlet-multinomial likelihood

**Usage**

```
polyaGradient(D, gamma, Nj = rowSums(D), K = ncol(D))
```

**Arguments**

D	matrix (JxK) of counts; each row is a sample from a MN distribution with K categories
gamma	current value of prior concentration parameter
Nj	sample sizes of the J groups
K	number of categories

**Value**

gradient



---

polyaHessian	<i>Obtain Hessian of the marginal Dirichlet-multinomial likelihood</i>
--------------	--

---

**Description**

Obtain Hessian of the marginal Dirichlet-multinomial likelihood

**Usage**

```
polyaHessian(D, gamma, Nj = rowSums(D), K = ncol(D))
```

**Arguments**

D	matrix (JxK) of counts; each row is a sample from a MN distribution with K categories
gamma	current value of prior concentration parameter
Nj	sample sizes of the J groups
K	number of categories

**Value**

Hessian

---

polyaMLE	<i>Obtain MLE of marginal Dirichlet-multinomial likelihood</i>
----------	--

---

**Description**

This function returns the MLE of the prior concentration from a marginal Dirichlet-multinomial likelihood. Default method iterates a Newton-Raphson algorithm until convergence.

**Usage**

```
polyaMLE(  
  D,  
  init = NA,  
  method = "Newton_Raphson",  
  epsilon = 1e-04,  
  print_progress = FALSE  
)
```

**Arguments**

D	matrix (JxK) of counts; each row is a sample from a MN distribution with K categories
init	If NA, use method moment matching procedure to obtain good init values
method	"Newton_Raphson", "fixed_point", "separate", "precision_only"
epsilon	convergence diagnostic
print_progress	if TRUE, print progress to screen

**Value**

mle of prior concentration from marginal Dirichlet-multinomial likelihood

---

predictionInterval      *Wrapper to obtain a prediction interval for continuous data*

---

**Description**

This function computes a prediction interval from a number of methods.

**Usage**

```
predictionInterval(Y, method = "FAB", alpha = 0.15, mu = 0, tau2 = 1)
```

**Arguments**

Y	Observed data vector
method	Choice of prediction method. Options include FAB, DTA, direct, Bayes.
alpha	Prediction error rate
mu	Prior expected mean of the population mean
tau2	Prior expected variance of the population mean

**Value**

pred object containing prediction interval bounds and interval coverage

**Examples**

```
# example data
data(radon)
y_county9 = radon$radon[radon$group==9]

fab.region = predictionInterval(y_county9,
  method = "FAB",
  alpha = .15,
  mu = 0.5, tau2 = 1)
fab.region$bounds
```

```
plot(fab.region)
```

---

predictionSet	<i>Wrapper to obtain a prediction set for categorical data</i>
---------------	--

---

### Description

This function computes a prediction set from a number of methods.

### Usage

```
predictionSet(  
  Y,  
  method = "FAB",  
  alpha = 0.15,  
  gamma = rep(1, length(Y)),  
  category_names = 1:length(Y)  
)
```

### Arguments

Y	Observed data vector
method	Choice of prediction method. Options include FAB, direct, Bayes.
alpha	Prediction mis-coverage rate
gamma	Dirichlet prior concentration for FAB/Bayes methods
category_names	Category names (optional)

### Value

pred object containing prediction set and interval coverage

### Examples

```
# obtain example categorical data  
set.seed(1)  
prob = rdirichlet(50:1)  
y = rmultinom(1,15,prob)  
  
fab.set = predictionSet(y,  
  method = "FAB",  
  gamma = c(50:1))  
plot(fab.set)
```

---

radon	<i>Minnesota Radon Data</i>
-------	-----------------------------

---

**Description**

Data from a national US EPA survey of household radon values. County index contained in group column.

**Usage**

```
data(radon)
```

**Format**

A matrix.

**Source**

*ARM Data*

**References**

US Environmental Protection Agency (1992) National residential radon survey: summary report. Washington, DC; DOI EPA402-R-92-011.

---

rdirichlet	<i>Generate a random sample from a Dirichlet distribution</i>
------------	---

---

**Description**

Generate a random sample from a Dirichlet distribution

**Usage**

```
rdirichlet(gamma)
```

**Arguments**

gamma                      Prior concentration vector of length K

**Value**

a vector of length K that is a random sample from a Dirichlet distribution

---

row_standardize	<i>Row standardize a matrix</i>
-----------------	---------------------------------

---

**Description**

Row standardize a matrix

**Usage**

```
row_standardize(W)
```

**Arguments**

W                    matrix

**Value**

row-standardized matrix

---

W	<i>Minnesota County Adjacency Matrix</i>
---	--

---

**Description**

Adjacency matrix for MN counties based on group index that matches radon data.

**Usage**

```
data(W)
```

**Format**

A matrix.

# Index

- \* **datasets**
  - radon, [12](#)
  - W, [13](#)
- \* **package**
  - fabPrediction, [5](#)
- \* **pred methods**
  - fabPrediction, [5](#)
- \* **pred plots**
  - plot.pred, [7](#)
- \* **prediction**
  - fabPrediction, [5](#)

bayesMultinomialPrediction, [2](#)  
bayesNormalPrediction, [3](#)

dtaPrediction, [3](#)

eye, [4](#)

fabCategoricalPrediction, [4](#)  
fabContinuousPrediction, [5](#)  
fabPrediction, [5](#)  
fayHerriotEB, [6](#)

initMoM, [6](#)

normalPrediction, [7](#)

plot.pred, [7](#)  
pluginValues, [8](#)  
polyaGradient, [8](#)  
polyaHessian, [9](#)  
polyaMLE, [9](#)  
predictionInterval, [10](#)  
predictionSet, [11](#)

radon, [12](#)  
rdirichlet, [12](#)  
row\_standardize, [13](#)

W, [13](#)