Package 'XOMultinom'

June 21, 2026

Type Package

Title Exact Distributions of Some Functions of the Ordered Multinomial Counts

Version 0.9.0

Date 2026-06-22

Maintainer Sergio Venturini <sergio.venturini@unicatt.it>

Description Implements exact algorithms for computing the distributions of the maximum, the minimum, the range, and the sum of the J largest order statistics of a multinomial random vector. Two complementary algorithm families are provided: the recursive tree-traversal method of Bonetti, Cirillo, and Ogay (2019) <doi:10.1098/rsos.190198>, which covers all four statistics under the equiprobable hypothesis; and the stochastic matrix method of Corrado (2011) <doi:10.1007/s11222-010-9174-3>, which handles the maximum, minimum, and range for arbitrary probability vectors. Functions for power evaluation and sample size determination for goodness-of-fit tests based on these order statistics are also provided. Computationally intensive routines are implemented in 'C++' for efficiency.

License GPL-3

URL https://github.com/sergioventurini/XOMultinom

BugReports https://github.com/sergioventurini/XOMultinom/issues

Depends R (>= 3.6.0), utils

Imports ggplot2 (>= 3.5.1), graphics, methods, Rcpp, rlang, stats, tools

LinkingTo Rcpp, RcppProgress

Config/testthat/edition 3

Encoding UTF-8

LazyData true

RoxygenNote 8.0.0

NeedsCompilation yes

Author Sergio Venturini [aut, cre],
Marco Bonetti [ctb]

Repository CRAN

Date/Publication 2026-06-21 13:40:06 UTC

Contents

`XOMultinom-package` *XOMultinom: Exact distributions of ordered multinomial counts*

Description

The **XOMultinom** package provides functions for computing exact distributions of selected functions of ordered multinomial counts, including the maximum, minimum, range, and sums of order statistics.

Main functions include:

- `dmaxmultinom()`, `pmaxmultinom()`, `qmaxmultinom()`, `rmaxmultinom()`
- `dminmultinom()`, `pminmultinom()`, `qminmultinom()`, `rminmultinom()`
- `drangemultinom()`, `prangemultinom()`, `qrangemultinom()`, `rrangemultinom()`
- `dJlargemultinom()`, `pJlargemultinom()`, `qJlargemultinom()`, `rJlargemultinom()`

Author(s)

Maintainer: Sergio Venturini `<sergio.venturini@unicatt.it>`

Authors:

- Sergio Venturini `<sergio.venturini@unicatt.it>`

Other contributors:

- Marco Bonetti `<marco.bonetti@unibocconi.it>` [contributor]

See Also

Useful links:

- https://github.com/sergioventurini/XOMultinom
- Report bugs at https://github.com/sergioventurini/XOMultinom/issues

as.data.frame.xomultinom_dist

Coerce an `xomultinom_dist` *object to a data frame*

Description

Converts the evaluation points and probability values stored in an `xomultinom_dist` object into a tidy `data.frame` suitable for further manipulation or export.

Usage

```
## S3 method for class 'xomultinom_dist'
as.data.frame(x, ...)
```

Arguments

`x`	An object of class `xomultinom_dist`.
`...`	Further arguments passed to or from other methods (currently unused).

Value

A `data.frame` with columns `x` (evaluation points) and either `pmf` or `cdf` (probability values). If the object was computed on the log scale the column is named `log_pmf` or `log_cdf` accordingly.

Examples

```
k <- 5; n <- 40
obj <- maxmultinomcdf(size = n, prob = rep(1/k, k))
head(as.data.frame(obj))
```

as.data.frame.xomultinom_size

Coerce an `xomultinom_size` *object to a data frame*

Description

Converts the sample size results stored in an `xomultinom_size` object into a single tidy `data.frame` with columns for m, the probability perturbation, and the required sample size.

Usage

```
## S3 method for class 'xomultinom_size'
as.data.frame(x, ...)
```

Arguments

x	An object of class xomultinom_size.
...	Further arguments passed to or from other methods (currently unused).

Value

A data.frame with columns m (integer number of categories), change (probability perturbation), and n_required (required sample size).

Examples

```
sz <- maxmin_multinom_size(
  m_seq = c(5, 10), change_seq = c(0.10, 0.15, 0.20),
  power = 0.80, alpha = 0.05, type = "max"
)
as.data.frame(sz)
```

---

autoplot.xomultinom_dist

ggplot2-based plot for xomultinom_dist *objects*

Description

Produces a ggplot2 plot of the exact distribution stored in an xomultinom_dist object. PMFs are displayed as lollipop (spike) charts; CDFs are displayed as step functions. An optional normal approximation overlay can be added for diagnostic comparison.

Usage

```
## S3 method for class 'xomultinom_dist'
autoplot(
  object,
  add_approx = FALSE,
  colour = "#2166ac",
  approx_colour = "#d6604d",
  title = NULL,
  ...
)
```

Arguments

object	An object of class xomultinom_dist.
add_approx	Logical; if TRUE, overlays the normal approximation to the distribution (mean and variance computed from the exact PMF). Defaults to FALSE.

colour	Character string; colour used for the exact distribution. Defaults to "#2166ac" (blue).
approx_colour	Character string; colour used for the approximation overlay when add_approx = TRUE. Defaults to "#d6604d" (red).
title	Character string; plot title. If NULL (default), a descriptive title is generated automatically.
...	Further arguments passed to or from other methods (currently unused).

Details

For multi-panel layouts use patchwork or gridExtra to combine multiple autoplot() outputs. For base R par(mfrow = ...) compatibility use plot.xomultinom_dist instead.

Value

Invisibly returns the ggplot object.

See Also

plot.xomultinom_dist for a base R alternative compatible with par(mfrow = ...).

Examples

```
k <- 5; n <- 40
obj <- maxmultinomcdf(size = n, prob = rep(1/k, k))
autoplot(obj)
autoplot(obj, add_approx = TRUE)
```

---

autoplot.xomultinom_size

ggplot2-based plot for xomultinom_size *objects*

Description

Produces a ggplot2 line chart of the required sample size as a function of the probability perturbation, with one line per value of m (number of categories).

Usage

```
## S3 method for class 'xomultinom_size'
autoplot(object, log_scale = FALSE, title = NULL, ...)
```

Arguments

`object`	An object of class `xomultinom_size`.
`log_scale`	Logical; if `TRUE`, the y-axis is displayed on a $\log_{10}$ scale. Defaults to `FALSE`.
`title`	Character string; plot title. If `NULL` (default), a descriptive title is generated automatically.
`...`	Further arguments passed to or from other methods (currently unused).

Details

For multi-panel layouts use `patchwork` or `gridExtra` to combine multiple `autoplot()` outputs. For base R `par(mfrow = ...)` compatibility use `plot.xomultinom_size` instead.

Value

Invisibly returns the `ggplot` object.

See Also

`plot.xomultinom_size` for a base R alternative compatible with `par(mfrow = ...)`.

Examples

```
sz_max <- maxmin_multinom_size(
  m_seq = c(5, 10, 20), change_seq = seq(0.10, 0.30, by = 0.05),
  power = 0.80, alpha = 0.05, type = "max"
)
autoplot(sz_max)
autoplot(sz_max, log_scale = TRUE)
```

`dJlargemultinom` *PMF of the sum of the J largest multinomial order statistics at specified points*

Description

Computes $P(S_J = x)$, where $S_J = \sum_{j=1}^{J} N_{\langle j \rangle}$, at each element of `x` for a multinomial random vector with `size` trials and equal cell probabilities `prob`. Returns a plain numeric vector, following the same conventions as `dbinom` and `dnorm`.

Usage

```
dJlargemultinom(x, size, prob, J = 2, log.p = FALSE, verbose = TRUE)
```

Arguments

x	Integer vector of values at which to evaluate the PMF.
size	Integer number of trials n.
prob	Numeric vector of non-negative *equal* cell probabilities (only the equiprobable case is implemented). Values are internally normalised to sum to 1.
J	Integer; number of largest order statistics to sum. Defaults to 2.
log.p	Logical; if TRUE, log-probabilities are returned. Defaults to FALSE.
verbose	Logical; if TRUE, displays progress information during the computation. Defaults to TRUE.

Details

Only the equiprobable case (prob proportional to a constant vector) is currently supported.

For the full distribution object (suitable for plotting, summaries, or repeated evaluation), use Jlargemultinomcdf directly.

Value

A numeric vector of the same length as x, containing $P(S_J = x)$ (or log-probabilities if log.p = TRUE). Points outside the support $\{0, \ldots, n\}$ return 0 (or -Inf on the log scale).

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

See Also

Jlargemultinomcdf for the full distribution object, pJlargemultinom for the CDF at specific points, dmaxmultinom for the PMF of the maximum, and dminmultinom for the PMF of the minimum.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)
J <- 3

# Evaluate at specific points -- plain numeric output, like dbinom()
dJlargemultinom(x = c(30, 35, 40), size = n, prob = probs, J = J)

# Log scale
dJlargemultinom(x = c(30, 35, 40), size = n, prob = probs, J = J,
                log.p = TRUE)

# For the full distribution object use Jlargemultinomcdf():
```

```
FJ <- Jlargemultinomcdf(size = n, prob = probs, J = J)
plot(FJ)
```

`dmaxmultinom` *PMF of the multinomial maximum at specified points*

Description

Computes $P(\max(N_1, \ldots, N_m) = x)$ at each element of `x` for a multinomial random vector with `size` trials and cell probabilities `prob`. Returns a plain numeric vector, following the same conventions as `dbinom` and `dnorm`.

Usage

```
dmaxmultinom(x, size, prob, log.p = FALSE, verbose = TRUE)
```

Arguments

`x`	Integer vector of values at which to evaluate the PMF.
`size`	Integer number of trials n.
`prob`	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
`log.p`	Logical; if `TRUE`, log-probabilities are returned. Defaults to `FALSE`.
`verbose`	Logical; if `TRUE`, displays progress information during the computation. Defaults to `TRUE`.

Details

The function first checks whether `prob` corresponds to the equiprobable case and then applies either the Bonetti et al.\ (2019) algorithm or the Corrado (2011) algorithm accordingly.

For the full distribution object (suitable for plotting, summaries, or repeated evaluation), use `maxmultinomcdf` directly.

Value

A numeric vector of the same length as `x`, containing $P(\max(N_1, \ldots, N_m) = x)$ (or log-probabilities if `log.p = TRUE`). Points outside the support $\{0, \ldots, n\}$ return 0 (or `-Inf` on the log scale).

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

maxmultinomcdf for the full distribution object, pmaxmultinom for the CDF at specific points, dminmultinom for the PMF of the minimum, and drangemultinom for the PMF of the range.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)

# Evaluate at specific points -- plain numeric output, like dbinom()
dmaxmultinom(x = c(18, 20, 22), size = n, prob = probs)

# Log scale
dmaxmultinom(x = c(18, 20, 22), size = n, prob = probs, log.p = TRUE)

# For the full distribution object use maxmultinomcdf():
Fmax <- maxmultinomcdf(size = n, prob = probs)
plot(Fmax)
```

---

dminmultinom *PMF of the multinomial minimum at specified points*

Description

Computes $P(\min(N_1, \ldots, N_m) = x)$ at each element of x for a multinomial random vector with size trials and cell probabilities prob. Returns a plain numeric vector, following the same conventions as dbinom and dnorm.

Usage

```
dminmultinom(x, size, prob, log.p = FALSE, verbose = TRUE)
```

Arguments

x	Integer vector of values at which to evaluate the PMF.
size	Integer number of trials n.
prob	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
log.p	Logical; if TRUE, log-probabilities are returned. Defaults to FALSE.
verbose	Logical; if TRUE, displays progress information during the computation. Defaults to TRUE.

Details

The function first checks whether `prob` corresponds to the equiprobable case and then applies either the Bonetti et al.\ (2019) algorithm or the Corrado (2011) algorithm accordingly.

For the full distribution object (suitable for plotting, summaries, or repeated evaluation), use `minmultinomcdf` directly.

Value

A numeric vector of the same length as `x`, containing $P(\min(N_1, \ldots, N_m) = x)$ (or log-probabilities if `log.p = TRUE`). Points outside the support $\{0, \ldots, n\}$ return 0 (or `-Inf` on the log scale).

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

`minmultinomcdf` for the full distribution object, `pminmultinom` for the CDF at specific points, `dmaxmultinom` for the PMF of the maximum, and `drangemultinom` for the PMF of the range.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)

# Evaluate at specific points -- plain numeric output, like dbinom()
dminmultinom(x = c(10, 12, 15), size = n, prob = probs)

# Log scale
dminmultinom(x = c(10, 12, 15), size = n, prob = probs, log.p = TRUE)

# For the full distribution object use minmultinomcdf():
Fmin <- minmultinomcdf(size = n, prob = probs)
plot(Fmin)
```

`drangemultinom` *PMF of the multinomial range at specified points*

Description

Computes $P(R = x)$, where $R = \max(N_1, \ldots, N_m) - \min(N_1, \ldots, N_m)$, at each element of `x` for a multinomial random vector with `size` trials and cell probabilities `prob`. Returns a plain numeric vector, following the same conventions as `dbinom` and `dnorm`.

Usage

```
drangemultinom(x, size, prob, log.p = FALSE, verbose = TRUE)
```

Arguments

`x`	Integer vector of values at which to evaluate the PMF.
`size`	Integer number of trials n.
`prob`	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
`log.p`	Logical; if `TRUE`, log-probabilities are returned. Defaults to `FALSE`.
`verbose`	Logical; if `TRUE`, displays progress information during the computation. Defaults to `TRUE`.

Details

The function first checks whether `prob` corresponds to the equiprobable case and then applies either the Bonetti et al.\ (2019) algorithm or the Corrado (2011) algorithm accordingly.

For the full distribution object (suitable for plotting, summaries, or repeated evaluation), use `rangemultinomcdf` directly.

Value

A numeric vector of the same length as `x`, containing $P(R = x)$ (or log-probabilities if `log.p = TRUE`). Points outside the support $\{0, \ldots, n\}$ return 0 (or `-Inf` on the log scale).

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

rangemultinomcdf for the full distribution object, prangemultinom for the CDF at specific points, dmaxmultinom for the PMF of the maximum, and dminmultinom for the PMF of the minimum.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)

# Evaluate at specific points -- plain numeric output, like dbinom()
drangemultinom(x = c(5, 10, 15), size = n, prob = probs)

# Log scale
drangemultinom(x = c(5, 10, 15), size = n, prob = probs, log.p = TRUE)

# For the full distribution object use rangemultinomcdf():
Frange <- rangemultinomcdf(size = n, prob = probs)
plot(Frange)
```

---

find_gamma_prob *Randomization probability for max/min multinomial tests*

Description

Computes the randomization probability γ associated with a critical value k_α for tests based on the maximum or minimum of a multinomial random vector.

Usage

```
find_gamma_prob(probs, n, alpha = 0.05, k_alpha, type)
```

Arguments

probs	Numeric vector of probabilities. Must correspond to the equiprobable case.
n	Integer number of trials.
alpha	Significance level in (0, 1).
k_alpha	Integer critical value.
type	Character string; either "max" or "min".

Value

Numeric value representing the randomization probability. Returns NA if not defined.

find_k_alpha *Critical value for max/min multinomial tests*

Description

Computes the critical value k_α for hypothesis tests based on the maximum or minimum of a multinomial random vector.

Usage

```
find_k_alpha(probs, n, alpha = 0.05, type)
```

Arguments

probs	Numeric vector of probabilities. Must correspond to the equiprobable case.
n	Integer number of trials.
alpha	Significance level in (0, 1).
type	Character string; either "max" or "min".

Value

Integer critical value k_α. Returns NA if no valid rejection region exists.

find_k_gamma *Critical value and randomization probability for max/min tests*

Description

Computes the critical value k_α and the corresponding randomization probability γ for hypothesis tests based on the maximum or minimum of a multinomial random vector under the null hypothesis of equiprobable categories.

Usage

```
find_k_gamma(probs, n, alpha = 0.05, type)
```

Arguments

probs	Numeric vector of probabilities. Must correspond to the equiprobable case (i.e., all equal).
n	Integer number of trials.
alpha	Significance level in (0, 1).
type	Character string; either "max" or "min" indicating the test statistic.

Details

The function determines the rejection region for tests based on the maximum or minimum cell count. When the test is not exact, a randomized decision rule is constructed via γ.

Value

A list with components:

k_alpha	Critical value.
gamma_prob	Randomization probability.

Jlargemultinomcdf *Distribution object for the sum of the J largest multinomial order statistics*

Description

Constructs an xomultinom_dist object containing the exact PMF and CDF of $S_J = \sum_{j=1}^{J} N_{\langle j \rangle}$, the sum of the J largest order statistics of a multinomial random vector, evaluated over its full support $\{0, 1, \ldots, n\}$. The returned object can be passed to plot(), autoplot(), summary(), and as.data.frame(), and its CDF and PMF values can be extracted with pJlargemultinom() and dJlargemultinom().

Usage

```
Jlargemultinomcdf(size, prob, J = 2, verbose = TRUE)
```

Arguments

size	Integer number of trials n.
prob	Numeric vector of non-negative *equal* cell probabilities (only the equiprobable case is implemented). Values are internally normalised to sum to 1.
J	Integer; number of largest order statistics to sum. Defaults to 2.
verbose	Logical; if TRUE, displays progress information during the computation. Defaults to TRUE.

Details

Jlargemultinomcdf() is the *distribution constructor*: it fixes size, prob, and J, performs the exact computation once over the full support, and returns a self-contained xomultinom_dist object. The companion functions pJlargemultinom and dJlargemultinom are lightweight wrappers that call Jlargemultinomcdf() internally and extract the CDF or PMF values at the requested points x, returning a plain numeric vector in the same style as pnorm and dnorm.

Only the equiprobable case (prob proportional to a constant vector) is currently supported.

Value

An object of class `xomultinom_dist` with components `x` (full integer support $0, \ldots, n$), `values` (CDF values), `stat = "J_largest"`, `type = "cdf"`, `size`, `prob`, and `log = FALSE`.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

See Also

`pJlargemultinom` for the CDF at specific points (numeric output), `dJlargemultinom` for the PMF at specific points, `maxmultinomcdf`, `minmultinomcdf`, and `rangemultinomcdf` for the analogous constructors.

Examples

```
m <- 4; n <- 60; J <- 3
probs <- rep(1 / m, m)

# Distribution constructor: compute once, reuse freely
FJ <- Jlargemultinomcdf(size = n, prob = probs, J = J)
plot(FJ)
summary(FJ)

# Standard p*/d* interface: plain numeric output
pJlargemultinom(x = c(30, 35, 40), size = n, prob = probs, J = J)
dJlargemultinom(x = c(30, 35, 40), size = n, prob = probs, J = J)
```

---

leukaemia *Data: Leukaemia cases*

Description

This is a well-known epidemiological dataset of diagnosed leukaemia cases over eight counties in upstate New York. These data originated from the New York State Cancer Registry, and were gathered during the 5-year period 1978-1982, with a total of 584 individuals diagnosed with leukaemia over a population of approximately 1 million people. The original data contain spatial information about registered events split into 790 census tracts.

Usage

```
data(leukaemia)
```

Format

A data frame with 790 observations and the following 5 variables:

- `ID` (`int`): 10 character long identification number for a cell or census district in the study area
- `x` (`num`): x-coordinate of the geographic centroid of each cell
- `y` (`num`): y-coordinate of the geographic centroid of each cell
- `pop` (`int`): 1980 U.S. Census population count for each cell
- `cases` (`num`): incident cases of leukemia (all types) occurring between 1978 and 1982 in each cell; fractional values can occur due to partially missing data

Source

The data set has been downloaded from https://www.stats.ox.ac.uk/pub/datasets/csb/.

References

Lange, N., Ryan, L., Billard, L., Brillinger, D., Conquest, L., Greenhouse, J. (1994), "Case Studies in Biometry", Hoboken, NJ: Wiley & Sons.

---

mainsail *MAINSAIL trial: comparator-arm data with Halabi 2014 risk scores*

Description

Baseline characteristics and Halabi (2014) prognostic linear predictor for the 520 patients randomised to the comparator arm (docetaxel plus prednisone) of the MAINSAIL trial (NCT00988208), a phase III study in metastatic castration-resistant prostate cancer (mCRPC). The dataset is used in `XOMultinom` to illustrate the sequential recalibration-alarm procedure described in Section~5.2 of the package paper.

Usage

```
mainsail
```

Format

A data frame with 520 rows and 21 variables:

RPT Character. Zero-padded patient identifier (e.g.\ `"00468"`).

ENROLLDAY Numeric. Randomisation day on the study-day scale (day 0 = study start). Ranges from -265 to 353; used to derive `entry_order`.

entry_order Integer. Patient's rank by ascending `ENROLLDAY`, from 1 (earliest randomised) to 520 (latest). Ties in `ENROLLDAY` are broken arbitrarily.

ecog Numeric. Eastern Cooperative Oncology Group (ECOG) performance status at baseline: 0 (fully active), 1 (restricted in strenuous activity), or 2 (ambulatory, capable of self-care only).

disease_site Character. Halabi (2014) disease-site classification: `"ln_only"` (lymph-node involvement only, $n = 89$) or `"visceral"` (any liver or lung metastasis, $n = 350$). `NA` for 81 patients for whom disease site could not be determined from the available tumour-assessment records; all such patients have `has_bone = 0`.

has_ln Integer. Binary indicator: 1 if lymph-node metastases were recorded at the screening visit, 0 otherwise.

has_bone Integer. Binary indicator: 1 if bone metastases were recorded at the screening visit, 0 otherwise.

has_visceral Integer. Binary indicator: 1 if visceral (liver or lung) metastases were recorded at the screening visit, 0 otherwise.

opioid Integer. Binary indicator: 1 if the patient was receiving opioid analgesics (ATC code `N02A*`) at the time of randomisation, 0 otherwise.

ldh Numeric. Lactate dehydrogenase (LDH) at baseline, in U/L. Missing for 8 patients.

ldh_uln Numeric. Upper limit of normal for LDH as recorded in the trial laboratory data. Constant at 250 U/L for all patients in this dataset.

ldh_gt_uln Integer. Binary indicator: 1 if `ldh > ldh_uln`, 0 otherwise. Complete for all 520 patients (missing `ldh` values were treated as not exceeding the ULN).

albumin Numeric. Serum albumin at baseline, in g/dL. Missing for 5 patients.

hgb Numeric. Haemoglobin at baseline, in g/dL. Missing for 16 patients.

psa Numeric. Prostate-specific antigen (PSA) at baseline, in ng/mL. Missing for 6 patients.

alp Numeric. Alkaline phosphatase (ALP) at baseline, in U/L. Missing for 8 patients.

ln_psa Numeric. Natural logarithm of `psa`. Missing for the same 6 patients as `psa`.

ln_alp Numeric. Natural logarithm of `alp`. Missing for the same 8 patients as `alp`.

halabi2014_lp Numeric. Halabi (2014) linear predictor computed by strict listwise deletion: `NA` for any patient missing at least one of the ten model covariates (99 patients). Identical to `halabi2014_lp_raw` for the 421 complete cases.

halabi2014_lp_raw Numeric. Halabi (2014) linear predictor computed under partial listwise deletion: available for the 498 patients with complete laboratory values, regardless of `disease_site` availability. For the 77 patients with missing `disease_site` but complete labs, both disease-site indicators are set to zero (equivalent to assigning the lymph-node-only reference category). `NA` for the 22 patients missing at least one laboratory value.

halabi2014_lp_imputed Numeric. Halabi (2014) linear predictor after single imputation: complete for all 520 patients. Continuous covariates (`albumin`, `hgb`, `ln_psa`, `ln_alp`) are imputed at their sample median; `disease_site` is imputed at its sample mode (`"visceral"`). Used as the risk score in the sequential recalibration-alarm illustration of Section~5.2.

Details

The MAINSAIL trial randomised 1059 patients with chemotherapy-naive mCRPC to docetaxel/prednisone with or without lenalidomide. Only the 520 patients on the comparator arm are included here. Patient entry order was determined by `ENROLLDAY` extracted from `assignmt.sas7bdat` in the Project Data Sphere release; all other covariates were extracted at or closest to the baseline visit. Full details of variable construction are given in Appendix~A of the package paper.

The Halabi (2014) linear predictor is defined as

$$\eta_i = \boldsymbol{\beta}^\top \mathbf{x}_i,$$

where the regression coefficients $\boldsymbol{\beta}$ are the log-hazard ratios from Table~2 of Halabi et al.\ (2014); see `vignette("recalibration", package = "XOMultinom")` for the full specification.

Source

Project Data Sphere, dataset identifier `Prostat\_Celgene\_2009\_90` (https://data.projectdatasphere.org/). Access requires registration and acceptance of the Project Data Sphere terms of use.

References

Halabi, S., Lin, C.-Y., Kelly, W.K., Fizazi, K.S., Moul, J.W., Kaplan, E.B., Morris, M.J. and Small, E.J. (2014). Updated prognostic model for predicting overall survival in first-line chemotherapy for patients with metastatic castration-resistant prostate cancer. *Journal of Clinical Oncology*, **32**(7), 671–677. doi:10.1200/JCO.2013.52.3696

Fizazi, K., Higano, C.S., Nelson, J.B., et al.\ (2013). Phase III, randomized, placebo-controlled study of docetaxel in combination with zibotentan in patients with metastatic castration-resistant prostate cancer. *Journal of Clinical Oncology*, **31**(14), 1740–1747. doi:10.1200/JCO.2012.46.4149

`make_breaks` *Create Quantile-Based Break Points*

Description

Computes `m` quantile-based intervals from a numeric vector of scores and replaces the outer boundaries with `-Inf` and `Inf` so that all possible values are included in the resulting intervals.

Usage

```
make_breaks(scores, m)
```

Arguments

`scores`	A numeric vector of scores from which quantile break points are computed.
`m`	An integer specifying the number of intervals (e.g., `m = 10` for deciles).

Details

Quantiles are computed using `stats::quantile()` with `type = 1`.

Value

A numeric vector of length `m + 1` containing the break points. The first and last elements are `-Inf` and `Inf`, respectively.

`maxmin_multinom_size` *Sample size determination for multinomial max/min tests*

Description

Computes the required sample size to achieve a target power for hypothesis tests based on the maximum or minimum of a multinomial random vector under deviations from equiprobability.

Usage

```
maxmin_multinom_size(
  m_seq,
  change_seq,
  power = 0.8,
  alpha = 0.05,
  n_max = 500,
  type,
  verbose = TRUE,
  optmethod = "uniroot",
  extendInt = "upX"
)
```

Arguments

`m_seq`	Integer vector of numbers of categories.
`change_seq`	Numeric vector of probability perturbations from the equiprobable case.
`power`	Desired power level in (0, 1).
`alpha`	Significance level in (0, 1).
`n_max`	Maximum sample size considered in the search.
`type`	Character string; either `"max"` or `"min"`.
`verbose`	Logical; if `TRUE`, progress messages are printed.
`optmethod`	Character string; optimization method, either `"uniroot"` or `"optimize"`.
`extendInt`	Passed to `uniroot()` when used.

Details

The function evaluates the sample size needed to detect deviations from equiprobability with a given power, using tests based on either the maximum or minimum multinomial cell count.

Value

A list where each element corresponds to a value of `m_seq` and contains the required sample sizes for each value in `change_seq`.

Examples

```
pow <- 0.8
alpha <- 0.05
m_seq <- 3:8
incr_seq <- seq(0.2, 0.8, 0.1)
res <- maxmin_multinom_size(m_seq, incr_seq, power = pow, alpha = alpha,
                            n_max = 200, type = "max",
                            verbose = TRUE, optmethod = "uniroot")
summary(res)
plot(res)
```

`maxmultinomcdf` *Distribution object for the multinomial maximum count*

Description

Constructs an `xomultinom_dist` object containing the exact CDF of the maximum cell count $\max(N_1, \ldots, N_m)$ of a multinomial random vector, evaluated over its full support $\{0, 1, \ldots, n\}$. The returned object can be passed to `plot()`, `autoplot()`, `summary()`, and `as.data.frame()`, and its CDF and PMF values can be extracted with `pmaxmultinom()` and `dmaxmultinom()`.

Usage

```
maxmultinomcdf(size, prob, verbose = TRUE)
```

Arguments

`size`	Integer number of trials n.
`prob`	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
`verbose`	Logical; if `TRUE`, displays progress information during the computation. Defaults to `TRUE`.

Details

`maxmultinomcdf()` is the *distribution constructor*: it fixes `size` and `prob`, performs the (potentially expensive) exact computation once over the full support, and returns a self-contained `xomultinom_dist` object. The companion functions `pmaxmultinom` and `dmaxmultinom` provide the CDF or PMF values at the requested points `x`, returning a plain numeric vector in the same style as `pnorm` and `dnorm`.

Use `maxmultinomcdf()` when you need the full distribution object (e.g., for plotting or for evaluating the CDF at many points without repeating the underlying computation). Use `pmaxmultinom` or `dmaxmultinom` when you need a numeric vector at specific quantiles, in the same way you would use `pnorm()` or `dnorm()`.

The function dispatches automatically to the Bonetti et al. (2019) recursive algorithm (equiprobable case) or the Corrado (2011) matrix algorithm (general case).

Value

An object of class `xomultinom_dist` with components `x` (full integer support $0, \ldots, n$), `values` (CDF values), `stat = "max"`, `type = "cdf"`, `size`, `prob`, and `log = FALSE`.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

`pmaxmultinom` for the CDF at specific points (numeric output), `dmaxmultinom` for the PMF at specific points (numeric output), `minmultinomcdf` and `rangemultinomcdf` for the analogous constructors for the minimum and the range.

Examples

```
m <- 4; n <- 60
probs <- rep(1 / m, m)

# Distribution constructor: compute once, reuse freely
Fmax <- maxmultinomcdf(size = n, prob = probs)
plot(Fmax)
summary(Fmax)

# Standard p*/d* interface: plain numeric output
pmaxmultinom(x = c(18, 20, 22), size = n, prob = probs)
dmaxmultinom(x = c(18, 20, 22), size = n, prob = probs)
```

---

max_count *Compute the Largest Bin Count*

Description

Assigns a sample of scores to intervals defined by a set of break points and returns the size of the largest resulting bin.

Usage

```
max_count(brks, samp_scores, m)
```

Arguments

brks	A numeric vector of break points defining the intervals.
samp_scores	A numeric vector of sample scores to be assigned to bins.
m	An integer specifying the expected number of bins.

Details

The function uses cut() to classify observations into bins and tabulate() to count the number of observations in each bin.

Value

A single integer giving the maximum number of observations contained in any bin.

---

minmultinomcdf *Distribution object for the multinomial minimum count*

Description

Constructs an xomultinom_dist object containing the exact PMF and CDF of the minimum cell count $\min(N_1, \ldots, N_m)$ of a multinomial random vector, evaluated over its full support $\{0, 1, \ldots, n\}$. The returned object can be passed to plot(), autoplot(), summary(), and as.data.frame(), and its CDF and PMF values can be extracted with pmaxmultinom() and dmaxmultinom().

Usage

```
minmultinomcdf(size, prob, verbose = TRUE)
```

Arguments

size	Integer number of trials n.
prob	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
verbose	Logical; if TRUE, displays progress information during the computation. Defaults to TRUE.

Details

maxmultinomcdf() is the *distribution constructor*: it fixes size and prob, performs the (potentially expensive) exact computation once over the full support, and returns a self-contained xomultinom_dist object. The companion functions pminmultinom and dminmultinom provide the CDF or PMF values at the requested points x, returning a plain numeric vector in the same style as pnorm and dnorm.

Use minmultinomcdf() when you need the full distribution object (e.g., for plotting or for evaluating the CDF at many points without repeating the underlying computation). Use pminmultinom or

dminmultinom when you need a numeric vector at specific quantiles, in the same way you would use pnorm() or dnorm().

The function dispatches automatically to the Bonetti et al. (2019) recursive algorithm (equiprobable case) or the Corrado (2011) matrix algorithm (general case).

Value

An object of class xomultinom_dist with components x (full integer support $0, \ldots, n$), values (CDF values), stat = "max", type = "cdf", size, prob, and log = FALSE.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

pminmultinom for the CDF at specific points (numeric output), dminmultinom for the PMF at specific points (numeric output), maxmultinomcdf and rangemultinomcdf for the analogous constructors for the maximum and the range.

Examples

```
m <- 4; n <- 60
probs <- rep(1 / m, m)

# Distribution constructor: compute once, reuse freely
Fmin <- minmultinomcdf(size = n, prob = probs)
plot(Fmin)
summary(Fmin)

# Standard p*/d* interface: plain numeric output
pminmultinom(x = c(18, 20, 22), size = n, prob = probs)
dminmultinom(x = c(18, 20, 22), size = n, prob = probs)
```

---

pJlargemultinom — *CDF of the sum of J largest order statistics for a multinomial distribution evaluated at specified points*

Description

Computes the cumulative distribution function of the sum of J largest order statistics, $S_J = \sum_{j=1}^{J} N_{\langle j \rangle}$, for a multinomial random vector with equal cell probabilities.

Usage

```
pJlargemultinom(
  x,
  size,
  prob,
  J = 2,
  lower.tail = TRUE,
  log.p = FALSE,
  verbose = TRUE
)
```

Arguments

x	Numeric vector of values at which to evaluate the CDF.
size	Integer number of trials n.
prob	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
J	Integer; number of largest order statistics to sum. Defaults to 2.
lower.tail	Logical; if TRUE (default), $P(S_J \leq x)$ is returned; otherwise $P(S_J > x)$.
log.p	Logical; if TRUE, probabilities are returned on the log scale. Defaults to FALSE.
verbose	Logical; if TRUE, displays progress information during the computation. Defaults to TRUE.

Details

The function only implements the equiprobable case.

Value

A numeric vector of the same length as x, containing $P(S_J \leq x)$ (or its complement or log, according to lower.tail and log.p). Values outside the support are handled consistently with base R: x < 0 gives 0 and x > n gives 1 (before lower.tail/log.p transformations).

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

See Also

Jlargemultinomcdf for the full distribution object, dJlargemultinom for the PMF, qJlargemultinom for quantiles, and rJlargemultinom for random generation.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)
J <- 3
xseq <- 0:n

cdflarge <- pJlargemultinom(x = xseq, size = n, prob = probs, J = J)
cdflarge
```

`plot.xomultinom_dist` *Plot method for* `xomultinom_dist` *objects*

Description

Produces a base R plot of the exact distribution stored in an `xomultinom_dist` object, compatible with `par(mfrow = ...)`, `layout()`, and all other base R multi-panel layout mechanisms. PMFs are displayed as spike (needle) charts; CDFs are displayed as step functions. An optional normal approximation overlay can be added for diagnostic comparison.

Usage

```
## S3 method for class 'xomultinom_dist'
plot(
  x,
  add_approx = FALSE,
  col = "#2166ac",
  approx_col = "#d6604d",
  main = NULL,
  xlab = "x",
  ylab = NULL,
  ...
)
```

Arguments

`x`	An object of class `xomultinom_dist`.
`add_approx`	Logical; if `TRUE`, overlays the normal approximation to the distribution (mean and variance computed from the exact PMF). Defaults to `FALSE`.
`col`	Character string; colour used for the exact distribution. Defaults to `"#2166ac"` (blue).
`approx_col`	Character string; colour used for the approximation overlay when `add_approx = TRUE`. Defaults to `"#d6604d"` (red).
`main`	Character string; plot title. If `NULL` (default), a descriptive title is generated automatically.

xlab	Character string; x-axis label. Defaults to "x".
ylab	Character string; y-axis label. If NULL (default), an appropriate label is generated automatically.
...	Further graphical parameters passed to the underlying base R plotting functions.

Value

Invisibly returns NULL.

See Also

autoplot.xomultinom_dist for a ggplot2-based alternative.

Examples

```
k <- 5; n <- 40
obj_cdf <- maxmultinomcdf(size = n, prob = rep(1/k, k))

plot(obj_cdf)
```

---

plot.xomultinom_size *Plot method for* xomultinom_size *objects*

Description

Produces a base R line chart of the required sample size as a function of the probability perturbation, with one line per value of m (number of categories), compatible with par(mfrow = ...), layout(), and all other base R multi-panel layout mechanisms.

Usage

```
## S3 method for class 'xomultinom_size'
plot(
  x,
  log_scale = FALSE,
  col = NULL,
  main = NULL,
  xlab = NULL,
  ylab = "Required n",
  ...
)
```

Arguments

x	An object of class xomultinom_size.
log_scale	Logical; if TRUE, the y-axis (required n) is displayed on a $\log_{10}$ scale. Useful when n varies over several orders of magnitude. Defaults to FALSE.
col	Character vector of colours, one per value of m_seq. If NULL (default), colours are taken from the default R palette.
main	Character string; plot title. If NULL (default), a descriptive title is generated automatically.
xlab	Character string; x-axis label. If NULL (default), an appropriate label is generated automatically.
ylab	Character string; y-axis label. Defaults to "Required n".
...	Further graphical parameters passed to the underlying base R plotting functions.

Value

Invisibly returns NULL.

See Also

autoplot.xomultinom_size for a ggplot2-based alternative.

Examples

```
sz <- maxmin_multinom_size(
  m_seq = c(5, 10, 20), change_seq = seq(0.10, 0.30, by = 0.05),
  power = 0.80, alpha = 0.05, type = "max"
)

# Compatible with par(mfrow = ...)
op <- par(mfrow = c(1, 2))
plot(sz)
plot(sz, log_scale = TRUE)
par(op)
```

---

pmaxmultinom *CDF of the multinomial maximum at specified points*

Description

Computes the cumulative distribution function of the maximum cell count of a multinomial random vector with arbitrary cell probabilities.

Usage

```
pmaxmultinom(x, size, prob, lower.tail = TRUE, log.p = FALSE, verbose = TRUE)
```

Arguments

x	Numeric vector of values at which to evaluate the CDF.
size	Integer number of trials n.
prob	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
lower.tail	Logical; if TRUE (default), $P(\max(N_1, \ldots, N_m) \leq x)$ is returned; otherwise $P(\max(N_1, \ldots, N_m) > x)$.
log.p	Logical; if TRUE, probabilities are returned on the log scale. Defaults to FALSE.
verbose	Logical; if TRUE, displays progress information during the computation. Defaults to TRUE.

Details

The function first checks whether prob corresponds to the equiprobable case and then applies either the Bonetti et al. (2019) algorithm or the Corrado (2011) algorithm accordingly.

Value

A numeric vector of the same length as x, containing $P(\max(N_1, \ldots, N_m) \leq x)$ (or its complement or log, according to lower.tail and log.p). Values outside the support are handled consistently with base R: x < 0 gives 0 and x > n gives 1 (before lower.tail/log.p transformations).

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

maxmultinomcdf for the full distribution object, pminmultinom for the CDF of the minimum, dmaxmultinom for the PMF of the maximum, and dminmultinom for the PMF of the minimum.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)
xseq <- 0:n

cdfmax <- pmaxmultinom(x = xseq, size = n, prob = probs)
cdfmax
```

`pminmultinom` *CDF of the multinomial minimum at specified points*

Description

Computes the cumulative distribution function of the minimum cell count of a multinomial random vector with arbitrary cell probabilities.

Usage

```
pminmultinom(x, size, prob, lower.tail = TRUE, log.p = FALSE, verbose = TRUE)
```

Arguments

`x`	Numeric vector of values at which to evaluate the CDF.
`size`	Integer number of trials n.
`prob`	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
`lower.tail`	Logical; if `TRUE` (default), $P(\min(N_1, \ldots, N_m) \leq x)$ is returned; otherwise $P(\min(N_1, \ldots, N_m) > x)$.
`log.p`	Logical; if `TRUE`, probabilities are returned on the log scale. Defaults to `FALSE`.
`verbose`	Logical; if `TRUE`, displays progress information during the computation. Defaults to `TRUE`.

Details

The function first checks whether `prob` corresponds to the equiprobable case and then applies either the Bonetti et al. (2019) algorithm or the Corrado (2011) algorithm accordingly.

Value

A numeric vector of the same length as `x`, containing $P(\min(N_1, \ldots, N_m) \leq x)$ (or its complement or log, according to `lower.tail` and `log.p`). Values outside the support are handled consistently with base R: `x < 0` gives 0 and `x > n` gives 1 (before `lower.tail`/`log.p` transformations).

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

minmultinomcdf for the full distribution object, pmaxmultinom for the CDF of the maximum, dminmultinom for the PMF of the minimum, and drangemultinom for the PMF of the range.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)
xseq <- 0:n

cdfmin <- pminmultinom(x = xseq, size = n, prob = probs)
cdfmin
```

---

prangemultinom *CDF of the multinomial range at specified points*

Description

Computes the cumulative distribution function of the range $R = \max(N_1, \ldots, N_m) - \min(N_1, \ldots, N_m)$ for a multinomial random vector with arbitrary cell probabilities.

Usage

```
prangemultinom(x, size, prob, lower.tail = TRUE, log.p = FALSE, verbose = TRUE)
```

Arguments

x	Numeric vector of values at which to evaluate the CDF.
size	Integer number of trials n.
prob	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
lower.tail	Logical; if TRUE (default), $P(R \leq x)$ is returned; otherwise $P(R > x)$.
log.p	Logical; if TRUE, probabilities are returned on the log scale. Defaults to FALSE.
verbose	Logical; if TRUE, displays progress information during the computation. Defaults to TRUE.

Details

The function first checks whether prob corresponds to the equiprobable case and then applies either the Bonetti et al. (2019) algorithm or the Corrado (2011) algorithm accordingly.

Value

A numeric vector of the same length as x, containing $P(R \leq x)$ (or its complement or log, according to `lower.tail` and `log.p`). Values outside the support are handled consistently with base R: `x < 0` gives 0 and `x > n` gives 1 (before `lower.tail`/`log.p` transformations).

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

`prangemultinom` for the CDF at specific points (numeric output), `drangemultinom` for the PMF at specific points (numeric output), `maxmultinomcdf` and `minmultinomcdf` for the analogous constructors for the maximum and the minimum.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)
xseq <- 0:n

cdfrange <- prangemultinom(x = xseq, size = n, prob = probs)
cdfrange
```

---

print.xomultinom_dist *Print method for* xomultinom_dist *objects*

Description

Displays a compact, human-readable table of evaluation points and the corresponding exact probabilities (or log-probabilities) stored in an `xomultinom_dist` object.

Usage

```
## S3 method for class 'xomultinom_dist'
print(x, digits = 4, max_rows = 20, ...)
```

Arguments

x	An object of class xomultinom_dist.
digits	Integer number of significant digits for probabilities. Defaults to 4.
max_rows	Maximum number of rows to display when the support is large. If the number of evaluation points exceeds max_rows, the first and last max_rows / 2 rows are shown with an ellipsis in between. Defaults to 20.
...	Further arguments passed to or from other methods (currently unused).

Value

Invisibly returns x.

Examples

```
k <- 5; n <- 40
obj <- maxmultinomcdf(size = n, prob = rep(1/k, k))
print(obj)
```

---

print.xomultinom_size *Print method for* xomultinom_size *objects*

Description

Displays the required sample sizes as a formatted table, one block per number of categories m.

Usage

```
## S3 method for class 'xomultinom_size'
print(x, digits = 4, ...)
```

Arguments

x	An object of class xomultinom_size.
digits	Integer number of decimal places for probability columns. Defaults to 4.
...	Further arguments passed to or from other methods (currently unused).

Value

Invisibly returns x.

Examples

```
sz <- maxmin_multinom_size(
  m_seq = c(5, 10), change_seq = c(0.10, 0.15, 0.20),
  power = 0.80, alpha = 0.05, type = "max"
)
print(sz)
```

`qJlargemultinom` *Quantile function of the sum of J largest order statistics for a multinomial distribution*

Description

Computes exact quantiles of the distribution of the sum of the J largest order statistics $S_J = \sum_{j=1}^{J} N_{\langle j \rangle}$ of a multinomial random vector with equal cell probabilities, by inverting the exact CDF obtained from `pJlargemultinom`.

Usage

```
qJlargemultinom(p, size, prob, J = 2, lower.tail = TRUE, log.p = FALSE)
```

Arguments

`p`	Numeric vector of probabilities (or log-probabilities if `log.p = TRUE`) at which to evaluate the quantile function.
`size`	Integer number of trials.
`prob`	Numeric vector of non-negative, equal cell probabilities. Only the equiprobable case is supported; a non-equiprobable `prob` will raise an error (propagated from `pJlargemultinom`).
`J`	Integer number of largest order statistics to sum. Defaults to `2`.
`lower.tail`	Logical; if `TRUE` (default), $Q(p) = \min\{x : F(x) \geq p\}$; if `FALSE`, $Q(p) = \min\{x : F(x) \geq 1 - p\}$.
`log.p`	Logical; if `TRUE`, `p` is taken to be on the log scale. Defaults to `FALSE`.

Details

The function obtains the exact CDF over the full support $\{0, 1, \ldots, n\}$ via a single vectorised call to `pJlargemultinom`. The quantile is then located as the smallest support point whose CDF value meets or exceeds `p`. Only the equiprobable case is supported, consistent with `pJlargemultinom`.

Value

Integer vector of the same length as `p` containing the corresponding exact quantiles of S_J.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

See Also

pJlargemultinom for the CDF, dJlargemultinom for the PMF, rJlargemultinom for random generation.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)

# Median and 95th percentile of S_3
qJlargemultinom(c(0.5, 0.95), size = n, prob = probs, J = 3)

# Upper tail
qJlargemultinom(0.05, size = n, prob = probs, J = 3, lower.tail = FALSE)
```

---

qmaxmultinom *Quantile function of the maximum for a multinomial distribution*

Description

Computes exact quantiles of the distribution of the maximum cell count of a multinomial random vector with arbitrary cell probabilities, by inverting the exact CDF obtained from pmaxmultinom.

Usage

```
qmaxmultinom(p, size, prob, lower.tail = TRUE, log.p = FALSE)
```

Arguments

p	Numeric vector of probabilities (or log-probabilities if log.p = TRUE) at which to evaluate the quantile function.
size	Integer number of trials.
prob	Numeric vector of non-negative cell probabilities. Values are internally normalized to sum to 1. Categories with zero probability are removed before computation.
lower.tail	Logical; if TRUE (default), $Q(p) = \min\{x : F(x) \geq p\}$; if FALSE, $Q(p) = \min\{x : F(x) \geq 1 - p\}$.
log.p	Logical; if TRUE, p is taken to be on the log scale. Defaults to FALSE.

Details

The function obtains the exact CDF over the full support $\{0, 1, \ldots, n\}$ via a single vectorised call to pmaxmultinom, which dispatches internally to the Bonetti et al. (2019) algorithm for equiprobable prob and to the Corrado (2011) algorithm otherwise. The quantile is then located as the smallest support point whose CDF value meets or exceeds p, an $O(n)$ lookup requiring no root-finding or approximation.

Value

Integer vector of the same length as p containing the corresponding exact quantiles of the multinomial maximum.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

pmaxmultinom for the CDF, dmaxmultinom for the PMF, rmaxmultinom for random generation.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)

# Median and 95th percentile
qmaxmultinom(c(0.5, 0.95), size = n, prob = probs)

# Upper tail
qmaxmultinom(0.05, size = n, prob = probs, lower.tail = FALSE)
```

---

qminmultinom *Quantile function of the minimum for a multinomial distribution*

Description

Computes exact quantiles of the distribution of the minimum cell count of a multinomial random vector with arbitrary cell probabilities, by inverting the exact CDF obtained from pminmultinom.

Usage

```
qminmultinom(p, size, prob, lower.tail = TRUE, log.p = FALSE)
```

Arguments

p	Numeric vector of probabilities (or log-probabilities if log.p = TRUE) at which to evaluate the quantile function.
size	Integer number of trials.
prob	Numeric vector of non-negative cell probabilities. Values are internally normalized to sum to 1. Categories with zero probability are removed before computation.
lower.tail	Logical; if TRUE (default), $Q(p) = \min\{x : F(x) \geq p\}$; if FALSE, $Q(p) = \min\{x : F(x) \geq 1 - p\}$.
log.p	Logical; if TRUE, p is taken to be on the log scale. Defaults to FALSE.

Details

The function obtains the exact CDF over the full support $\{0, 1, \ldots, n\}$ via a single vectorised call to pminmultinom, which dispatches internally to the Bonetti et al. (2019) algorithm for equiprobable prob and to the Corrado (2011) algorithm otherwise. The quantile is then located as the smallest support point whose CDF value meets or exceeds p, an $O(n)$ lookup requiring no root-finding or approximation.

Value

Integer vector of the same length as p containing the corresponding exact quantiles of the multinomial minimum.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

pminmultinom for the CDF, dminmultinom for the PMF, rminmultinom for random generation.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)

# Median and 95th percentile
```

```
qminmultinom(c(0.5, 0.95), size = n, prob = probs)

# Upper tail
qminmultinom(0.05, size = n, prob = probs, lower.tail = FALSE)
```

`qrangemultinom` *Quantile function of the range for a multinomial distribution*

Description

Computes exact quantiles of the distribution of the range $R = \max(N_1, \ldots, N_m) - \min(N_1, \ldots, N_m)$ of a multinomial random vector with arbitrary cell probabilities, by inverting the exact CDF obtained from `prangemultinom`.

Usage

```
qrangemultinom(p, size, prob, lower.tail = TRUE, log.p = FALSE)
```

Arguments

`p`	Numeric vector of probabilities (or log-probabilities if `log.p = TRUE`) at which to evaluate the quantile function.
`size`	Integer number of trials.
`prob`	Numeric vector of non-negative cell probabilities. Values are internally normalized to sum to 1. Categories with zero probability are removed before computation.
`lower.tail`	Logical; if `TRUE` (default), $Q(p) = \min\{x : F(x) \geq p\}$; if `FALSE`, $Q(p) = \min\{x : F(x) \geq 1 - p\}$.
`log.p`	Logical; if `TRUE`, `p` is taken to be on the log scale. Defaults to `FALSE`.

Details

The function obtains the exact CDF over the full support $\{0, 1, \ldots, n\}$ via a single vectorised call to `prangemultinom`, which dispatches internally to the Bonetti et al. (2019) algorithm for equiprobable `prob` and to the Corrado (2011) algorithm otherwise. The quantile is then located as the smallest support point whose CDF value meets or exceeds `p`, an $O(n)$ lookup requiring no root-finding or approximation.

Value

Integer vector of the same length as `p` containing the corresponding exact quantiles of the multinomial range.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

`prangemultinom` for the CDF, `drangemultinom` for the PMF, `rrangemultinom` for random generation.

Examples

```
m <- 4
n <- 60
probs <- rep(1 / m, m)

# Median and 95th percentile
qrangemultinom(c(0.5, 0.95), size = n, prob = probs)

# Upper tail
qrangemultinom(0.05, size = n, prob = probs, lower.tail = FALSE)
```

---

rand_test *Apply a Randomized Test Decision Rule*

Description

Implements a randomized decision rule based on an observed maximum bin count. The test rejects with probability 1 when the observed count is at least `kappa`, rejects with probability `gamma` when the observed count equals `kappa - 1`, and does not reject otherwise.

Usage

```
rand_test(obs_max, kappa, gamma)
```

Arguments

`obs_max`	An integer giving the observed maximum bin count.
`kappa`	An integer threshold defining the rejection region.
`gamma`	A numeric value in `[0, 1]` giving the rejection probability when `obs_max == kappa - 1`.

Details

Randomization at the boundary is performed using `stats::rbinom()`.

Value

A logical or integer indicator of rejection:

`1L` the test rejects deterministically (`obs_max >= kappa`).

`0L` **or** `1L` a randomized decision when `obs_max == kappa - 1`.

`FALSE` the test does not reject (`obs_max < kappa - 1`).

`rangemultinomcdf` *Distribution object for the multinomial range*

Description

Constructs an `xomultinom_dist` object containing the exact PMF and CDF of the range $R = \max(N_1, \ldots, N_m) - \min(N_1, \ldots, N_m)$ of a multinomial random vector, evaluated over its full support $\{0, 1, \ldots, n\}$. The returned object can be passed to `plot()`, `autoplot()`, `summary()`, and `as.data.frame()`, and its CDF and PMF values can be extracted with `prangemultinom()` and `drangemultinom()`.

Usage

```
rangemultinomcdf(size, prob, verbose = TRUE)
```

Arguments

`size`	Integer number of trials n.
`prob`	Numeric vector of non-negative cell probabilities. Values are internally normalised to sum to 1. Categories with zero probability are removed before computation.
`verbose`	Logical; if `TRUE`, displays progress information during the computation. Defaults to `TRUE`.

Details

`rangemultinomcdf()` is the *distribution constructor*: it fixes `size` and `prob`, performs the exact computation once over the full support, and returns a self-contained `xomultinom_dist` object. The companion functions `prangemultinom` and `drangemultinom` are lightweight wrappers that call `rangemultinomcdf()` internally and extract the CDF or PMF values at the requested points `x`, returning a plain numeric vector in the same style as `pnorm` and `dnorm`.

Use `rangemultinomcdf()` when you need the full distribution object (e.g., for plotting or for evaluating the CDF at many points without repeating the underlying computation). Use `prangemultinom` or `drangemultinom` when you need a numeric vector at specific quantiles.

The function dispatches automatically to the Bonetti et al. (2019) recursive algorithm (equiprobable case) or the Corrado (2011) matrix algorithm (general case).

Value

An object of class `xomultinom_dist` with components `x` (full integer support $0, \ldots, n$), `values` (CDF values), `stat = "range"`, `type = "cdf"`, `size`, `prob`, and `log = FALSE`.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

`prangemultinom` for the CDF at specific points (numeric output), `drangemultinom` for the PMF at specific points (numeric output), `maxmultinomcdf` and `minmultinomcdf` for the analogous constructors for the maximum and the minimum.

Examples

```
m <- 4; n <- 60
probs <- rep(1 / m, m)

# Distribution constructor: compute once, reuse freely
Frange <- rangemultinomcdf(size = n, prob = probs)
plot(Frange)
summary(Frange)

# Standard p*/d* interface: plain numeric output
prangemultinom(x = c(5, 10, 15), size = n, prob = probs)
drangemultinom(x = c(5, 10, 15), size = n, prob = probs)
```

---

rdirichlet *Random generation from a Dirichlet distribution*

Description

Generates random samples from a Dirichlet distribution using gamma variates.

Usage

```
rdirichlet(n, alpha)
```

Arguments

`n`	Integer number of observations to generate.
`alpha`	Numeric vector or matrix of positive concentration parameters.

Details

Each sample is obtained by drawing independent gamma random variables and normalizing them to sum to one. If `alpha` is a vector, it is recycled across rows.

Value

A numeric matrix with `n` rows, where each row is a sample from the Dirichlet distribution and sums to 1.

Examples

```
rdirichlet(5, c(1, 1, 1))
rdirichlet(3, c(2, 5, 3))
```

`rJlargemultinom` *Random generation from the distribution of the sum of J largest order statistics for a multinomial distribution*

Description

Draws independent random samples from the exact distribution of $S_J = \sum_{j=1}^{J} N_{\langle j \rangle}$ for a multinomial random vector with equal cell probabilities.

Usage

```
rJlargemultinom(n, size, prob, J = 2)
```

Arguments

`n`	Integer number of random samples to draw.
`size`	Integer number of trials in each multinomial experiment.
`prob`	Numeric vector of non-negative, equal cell probabilities. Only the equiprobable case is supported; a non-equiprobable `prob` will raise an error (propagated from `dJlargemultinom`).
`J`	Integer number of largest order statistics to sum. Defaults to `2`.

Details

The exact PMF over the support $\{0, 1, \ldots, \texttt{size}\}$ is computed once using `dJlargemultinom`, and `n` independent draws are then obtained via `sample` with those probabilities as weights. Only the equiprobable case is supported, consistent with `dJlargemultinom`.

Value

Integer vector of length `n` containing independent draws from the distribution of S_J.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

See Also

dJlargemultinom for the PMF, pJlargemultinom for the CDF, qJlargemultinom for quantiles.

Examples

```
m <- 4; n <- 60
probs <- rep(1 / m, m)

set.seed(42)
sims <- rJlargemultinom(n = 1000, size = n, prob = probs, J = 3)
hist(sims, breaks = 20, main = "Simulated sums of 3 largest order statistics")
```

---

rmaxmultinom *Random generation from the distribution of the multinomial maximum*

Description

Draws independent random samples from the exact distribution of the maximum cell count of a multinomial random vector with arbitrary cell probabilities.

Usage

```
rmaxmultinom(n, size, prob)
```

Arguments

n	Integer number of random samples to draw.
size	Integer number of trials in each multinomial experiment.
prob	Numeric vector of non-negative cell probabilities. Values are internally normalized to sum to 1. Categories with zero probability are removed before computation.

Details

The exact PMF over the support $\{0, 1, \ldots, \texttt{size}\}$ is computed once using dmaxmultinom, and n independent draws are then obtained via sample with those probabilities as weights. The cost is therefore dominated by the single PMF evaluation and is independent of n.

Value

Integer vector of length n containing independent draws from the distribution of $\max(N_1, \ldots, N_m)$.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

`dmaxmultinom` for the PMF, `pmaxmultinom` for the CDF, `qmaxmultinom` for quantiles.

Examples

```
m <- 4; n <- 60
probs <- rep(1 / m, m)

set.seed(42)
sims <- rmaxmultinom(n = 1000, size = n, prob = probs)
hist(sims, breaks = 20, main = "Simulated multinomial maxima")
```

`rminmultinom` *Random generation from the distribution of the multinomial minimum*

Description

Draws independent random samples from the exact distribution of the minimum cell count of a multinomial random vector with arbitrary cell probabilities.

Usage

```
rminmultinom(n, size, prob)
```

Arguments

`n`	Integer number of random samples to draw.
`size`	Integer number of trials in each multinomial experiment.
`prob`	Numeric vector of non-negative cell probabilities. Values are internally normalized to sum to 1. Categories with zero probability are removed before computation.

Details

The exact PMF over the support $\{0, 1, \ldots, \texttt{size}\}$ is computed once using `dminmultinom`, and `n` independent draws are then obtained via `sample` with those probabilities as weights. The cost is therefore dominated by the single PMF evaluation and is independent of `n`.

Value

Integer vector of length n containing independent draws from the distribution of $\min(N_1, \ldots, N_m)$.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

`dminmultinom` for the PMF, `pminmultinom` for the CDF, `qminmultinom` for quantiles.

Examples

```
m <- 4; n <- 60
probs <- rep(1 / m, m)

set.seed(42)
sims <- rminmultinom(n = 1000, size = n, prob = probs)
hist(sims, breaks = 20, main = "Simulated multinomial minima")
```

`rrangemultinom` *Random generation from the distribution of the multinomial range*

Description

Draws independent random samples from the exact distribution of the range $R = \max(N_1, \ldots, N_m) - \min(N_1, \ldots, N_m)$ of a multinomial random vector with arbitrary cell probabilities.

Usage

```
rrangemultinom(n, size, prob)
```

Arguments

`n`	Integer number of random samples to draw.
`size`	Integer number of trials in each multinomial experiment.
`prob`	Numeric vector of non-negative cell probabilities. Values are internally normalized to sum to 1. Categories with zero probability are removed before computation.

Details

The exact PMF over the support $\{0, 1, \ldots, \texttt{size}\}$ is computed once using drangemultinom, and n independent draws are then obtained via sample with those probabilities as weights. The cost is therefore dominated by the single PMF evaluation and is independent of n.

Value

Integer vector of length n containing independent draws from the distribution of R.

References

Bonetti, M., Cirillo, P., Ogay, A. (2019). Computing the exact distributions of some functions of the ordered multinomial counts: maximum, minimum, range and sums of order statistics. Royal Society Open Science, 6, 190198. doi:10.1098/rsos.190198

Corrado, C.J. (2011). The exact distribution of the maximum, minimum and the range of Multinomial/Dirichlet and Multivariate Hypergeometric frequencies. Statistical Computing, 21, 349–359. doi:10.1007/s1122201091743

See Also

drangemultinom for the PMF, prangemultinom for the CDF, qrangemultinom for quantiles.

Examples

```
m <- 4; n <- 60
probs <- rep(1 / m, m)

set.seed(42)
sims <- rrangemultinom(n = 1000, size = n, prob = probs)
hist(sims, breaks = 20, main = "Simulated multinomial ranges")
```

---

summary.xomultinom_dist

Summary method for xomultinom_dist *objects*

Description

Computes and displays descriptive statistics of the exact distribution stored in an xomultinom_dist object, including the mean, median, mode, standard deviation, effective support, and a central 95\ interval.

Usage

```
## S3 method for class 'xomultinom_dist'
summary(object, digits = 4, ...)
```

Arguments

object	An object of class xomultinom_dist.
digits	Integer number of significant digits. Defaults to 4.
...	Further arguments passed to or from other methods (currently unused).

Value

Invisibly returns a named list with components mean, median, mode, sd, var, support, q025, and q975.

Examples

```
k <- 5; n <- 40
obj <- maxmultinomcdf(size = n, prob = rep(1/k, k))
summary(obj)
```

---

summary.xomultinom_size

Summary method for xomultinom_size *objects*

Description

Prints a condensed overview of the required sample sizes across all combinations of m and probability perturbations, reporting the range of n for each m.

Usage

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

Arguments

object	An object of class xomultinom_size.
...	Further arguments passed to or from other methods (currently unused).

Value

Invisibly returns a named list where each element corresponds to a value of m and contains n_min, n_max, and n_median.

Examples

```
sz <- maxmin_multinom_size(
  m_seq = c(5, 10), change_seq = c(0.10, 0.15, 0.20),
  power = 0.80, alpha = 0.05, type = "max"
)
summary(sz)
```

Index