Package 'AccSamplingDesign'

November 12, 2026

Title Acceptance Sampling Plans Design

Version 0.1.3

Description Provides tools for designing and analyzing Acceptance Sampling plans. Supports both Attributes Sampling (Binomial and Poisson distributions) and Variables Sampling (Normal and Beta distributions), enabling quality control for fractional and compositional data. Uses nonlinear programming for sampling plan optimization, minimizing sample size while controlling producer's and consumer's risks. Operating Characteristic curves are available for plan visualization.

License GPL-3

Encoding UTF-8

Imports stats, methods

Suggests VGAM, knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

URL https://github.com/vietha/AccSamplingDesign, https://doi.org/10.32614/RJ-2026-007

BugReports https://github.com/vietha/AccSamplingDesign/issues

RoxygenNote 7.3.2

NeedsCompilation no

Author Ha Truong [aut, cre, cph], Victor Miranda [ths, rev], Roger Kissling [ths, rev]

Maintainer Ha Truong <truongvietha87@gmail.com>

Repository CRAN

Date/Publication 2026-11-12 10:41:21 UTC

Contents

accProb 2
manualPlan 3
muEst 5
OCdata 6
optAttrPlan 7
optPlan 8
optVarPlan 11
plot.AttrPlan 14
plot.OCdata 15
plot.VarPlan 16
summary.AttrPlan 17
summary.VarPlan 18

Index **20**

`accProb` *Acceptance Probability*

Description

Calculate the probability of acceptance for a given quality level.

Usage

```
accProb(plan, p)
```

Arguments

`plan`	Acceptance plan object (`AttrPlan`/`VarPlan`).
`p`	True quality level (proportion of nonconforming).

Details

The experimental `"saddlepoint_mom"` method uses the joint sample moments, the finite-integer-sample moment support, and the unchanged MOM inspection rule. Its numerical and finite-sample validation is incomplete: ordinary settings can be unsupported. Unsupported integrations or searches raise an error and do not return a substitute probability or plan. Population quadrature orders 128 and 256 and integration refinements must agree within 0.00002. A conservative log-concavity check on the tilted logit-integration kernel also rejects cases with unresolved remote tilted mass; rejected nodes are not omitted and renormalized. This numerical discrepancy is not a statistical risk bound. Integer plan searches check working risks and adjacent sizes; global minimality is not established. The method is slower than the Delta methods and caches successful and failed population calculations. The default remains `"delta_mle"`.

For Beta plans with unknown theta, the calculation follows the `method` stored in the plan: analytical Delta–MLE, analytical Delta–MoM, the Govindaraju–Kissling sample-size adjustment, or experimental Saddlepoint MLE OC. The latter numerically integrates a normalized joint MLE density and

returns an approximate probability. It is more expensive than Delta evaluation, caches the distribution for repeated constants, and raises an error if integration does not converge. Integration error and finite-sample approximation error are distinct. The endpoints `p=0` and `p=1` use their limiting probabilities.

Value

Numeric probability between 0 and 1.

Author(s)

Ha Truong

Examples

```
# Example for attribute plan
attr_plan <- optAttrPlan(PRQ = 0.01, CRQ = 0.1)
accProb(attr_plan, 0.05)

# Example for variable plan (normal distribution)
var_plan <- optVarPlan(
  PRQ = 0.025,        # Acceptable quality level (% nonconforming)
  CRQ = 0.1,         # Rejectable quality level (% nonconforming)
  alpha = 0.05,      # Producer's risk
  beta = 0.1,        # Consumer's risk
  distribution = "normal"
)
accProb(var_plan, 0.05)
```

---

manualPlan *Create manual acceptance sampling (AS) plan from User Inputs*

Description

Constructs an `AttrPlan` or `VarPlan` object from the given parameters.

Usage

```
manualPlan(distribution = c("binomial", "poisson", "normal", "beta"),
           n = NULL, c = NULL, k = NULL,
           USL = NULL, LSL = NULL, sigma = NULL, theta = NULL,
           sigma_type = c("known", "unknown"),
           theta_type = c("known", "unknown"),
           method = c("delta_mle", "delta_mom", "gk_adjustment",
                      "saddlepoint_mle", "saddlepoint_mom"))
```

Arguments

distribution	One of "binomial", "poisson", "normal", or "beta".
n	Sample size.
c	Acceptance number (for attribute sampling).
k	Acceptability constant (for variable sampling).
USL	Upper specification limit.
LSL	Lower specification limit.
sigma	Standard deviation (for normal plans).
theta	Precision parameter (for beta plans).
sigma_type	Either "known" or "unknown" (for normal).
theta_type	Either "known" or "unknown" (for beta).
method	Method used by an unknown-theta Beta plan. Delta–MLE is the default; alternatives are "delta_mom", "gk_adjustment", and experimental "saddlepoint_mle" or "saddlepoint_mom".

Details

The experimental "saddlepoint_mom" method uses the joint sample moments, the finite-integer-sample moment support, and the unchanged MOM inspection rule. Its numerical and finite-sample validation is incomplete: ordinary settings can be unsupported. Unsupported integrations or searches raise an error and do not return a substitute probability or plan. Population quadrature orders 128 and 256 and integration refinements must agree within 0.00002. A conservative log-concavity check on the tilted logit-integration kernel also rejects cases with unresolved remote tilted mass; rejected nodes are not omitted and renormalized. This numerical discrepancy is not a statistical risk bound. Integer plan searches check working risks and adjacent sizes; global minimality is not established. The method is slower than the Delta methods and caches successful and failed population calculations. The default remains "delta_mle".

This function provides a user-friendly wrapper to construct AS plan directly from parameters. Internally, it constructs the appropriate AttrPlan or VarPlan, from given paramenters. Delta–MLE requires finite n >= 2 and finite nonnegative k. Saddlepoint MLE additionally requires an integer inspection size. Supply explicit quality points pd when using OCdata() for a manual saddlepoint plan.

Value

An object of class "AttrPlan" or VarPlan.

Author(s)

Ha Truong

See Also

optPlan, OCdata

Examples

```
# Attribute sampling with user-defined parameters
plan1 <- manualPlan(n = 100, c = 2, distribution = "binomial")

# Variable sampling (normal)
plan2 <- manualPlan(n = 30, k = 1.5, distribution = "normal", USL = 10, sigma = 1)

# Unknown-theta Beta plan using analytical Delta--MoM
plan3 <- manualPlan(n = 45, k = 2.2, distribution = "beta",
                    theta = 300, theta_type = "unknown", method = "delta_mom",
                    USL = 0.05)
```

---

muEst *Estimate Mean μ Based on Specification Limits and Probability*

Description

Computes the estimated mean μ for a given level of quality and specification limit under either a normal or beta distribution.

Usage

```
muEst(p, USL = NULL, LSL = NULL,
       sigma = NULL, theta = NULL,
       dist = c("normal", "beta"))
```

Arguments

p	Level of quality (numeric, between 0 and 1).
USL	Upper specification limit (numeric). Only one of USL or LSL should be provided.
LSL	Lower specification limit (numeric). Only one of USL or LSL should be provided.
sigma	Standard deviation (numeric) for the normal distribution. Must be provided if dist = "normal".
theta	Theta parameter (numeric) for the beta distribution. Must be provided if dist = "beta".
dist	Distribution type. Either "normal" or "beta".

Details

The function estimates the mean μ corresponding to a given tail probability p, assuming that the process output follows either a normal or beta distribution, and that the probability of being beyond the provided specification limit equals 1 - p.

- For the normal distribution, the mean is calculated using the inverse cumulative distribution function (quantile function) of the normal distribution.

- For the beta distribution, the mean is solved numerically such that the CDF at the specified limit equals p, given the shape determined by `theta`.

Exactly one of `USL` or `LSL` must be provided to define whether the probability refers to the upper or lower tail.

Value

Returns the estimated mean μ as a numeric value.

Author(s)

Ha Truong

Examples

```
# Example for normal distribution with lower specification limit (LSL)
muEst(p = 0.95, LSL = 10, sigma = 2, dist = "normal")

# Example for beta distribution with upper specification limit (USL)
muEst(p = 0.95, USL = 0.7, theta = 500, dist = "beta")
```

`OCdata` *Generic function for OC Curve Generation*

Description

Generic function to compute Operating Characteristic (OC) curve data from an acceptance sampling plan.

Usage

```
OCdata(plan, pd = NULL)
```

Arguments

`plan`	An object of class `AttrPlan` or `VarPlan`.
`pd`	Vector of quality levels (proportions of nonconforming items).

Details

This is a generic function. Methods are defined for objects of class `AttrPlan` and `VarPlan`, which compute the probability of acceptance across a range of quality levels (proportions of nonconforming).

See `OCdata.AttrPlan` and `OCdata.VarPlan` for details.

Value

An object of class `"OCdata"`, a list containing:

- `pd` — quality levels (e.g. proportion defective)
- `paccept` — probability of acceptance at each level
- `process_means` — process means, if applicable
- `dist`, `n`, `k`, `c` — plan parameters

Author(s)

Ha Truong

See Also

`optPlan`, `manualPlan`

`optAttrPlan` *Attributes Acceptance Sampling Plan*

Description

Designs Attributes Acceptance Sampling plans using producer/consumer risk criteria.

Usage

```
optAttrPlan(PRQ, CRQ, alpha = 0.05, beta = 0.10,
            distribution = c("binomial", "poisson"))
```

Arguments

`PRQ`	Producer Risk Quality (0 < PRQ < 1)
`CRQ`	Consumer Risk Quality (PRQ < CRQ < 1)
`alpha`	Producer's risk (0.05 default)
`beta`	Consumer's risk (0.10 default)
`distribution`	Support binomial and poisson distribution

Value

AttrPlan object containing:

`n`	Sample size
`c`	Acceptance number
`PRQ`	Input PRQ value
`CRQ`	Input CRQ value
`distribution`	Selected distribution

Author(s)

Ha Truong

References

ISO 2859-1:1999 - Sampling procedures for inspection by attributes

Schilling, E.G., & Neubauer, D.V. (2017). Acceptance Sampling in Quality Control (3rd ed.). Chapman and Hall/CRC. https://doi.org/10.4324/9781315120744

Examples

```
plan <- optAttrPlan(PRQ = 0.01, CRQ = 0.1, alpha = 0.05, beta = 0.1,
                    distribution = "binomial")
```

`optPlan` *Optimal Acceptance Sampling Plan*

Description

Design optimal variable acceptance sampling plans based on specified parameters. Supports different distributions (binomial, poisson, normal, beta) and accommodates known or unknown standard deviation and process parameters.

Usage

```
optPlan(PRQ, CRQ, alpha = 0.05, beta = 0.10, USL = NULL, LSL = NULL,
        distribution = c("binomial", "poisson", "normal", "beta"),
        sigma_type = c("known", "unknown"),
        theta_type = c("known", "unknown"),
        sigma = NULL, theta = NULL,
        method = c("delta_mle", "delta_mom", "gk_adjustment",
                   "saddlepoint_mle", "saddlepoint_mom"))
```

Arguments

`PRQ`	Producer's risk quality level (e.g., acceptable quality level).
`CRQ`	Consumer's risk quality level (e.g., rejectable quality level).
`alpha`	Producer's risk (Type I error), default is 0.05.
`beta`	Consumer's risk (Type II error), default is 0.10.
`USL`	Upper Specification Limit. Required for variable sampling plans.
`LSL`	Lower Specification Limit. Required for variable sampling plans.
`distribution`	Distribution type used in the plan. Can be `"binomial"`, `"poisson"`, `"normal"`, or `"beta"`.
`sigma_type`	Indicates if the standard deviation (`sigma`) is known or unknown.

theta_type	Indicates if the process parameter (theta) is known or unknown.
sigma	Known standard deviation of the process, if applicable.
theta	Precision (dispersion) parameter for the beta distribution. If unknown, it can be estimated from historical data using functions such as betaff.
method	Method for a Beta plan with theta_type = "unknown": "delta_mle" (default), "delta_mom", or "gk_adjustment", or experimental "saddlepoint_mle" and "saddlepoint_mom". It must not be supplied in other cases.

Details

The experimental "saddlepoint_mom" method uses the joint sample moments, the finite-integer-sample moment support, and the unchanged MOM inspection rule. Its numerical and finite-sample validation is incomplete: ordinary settings can be unsupported. Unsupported integrations or searches raise an error and do not return a substitute probability or plan. Population quadrature orders 128 and 256 and integration refinements must agree within 0.00002. A conservative log-concavity check on the tilted logit-integration kernel also rejects cases with unresolved remote tilted mass; rejected nodes are not omitted and renormalized. This numerical discrepancy is not a statistical risk bound. Integer plan searches check working risks and adjacent sizes; global minimality is not established. The method is slower than the Delta methods and caches successful and failed population calculations. The default remains "delta_mle".

The experimental "saddlepoint_mle" method integrates a normalized joint MLE saddlepoint density. It uses deterministic numerical integration, not simulation, and retains the MLE inspection rule. Its initial cost is larger than the Delta methods. Returned risks are numerical approximations at the delivered integer size, not guarantees of finite-sample risk control. Nonconvergent integration raises an error; no alternative OC is substituted. See optVarPlan for details.

This function designs optimal acceptance sampling plans by balancing producer's and consumer's risks under specified quality levels. It supports plans for attributes (binomial) and variables (normal or beta distributions), including cases with unknown standard deviation or distributional parameters.

For the "normal" model, the plan can be computed assuming the standard deviation (sigma) is either known or estimated from data. Optimization for unknown sigma is performed using the derivative-free Nelder-Mead method (Nelder and Mead, 1965) via optim in base R.

For the "beta" model, the dispersion (precision) parameter theta determines how concentrated the distribution is around the mean. Users must either:

- provide a known value for theta, typically obtained from prior process studies or historical data; or
- estimate it from sample data using a fitting function such as betaff in the **VGAM** package.

Optimization for Beta plans is performed using optim with the "L-BFGS-B" method (Byrd et al., 1995) to handle bounds on sample size and acceptance numbers.

For unknown theta, the Delta methods approximate the joint uncertainty of the estimated Beta mean and precision. Delta–MLE uses inverse expected Fisher information; Delta–MoM propagates the first four Beta raw moments. The "gk_adjustment" option preserves the earlier Govindaraju–Kissling/package-style sample-size adjustment.

This approach ensures stable and efficient plan calculation while relying solely on base R functionality.

Value

Returns a list or data frame with optimal sample size(s) and critical value(s) based on the specified parameters and distribution.

`distribution`	Selected distribution
`sample_size`	Final sample size after rounding up to the next integer, for practical application.
`n`	Sample size (unrounded for Variables Acceptance Sampling).
`k`	Acceptability constant - For Variables Acceptance Sampling.
`c`	Acceptance number - For Attributes Acceptance Sampling.

Author(s)

Ha Truong

References

* H. Truong, V. Miranda and R. Kissling (2026). AccSamplingDesign: An R Package for Optimizing Acceptance Sampling Plans. *The R Journal*, 18(1), 368–381. doi:10.32614/RJ2026007.

* ISO 2859-1:1999 - Sampling procedures for inspection by attributes

* ISO 3951-1:2013 - Sampling procedures for inspection by variables.

* Wilrich, PT. (2004). Single Sampling Plans for Inspection by Variables under a Variance Component Situation. In: Lenz, HJ., Wilrich, PT. (eds) Frontiers in Statistical Quality Control 7. Physica, Heidelberg.

* K. Govindaraju and R. Kissling (2015). Sampling plans for Beta-distributed compositional fractions.

* J. A. Nelder and R. Mead. A simplex method for function minimization. The Computer Journal, 7(4): 308–313, 1965. DOI 10.1093/comjnl/7.4.308.

* R. H. Byrd, P. Lu, J. Nocedal and C. Zhu. A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing, 16(5): 1190–1208, 1995. DOI 10.1137/0916069.

Examples

```
# Example usage (normal distribution, known sigma):
optPlan(PRQ = 0.005, CRQ = 0.03, alpha = 0.05, beta = 0.10,
        distribution = "normal", sigma_type = "known")

# Example usage (beta distribution, unknown theta):
optPlan(PRQ = 0.025, CRQ = 0.10, alpha = 0.05, beta = 0.10,
        distribution = "beta", theta = 6.6e8,
        theta_type = "unknown", method = "delta_mle", LSL = 5.65e-6)
```

optVarPlan *Variables Acceptance Sampling Plan*

Description

Creates Variable Acceptance Sampling plans for normal or beta distributed measurements.

Usage

```
optVarPlan(PRQ, CRQ, alpha = 0.05, beta = 0.10, USL = NULL, LSL = NULL,
         distribution = c("normal", "beta"), sigma_type = c("known", "unknown"),
           theta_type = c("known", "unknown"), sigma = NULL, theta = NULL,
           method = c("delta_mle", "delta_mom", "gk_adjustment",
                      "saddlepoint_mle", "saddlepoint_mom"))
```

Arguments

PRQ	Producer Risk Quality (must be within valid range for the chosen distribution).
CRQ	Consumer Risk Quality (must be greater than PRQ and within valid range).
alpha	Producer's risk (numeric between 0 and 1).
beta	Consumer's risk (numeric between 0 and 1).
USL	Upper Specification Limit (numeric). Only one of USL or LSL should be provided.
LSL	Lower Specification Limit (numeric). Only one of USL or LSL should be provided.
distribution	Measurement distribution: "normal" or "beta".
sigma_type	Indicates whether sigma (population standard deviation) is "known" or "unknown".
theta_type	Indicates whether theta (population precision parameter for beta) is "known" or "unknown".
sigma	Known standard deviation (used for normal distribution). Required if sigma_type = "known".
theta	Precision (dispersion) parameter for the beta distribution. If unknown, it can be estimated from historical data using functions such as betaff.
method	Unknown-theta Beta method: "delta_mle" (default), "delta_mom", "gk_adjustment", or experimental "saddlepoint_mle" and "saddlepoint_mom". It is only applicable when distribution = "beta" and theta_type = "unknown".

Details

The experimental "saddlepoint_mom" method uses the joint sample moments, the finite-integer-sample moment support, and the unchanged MOM inspection rule. Its numerical and finite-sample validation is incomplete: ordinary settings can be unsupported. Unsupported integrations or searches raise an error and do not return a substitute probability or plan. Population quadrature orders 128 and 256 and integration refinements must agree within 0.00002. A conservative log-concavity check

on the tilted logit-integration kernel also rejects cases with unresolved remote tilted mass; rejected nodes are not omitted and renormalized. This numerical discrepancy is not a statistical risk bound. Integer plan searches check working risks and adjacent sizes; global minimality is not established. The method is slower than the Delta methods and caches successful and failed population calculations. The default remains `"delta_mle"`.

The function generates variable acceptance sampling plans based on specified producer and consumer risks and either a normal or beta distribution model.

The specification limit must be defined via either `USL` (upper specification limit) or `LSL` (lower specification limit), depending on whether the one-sided quality criterion concerns the upper or lower tail. Only one limit should be provided.

For the `"normal"` model, the plan can be computed assuming the standard deviation (`sigma`) is either known or estimated from data. Optimization for unknown `sigma` is performed using the derivative-free Nelder-Mead method (Nelder and Mead, 1965) via `optim` in base R.

For the `"beta"` model, the dispersion (precision) parameter `theta` determines how concentrated the distribution is around the mean. Users must either:

- provide a known value for `theta`, typically obtained from prior process studies or historical data; or
- estimate it from sample data using a fitting function such as `betaff` in the **VGAM** package.

Optimization for Beta plans is performed using `optim` with the `"L-BFGS-B"` method (Byrd et al., 1995) to handle bounds on sample size and acceptance numbers, followed by a Nelder-Mead refinement that solves the two binding risk equations. For known-theta plans, the sample size is then rounded up from the binding solution and both risks are verified at the delivered (integer) size, so the plan meets `PR <= alpha` and `CR <= beta` as applied.

For unknown precision, Delta–MLE uses inverse expected Fisher information and Delta–MoM uses analytical covariance propagated from the first four Beta raw moments. The Govindaraju–Kissling adjustment retains the package's earlier sample-size approximation.

The experimental `"saddlepoint_mle"` method constructs a positive joint approximation to the fitted Beta mean and precision, including the parameter transformation Jacobian, then numerically normalizes and integrates it over the existing acceptance region. No simulation is used in its OC or optimization. The distribution is reused across acceptance constants. It is slower than Delta evaluation. Unsupported or nonconvergent integrations raise errors.

Saddlepoint plans use an integer size, choose the midpoint of the feasible acceptance-constant interval, and verify both numerical risk constraints. Adjacent smaller integer feasibility is checked; global sample-size minimality is not guaranteed. The numerical diagnostics do not bound the statistical approximation error, and finite-sample risk control is not guaranteed.

This approach ensures stable and efficient plan calculation while relying solely on base R functionality.

Value

A `VarPlan` object containing:

`distribution`	Distribution used ("normal" or "beta").
`sample_size`	Final sample size after rounding up to the next integer, for practical application.

k	Acceptability constant.
n	The sample size the plan is evaluated at. For known-theta Beta plans and "saddlepoint_mle" this is the delivered integer size (equal to sample_size); for other plans it is the unrounded value from the optimization.
integration	For saddlepoint plans, experimental status, estimated numerical convergence error and quadrature order at the two design points.
n_continuous	For saddlepoint plans, the continuous size used to locate the integer search neighbourhood.

Author(s)

Ha Truong

References

P. McCullagh. *Tensor Methods in Statistics*, Chapter 6. https://www.stat.uchicago.edu/~pmcc/tensorbook/edition2.pdf

* H. Truong, V. Miranda and R. Kissling (2026). AccSamplingDesign: An R Package for Optimizing Acceptance Sampling Plans. *The R Journal*, 18(1), 368–381. doi:10.32614/RJ2026007.

* ISO 3951-1:2013 - Sampling procedures for inspection by variables.

* Wilrich, PT. (2004). Single Sampling Plans for Inspection by Variables under a Variance Component Situation. In: Lenz, HJ., Wilrich, PT. (eds) Frontiers in Statistical Quality Control 7. Physica, Heidelberg.

* K. Govindaraju and R. Kissling (2015). Sampling plans for Beta-distributed compositional fractions.

* J. A. Nelder and R. Mead. A simplex method for function minimization. The Computer Journal, 7(4): 308–313, 1965. DOI 10.1093/comjnl/7.4.308.

* R. H. Byrd, P. Lu, J. Nocedal and C. Zhu. A limited memory algorithm for bound constrained optimization. SIAM Journal on Scientific Computing, 16(5): 1190–1208, 1995. DOI 10.1137/0916069.

Examples

```
# Example for normal distribution plan
norm_plan <- optVarPlan(
  PRQ = 0.025,        # Acceptable quality level (% nonconforming)
  CRQ = 0.1,          # Rejectable quality level (% nonconforming)
  alpha = 0.05,       # Producer's risk
  beta = 0.1,         # Consumer's risk
  distribution = "normal",
  USL = 10
)
summary(norm_plan)

# Example for beta distribution plan
beta_plan <- optVarPlan(
  PRQ = 0.025,        # Target quality level (% nonconforming)
  CRQ = 0.1,          # Minimum quality level (% nonconforming)
  alpha = 0.05,       # Producer's risk
```

```
  beta = 0.1,         # Consumer's risk
  distribution = "beta",
  theta = 44000000,   # Beta distribution parameter
  LSL = 0.00001
)
summary(beta_plan)
```

`plot.AttrPlan` *Plot the OC Curve for Attribute Sampling Plans*

Description

Plots the Operating Characteristic (OC) curve for an attribute sampling plan object of class `AttrPlan`.

Usage

```
## S3 method for class 'AttrPlan'
plot(x, pd = NULL, ...)
```

Arguments

`x`	An object of class `AttrPlan` representing an attribute acceptance sampling plan.
`pd`	Optional vector of proportions of nonconforming items. If `NULL` (default), a range is automatically generated.
`...`	Additional graphical parameters passed to `plot()`.

Details

This method computes and visualizes the probability of acceptance (`P(accept)`) as a function of the proportion of nonconforming items in the population, based on the attribute sampling plan.

The plot also includes reference lines at the plan's producer and consumer quality levels (`PRQ`, `CRQ`) and their corresponding acceptance probabilities.

Value

A plot showing the OC curve for the given attribute sampling plan.

Author(s)

Ha Truong

See Also

`optAttrPlan`, `accProb`, `OCdata`

Examples

```
# Create attribute plan
plan <- optAttrPlan(PRQ = 0.01, CRQ = 0.1)

# Plot OC curve
plot(plan)

# With custom pd
plot(plan, pd = seq(0, 0.15, by = 0.001))
```

`plot.OCdata` *Plot Method for OCdata Objects*

Description

Plots the Operating Characteristic (OC) curve from an object of class `"OCdata"`, either by proportion nonconforming or process mean levels.

Usage

```
## S3 method for class 'OCdata'
plot(x, by = c("pd", "mean"), ...)
```

Arguments

`x`	An object of class `"OCdata"`, typically generated using `OCdata()`.
`by`	A character string indicating the type of OC curve to plot. Options are:
	`"pd"` (Default) Plot the OC curve by proportion nonconforming.
	`"mean"` Plot the OC curve by estimated process mean levels (only available for variable sampling plans).
`...`	Additional graphical parameters passed to the `plot()` function.

Details

This method visualizes the OC curve based on the content of the `"OCdata"` object.

By default, the curve is plotted against the proportion of nonconforming items (`@pd`). If `by = "mean"` is specified and the plan includes valid mean-level estimates (`@process_means`), the curve is plotted against mean levels.

If `by = "mean"` is requested but no mean estimates are available (e.g., for attribute plans), a message will be shown and no plot will be drawn.

Value

A plot showing the OC curve for the given attribute/variable sampling plan.

Author(s)

Ha Truong

See Also

OCdata, optAttrPlan, optVarPlan

Examples

```
# Attribute plan
plan_attr <- optAttrPlan(PRQ = 0.01, CRQ = 0.05)
oc_attr <- OCdata(plan_attr)
plot(oc_attr)               # OC curve by pd (default)
plot(oc_attr, by = "mean")  # Will show message if not available

# Variable plan
plan_var <- optVarPlan(PRQ = 0.025, CRQ = 0.1, USL = 0.1,
                       distribution = "normal", sigma=0.01)
oc_var <- OCdata(plan_var)
plot(oc_var)                # OC curve by pd
plot(oc_var, by = "mean")   # OC curve by mean levels
```

---

plot.VarPlan *Plot the OC Curve for Variable Sampling Plans*

Description

Plots the Operating Characteristic (OC) curve for an object of class `VarPlan`. Supports plotting against either the proportion of nonconforming items or the corresponding process mean levels, depending on availability.

Usage

```
## S3 method for class 'VarPlan'
plot(x, pd = NULL, by = c("pd", "mean"), ...)
```

Arguments

`x`	An object of class `VarPlan` representing a variable acceptance sampling plan.
`pd`	Optional numeric vector of proportions of nonconforming items to evaluate. If `NULL` (default), a suitable range is generated automatically.
`by`	Character string indicating which x-axis to use for plotting. Either `"pd"` for proportion nonconforming (default) or `"mean"` for process mean levels. If `"mean"` is selected but the plan lacks specification limits, an error is raised.
`...`	Additional graphical parameters passed to `plot()`.

Details

This plotting method visualizes the probability of acceptance (`P(accept)`) against the desired metric, based on the parameters of a variable sampling plan.

If `by = "pd"`, the x-axis represents the proportion of nonconforming items. If `by = "mean"` and the plan defines `limit_type` and `spec_limit`, the function estimates corresponding process means using `muEst` and plots the OC curve by those mean values.

Reference lines for the Producer's Risk Quality (PRQ) and Consumer's Risk Quality (CRQ), along with their respective acceptance probabilities, are shown when plotting by proportion.

Value

A plot showing the OC curve for the given variable sampling plan, either by nonconforming proportion or mean level.

Author(s)

Ha Truong

See Also

`optVarPlan`, `accProb`, `muEst`, `OCdata`, `plot.OCdata`

Examples

```
# Variable sampling plan with specification limits
plan <- optVarPlan(
  PRQ = 0.025, CRQ = 0.1,
  alpha = 0.05, beta = 0.1,
  distribution = "normal",
  USL = 3, sigma = 0.1
)

# Plot by proportion nonconforming
plot(plan, by = "pd")

# Plot by estimated mean level (requires spec_limit and limit_type)
plot(plan, by = "mean")

# Custom pd vector
plot(plan, pd = seq(0.01, 0.15, by = 0.001))
```

---

summary.AttrPlan *Summarize Attribute Acceptance Plan*

Description

Detailed summaries for attribute acceptance plans.

Usage

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

Arguments

object	Plan object to summarize
...	Additional parameters (ignored)

Value

No return value. This function is called for its side effect of printing a formatted summary of the attribute sampling plan to the console.

Author(s)

Ha Truong

Examples

```
attr_plan <- optAttrPlan(PRQ = 0.01, CRQ = 0.1)
summary(attr_plan)
```

---

summary.VarPlan *Summarize Variables Acceptance Plan*

Description

Detailed summaries for Variables Acceptance Sampling plans.

Usage

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

Arguments

object	Plan object to summarize
...	Additional parameters (ignored)

Value

No return value. This function is called for its side effect of printing a formatted summary of the variable sampling plan to the console.

Author(s)

Ha Truong

Examples

```
var_plan <- optVarPlan(
  PRQ = 0.025,       # Acceptable quality level (% nonconforming)
  CRQ = 0.1,         # Rejectable quality level (% nonconforming)
  alpha = 0.05,      # Producer's risk
  beta = 0.1,        # Consumer's risk
  distribution = "normal"
)
summary(var_plan)
```

Index

accProb, 2, *14*, *17*

betaff, *9*, *11, 12*

manualPlan, 3, *7*
muEst, 5, *17*

OCdata, *4*, 6, *14*, *16, 17*
OCdata.AttrPlan, *6*
OCdata.VarPlan, *6*
optAttrPlan, 7, *14*, *16*
optim, *9*, *12*
optPlan, *4*, *7*, 8
optVarPlan, *9*, 11, *16, 17*

plot.AttrPlan, 14
plot.OCdata, 15, *17*
plot.VarPlan, 16

summary.AttrPlan, 17
summary.VarPlan, 18