

# AccSamplingDesign: An R Package for Optimizing Acceptance Sampling Plans

by Ha Truong, Victor Miranda, and Roger Kissling

**Abstract** This paper introduces an R package for the optimization and visualization of acceptance sampling plans. It supports classical distributions (Binomial, Poisson, Normal) and provides the first R implementation of Beta-based plans tailored for bounded, non-normal data such as proportions. The package leverages nonlinear programming to compute optimal designs significantly faster and employs less memory than traditional grid search methods, achieving over 1000× speedup for Beta-based plans and substantial improvements for Normal-based plans. The package also delivers stable performance across a wide range of sampling plan settings and constraints. Users can visualize operating characteristic curves, compare sampling plans, and apply the tools in real-world quality control settings. We outline the statistical foundation, software architecture, and key applications.

## 1 Introduction

Acceptance sampling (AS) is a statistical quality control method used to decide whether to accept or reject a lot based on a sample, aiming to minimize inspection effort while controlling producer's and consumer's risks (Juran and Gryna, 1988). AS procedures are traditionally divided into two categories: attributes sampling and variables sampling (Schilling and Neubauer, 2017). **Attributes sampling** classifies items as conforming or nonconforming, with decisions based on the number of defectives in the sample (ISO 2859-1, 1999). While simple to apply, it discards measurement detail and typically requires sample sizes around 50% larger than those needed for variables-based methods (Mittag, 1997). In contrast, **variables sampling** uses continuous measurements such as weight, concentration, or dimension (ISO 3951-1, 2022), leading to more efficient decision-making. However, variables sampling plans typically assume normality, which limits their effectiveness for skewed or bounded data (Brown, 1985). **Beta-based AS plans** address this limitation by accommodating asymmetry, especially near the bounds of 0 and 1, and often require smaller sample sizes (Govindaraju and Kissling, 2016). Despite these benefits, Beta-based methods remain underused in practice due to limited software support.

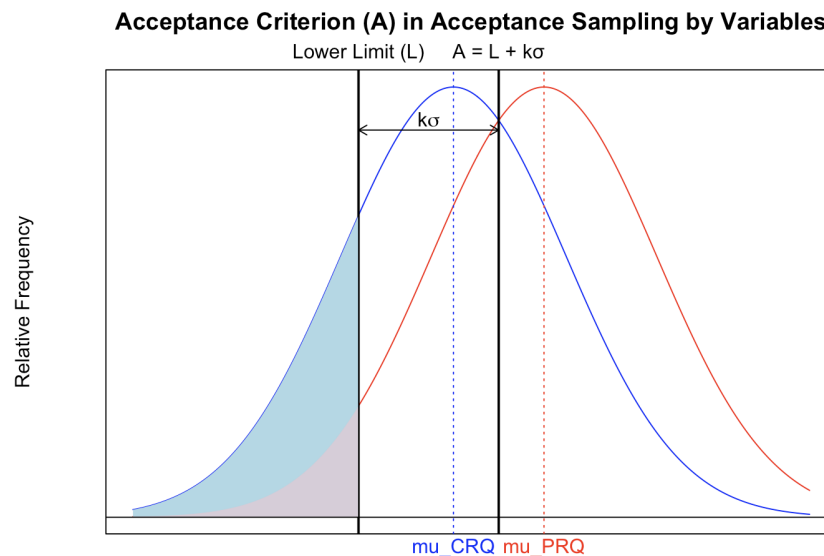
**Optimal design in AS** is crucial for efficient quality control, since it minimizes sample size while controlling error risks, directly reducing inspection time and cost. Existing tools like **AcceptanceSampling** (Andreas Kiermeier, 2008) and the methods introduced by Cano et al. (2015) focus on traditional distributions and use fixed tables or grid search for plan selection. Although nonlinear programming (NLP) has been proven to optimize Normal-based AS plans more effectively than grid search (Duarte and Saraiva, 2013), its practical adoption remains limited. Meanwhile, several packages listed in the *ExperimentalDesign* Task View—such as **OptimalDesign** (Harman and Filova, 2025) and **AlgDesign** (Wheeler and Braun, 2025)—incorporate NLP or heuristic methods for optimizing experimental designs, they do not directly support AS or the specification of risk-based design criteria.

To address this gap, we developed the **AccSamplingDesign** package (Truong et al., 2025), the first R package to support AS plans based on the Beta distribution. It leverages NLP to efficiently generate statistically optimal sampling plans under both Normal and Beta assumptions, offering a lightweight and user-friendly solution for risk-controlled design across practical constraints. The statistical foundations of this package are based on the methodologies presented in Schilling and Neubauer (2017), Wilrich (2004), and Govindaraju and Kissling (2016). This paper outlines the AS framework and models for computing the probability of acceptance, then focuses on optimization methods, the R implementation, and practical applications.

## 2 Acceptance sampling framework

AS assesses whether the **true quality level**  $p$  of the inspected lot meets required standards. Since  $p$  is unknown, decisions are based on a sample, and the **probability of acceptance** ( $P_a$ ) represents the chance that the sample leads to acceptance under a given plan. A sampling plan controls two decision errors: the **producer's risk** (Type I error)—rejecting a good-quality lot—and the **consumer's risk** (Type II error)—accepting a poor-quality lot. These are bounded by tolerances: the **producer's risk tolerance** ( $\alpha$ ) is the maximum acceptable probability of rejecting a lot at the **producer's risk quality** (PRQ)<sup>1</sup>, and the **consumer's risk tolerance** ( $\beta$ ) is the maximum acceptable probability of accepting a lot at the

<sup>1</sup>PRQ corresponds to the acceptance quality limit (AQL); see ISO 2859-1:1999.



**Figure 1:** Visualization of the acceptance criterion ( $A$ ) in variables sampling, defined relative to the lower specification limit  $L$ . The plot illustrates two process means (PRQ and CRQ), with shaded areas showing nonconforming proportions.  $A$  serves as the threshold for deciding lot acceptance based on the sample mean.

**consumer's risk quality (CRQ)**<sup>2</sup>. PRQ reflects an acceptable low defect level, while CRQ represents an unacceptable high level (ISO 3534-2, 2006), (CXG 50-2004, 2004).

An AS plan is defined by a sample size ( $n$ ) and either an acceptance number ( $c$ ) or an acceptability constant ( $k$ ). In attributes sampling,  $c$  sets the maximum number of nonconforming units allowed in the sample. In variables sampling,  $k$  is used to compute the **acceptance criterion** ( $A$ ), a threshold based on the specification limit ( $SL$ ), such as a lower ( $LSL$  or  $L$ ) or upper limit ( $USL$  or  $U$ ) (see Figure 1).

The performance of a plan is summarized by the **operating characteristic (OC) curves** (Dodge and Romig, 1944), which show  $P_a$  as a function of the true quality level, such as the nonconforming proportion (or proportion defective -  $pd$ ) (see Figure 2).

**The goal of AS design** is to identify a sampling plan that satisfies specified levels of producer's risk  $\alpha$  and consumer's risk  $\beta$ . This is achieved using the two-point estimation method at PRQ and CRQ:

$$P_a(\text{PRQ}) \geq 1 - \alpha \quad \text{and} \quad P_a(\text{CRQ}) \leq \beta. \quad (1)$$

This condition ensures that good-quality lots are accepted with high probability, and poor-quality lots are rejected with high probability. The probability of acceptance  $P_a$  depends on the sampling methods and the assumed distribution of the data.

### 3 Probability of acceptance models

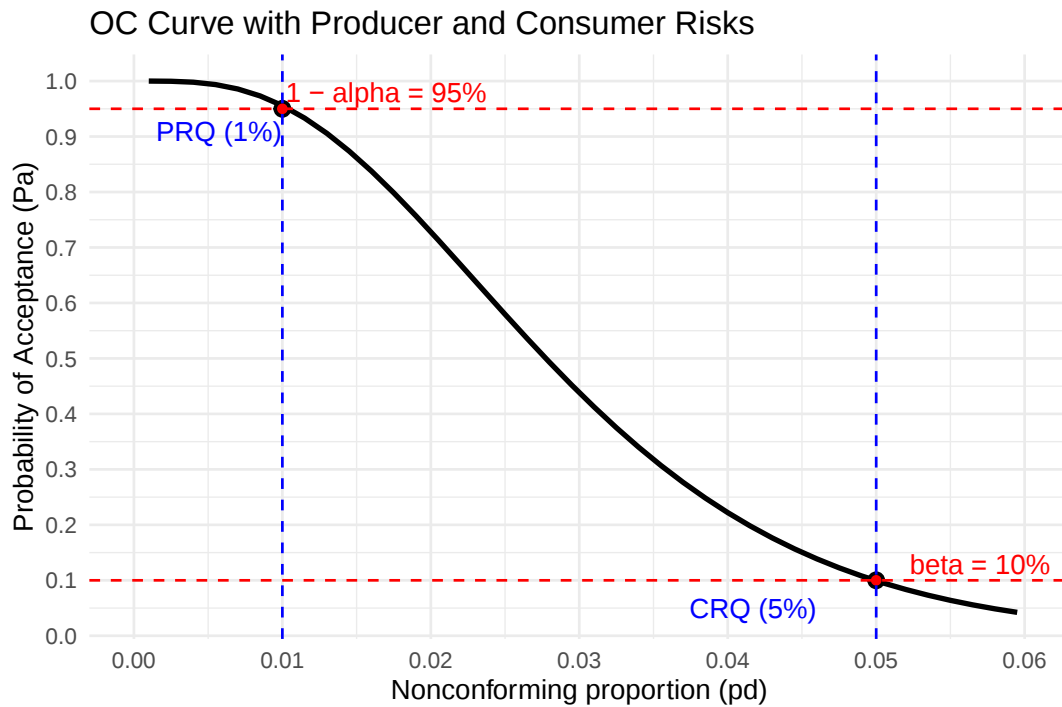
This section presents the derivation of  $P_a$  under different distributional assumptions. For variables sampling, we consider both Normal and Beta distributions, each under two scenarios—known and unknown standard deviation—as classified by Lieberman and Resnikoff (1955).

#### 3.1 Attributes sampling plans - Binomial and Poisson distributions

Attributes sampling is used to determine whether a lot should be accepted or rejected based on the number of nonconforming items. The number of nonconforming items  $X$  is modeled by either Binomial or Poisson distribution, depending on the application context.

For **Binomial model**, let  $X \sim \text{Bin}(n, p)$ , the probability of acceptance is:

<sup>2</sup>CRQ corresponds to the limiting quality (LQ); see ISO 3951-1:2022.



**Figure 2:** OC curve showing the probability of lot acceptance as a function of the nonconforming proportion. The curve includes two key points used to define the sampling plan: a 95% acceptance probability at PRQ = 0.01 and a 10% acceptance probability at CRQ = 0.05, indicating the defined producer's and consumer's risk control.

$$Pa(p) = P(X \leq c) = \sum_{i=0}^c \binom{n}{i} p^i (1-p)^{n-i}. \quad (2)$$

For **Poisson model**, let  $X \sim \text{Poisson}(\lambda = np)$ , the probability of acceptance is:

$$Pa(p) = P(X \leq c) = \sum_{i=0}^c \frac{(np)^i e^{-np}}{i!}. \quad (3)$$

Here,  $n$  is the sample size,  $c$  is the acceptance number, and  $p$  is the probability that a randomly selected item is defective. See [Schilling and Neubauer \(2017\)](#) for more details.

### 3.2 Variables sampling plans - Normal distribution with known $\sigma$

Let the quality characteristic  $X$  follow a normal distribution:  $X \sim N(\mu, \sigma^2)$ , where  $\mu$  is the process mean and  $\sigma^2$  is the process variance. The sample mean  $\bar{X}$  for a sample size  $n_\sigma$  is then distributed as  $\bar{X} \sim N(\mu, \sigma^2/n_\sigma)$ . Let  $k_\sigma$  denote the acceptability constant. To evaluate the lot for a one-sided upper specification limit ( $USL$ ), we use the test statistic:

$$Z_\sigma = \bar{X} + k_\sigma \sigma. \quad (4)$$

The lot is accepted if  $Z_\sigma \leq USL$ . The probability of lot acceptance is given by:

$$Pa(p) = P(Z_\sigma \leq USL) = \Phi\left(\sqrt{n_\sigma} \left(\Phi^{-1}(1-p) - k_\sigma\right)\right), \quad (5)$$

where  $n_\sigma$  is the sample size,  $p$  is the true nonconforming proportion and  $\Phi(\cdot)$  is the standard normal cumulative distribution function (CDF). See [Wilrich \(2004\)](#) for derivation.

### 3.3 Variables sampling plans - Normal distribution with unknown $\sigma$

When  $\sigma$  is unknown, the sample standard deviation  $S$  estimates  $\sigma$ . The test statistic (4) becomes:  $Z_s = \bar{X} + k_s S$  and the distribution of  $Z_s$  follows a **non-central t-distribution**. Then, the probability of

acceptance  $P_a(p)$  is given by:

$$P_a(p) = P(Z_s \leq USL) = 1 - F_t(k_s \sqrt{n_s}, n_s - 1, -\Phi^{-1}(p) \sqrt{n_s}), \quad (6)$$

where  $n_s$  is the sample size,  $k_s$  is the acceptability constant,  $p$  is the proportion of nonconforming and  $F_t(\cdot)$  is the CDF of the non-central t-distribution with  $n_s - 1$  degrees of freedom (Wilrich, 2004).

### 3.4 Variables sampling plans - Beta distribution with known $\theta$

For compositional data modeled as  $X \sim \text{Beta}(a, b)$ , Govindaraju and Kissling (2016) used a re-parameterization of the form:

$$\mu = \frac{a}{a+b}, \quad \theta = a+b, \quad \sigma^2 = \frac{\mu(1-\mu)}{\theta+1} \approx \frac{\mu(1-\mu)}{\theta} \quad (\text{for large } \theta), \quad (7)$$

where  $\theta$  controls the precision of the distribution. A larger  $\theta$  implies smaller variance. Thus,  $X \sim \text{Beta}(\theta\mu, \theta(1-\mu))$ . The nonconforming proportion  $p$  is fully determined by the Beta parameters  $\mu$  and  $\theta$ ; for example,  $p = \Pr(X < L \mid \mu, \theta)$  for an LSL and  $p = \Pr(X > U \mid \mu, \theta)$  for a USL. Consider an AS plan with sample size  $n$ , the lot is accepted if:

$$\begin{cases} \bar{X} - k\hat{\sigma} \geq L, & (\text{case of LSL}) \\ \bar{X} + k\hat{\sigma} \leq U, & (\text{case of USL}) \end{cases} \quad (8)$$

where  $\bar{X}$  is the sample mean,  $k$  is the acceptability constant and  $\hat{\sigma} \approx \sqrt{\frac{\bar{X}(1-\bar{X})}{\theta}}$ . In both LSL and USL cases, rearranging (8) yields the same quadratic inequality:

$$(\theta + k^2) \bar{X}^2 - (2\theta L + k^2) \bar{X} + \theta L^2 \geq 0. \quad (9)$$

Solving this quadratic yields the roots:

$$z_1 = \frac{2\theta L + k^2 - \sqrt{\Delta}}{2(\theta + k^2)}, \quad z_2 = \frac{2\theta L + k^2 + \sqrt{\Delta}}{2(\theta + k^2)}, \quad (10)$$

where the discriminant is:  $\Delta = (2\theta L + k^2)^2 - 4(\theta + k^2)(\theta L^2)$ . Let  $F_b(z)$  denote the CDF of the Beta distribution. Using Equation (8), the probability of acceptance  $P_a(p)$  can then be calculated as follows:

$$P_a(p) = \begin{cases} P(\bar{X} - k\hat{\sigma} \geq L) = 1 - F_b(z_2), & (\text{for LSL case}) \\ P(\bar{X} + k\hat{\sigma} \leq U) = F_b(z_1), & (\text{for USL case}) \end{cases} \quad (11)$$

## 4 Optimization methods for sampling plan design

This section presents AS design and optimization methods based on the  $P_a(\cdot)$  functions derived earlier. Given producer's risk  $\alpha$  and consumer's risk  $\beta$ , the goal is to find a sampling plan that meets the risk conditions in Equation (1) while minimizing sample size. While closed-form solutions and search methods are available in some cases (e.g., for attributes sampling or Normal-based plans with known  $\sigma$ ), NLP provides an effective and flexible solution when such formulas are unavailable or intractable.

**For Normal-based variables sampling and when  $\sigma$  is known**, a closed-form solution exists for the optimal plan  $(n_\sigma, k_\sigma)$ . The acceptance probability is defined in Equation (5), and the risk conditions in (1) lead to the following expressions (Wilrich, 2004):

$$n_\sigma = \left( \frac{\Phi^{-1}(1-\alpha) + \Phi^{-1}(1-\beta)}{\Phi^{-1}(1-\text{PRQ}) - \Phi^{-1}(1-\text{CRQ})} \right)^2, \quad (12)$$

$$k_\sigma = \frac{\Phi^{-1}(1-\text{PRQ}) \cdot \Phi^{-1}(\beta) + \Phi^{-1}(1-\text{CRQ}) \cdot \Phi^{-1}(1-\alpha)}{\Phi^{-1}(1-\alpha) + \Phi^{-1}(\beta)}. \quad (13)$$

**For Normal-based variables sampling and when  $\sigma$  is unknown**, the acceptance probability  $P_a(\cdot)$  is defined using a non-central  $t$ -distribution, as in Equation (6). Since no closed-form solution exists for this case (though an approximation is available from Wilrich (2004)), we formulate a nonlinear optimization problem to determine the optimal sample size  $n_s$  and acceptability constant  $k_s$ :

$$\text{Objective} = \min_{n_s, k_s} (|\text{PR} - \alpha| + |\text{CR} - \beta|). \quad (14)$$

Here, the producer's risk is defined as  $PR = 1 - P_a(\text{PRQ})$ , where PRQ is the quality level at which producer's risk is evaluated. Similarly, the consumer's risk is  $CR = P_a(\text{CRQ})$ , with CRQ as the corresponding quality level. The optimization is performed using the derivative-free method from [Nelder and Mead \(1965\)](#) via the `optim()` function in R. Although numerous advanced NLP solvers are available in the *Optimization* task view, `optim()` was adequate for this setting for several reasons: First, the 2-dimensional problem  $(n_s, k_s)$  is well suited to derivative-free methods. Second, the objective function contains absolute value operations and involves non-central  $t$ -distribution computations that can introduce numerical non-smoothness. Third, initializing from the Normal-known- $\sigma$  solution provides a warm start near the optimum, reducing concerns about local minima. Finally, using base R's `optim()` avoids external dependencies while providing adequate performance for this application.

**For Beta-based variables sampling and when  $\theta$  is known**, the optimal sampling parameters  $(n_\theta, k_\theta)$  are found by solving the risk conditions in (1), using the  $P_a$  function from (11). To enable efficient search, we reformulate the task as a NLP problem with a penalized objective function:

$$\text{Objective} = \min_{n_\theta, k_\theta} \left[ n_\theta + \lambda \cdot \left( \max(\text{PR} - \alpha, 0)^2 + \max(\text{CR} - \beta, 0)^2 \right) \right], \quad (15)$$

where  $\text{PR} = 1 - P_a(\text{PRQ})$  and  $\text{CR} = P_a(\text{CRQ})$ . A penalty coefficient  $\lambda$  (i.e.,  $\lambda = 1e4$ ) is introduced to enforce these constraints by penalizing violations in the objective function. The implementation of (15) used the `optim()` function in R with the L-BFGS-B method ([Byrd et al., 1995](#)). Initial values for  $(n_\theta, k_\theta)$  are taken from Equations (12) and (13), placing the search close to the optimum and reducing the risk of slow convergence or suboptimal solutions. Practical search ranges for the parameters are defined around these initial values, and L-BFGS-B naturally enforces the box constraints, which improves stability and avoids unrealistic proposals.

**When  $\theta$  is unknown**, both the sample mean  $\bar{X}$  and  $\theta$  must be estimated—e.g., via the `betaff()` function from the **VGAM** package ([Yee, 2025](#))—introducing additional uncertainty into the plan. To account for this, the adjusted sample size  $n_{\text{adj}}$  is computed as:

$$n_{\text{adj}} = \left( 1 + \rho \cdot k_\theta^2 \right) n_\theta, \quad (16)$$

where  $(n_\theta, k_\theta)$  are the optimal parameters from Equation (15), and  $\rho$  reflects the added variability from estimating  $\theta$ . This adjustment is similar in form to the Normal-based approximation by [Wilrich \(2004\)](#). Based on simulation by [Govindaraju and Kissling \(2016\)](#), a conservative value  $\rho \approx 0.85$  is recommended to adjust the sample size when  $\theta$  is estimated.

## 5 Package overview

The **AccSamplingDesign** package supports attributes sampling using the Binomial or Poisson distributions, as well as variables sampling based on Normal or Beta distributions. It centers on two core classes: `AttrPlan` for attributes sampling and `VarPlan` for variables sampling. The unified interface `optPlan()` selects `optAttrPlan()` or `optVarPlan()` based on the chosen distribution. The package also provides S3 methods for summary and visualization, and functions for generating OC data to compare manual and optimal designs (Table 1).

This R package is available on CRAN under the GPL-3 license. The released version can be installed as follows:

```
install.packages("AccSamplingDesign")
```

The development version is maintained at [GitHub repository](#) and can be installed through the **devtools** package ([Wickham et al., 2022](#)) as follows:

```
devtools::install_github("vietha/AccSamplingDesign")
```

### 5.1 Method `optPlan()` as the primary interface

The `optPlan()` function serves as the primary entry point for users to design optimal AS plans across a range of classical distributions. It provides a unified interface that automatically dispatches to either `optAttrPlan()` or `optVarPlan()`, depending on whether the specified distribution and parameters correspond to an attributes or variables plan. This design simplifies usage for most users, abstracting the underlying logic while still allowing advanced users to call lower-level methods directly for finer control. The function supports Binomial, Poisson, Normal, and Beta distributions, enabling both attributes-based and variables-based sampling schemes.

**Table 1:** Summary of sampling plan methods and utilities provided by the package.

| Method                     | Description                                                                                                                             |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------|
| <code>optPlan()</code>     | Unified entry point; automatically selects between attributes or variables sampling based on inputs.                                    |
| <code>optAttrPlan()</code> | Designs attributes sampling plans using the Binomial or Poisson distribution.                                                           |
| <code>optVarPlan()</code>  | Designs variables sampling plans for Normal or Beta distributions. Supports known/unknown standard deviation and one-sided limits.      |
| <code>manualPlan()</code>  | Manually generate attribute or variable sampling plans based on user-specified parameters, typically used for evaluation or comparison. |
| <code>OCdata()</code>      | Generates data for OC curves based on an optimal or manual plan.                                                                        |
| <code>summary()</code>     | Summarizes a sampling plan object, including key values such as sample size and risk levels.                                            |
| <code>plot()</code>        | Visualizes sampling plans or OC curve data. Adapts content and axis labeling based on the input object.                                 |

For attributes sampling, the optimal plan  $(n, c)$  is obtained by evaluating the Binomial case in Equation (2) or the Poisson case in Equation (3), followed by a discrete search over feasible  $(n, c)$  pairs to find the smallest sample size satisfying the risk constraints in (1). For Normal-based variables sampling and  $\sigma$  is known, the optimal plan  $(n, k)$  is computed using closed-form formulas in Equation (12) and (13). When  $\sigma$  is unknown, the plan is obtained by minimizing the objective in Equation (14) via a NLP method. For Beta-based variables sampling, when the precision parameter  $\theta$  is known, the optimal plan is obtained by minimizing the objective function in Equation (15) using NLP. When  $\theta$  is unknown, the approach relies on the approximation described in Equation (16).

This flexible design allows users to specify target quality thresholds—PRQ and CRQ—along with the associated risks  $\alpha$  and  $\beta$ , and specification limits such as the LSL or USL for variables plans. Internally, the function manages distribution selection, parameter validation, and optimization, returning an object of class `AttrPlan` or `VarPlan`, ready for further evaluation or visualization. By streamlining the modeling workflow while supporting advanced configurations, `optPlan()` facilitates efficient design of AS plans across diverse quality control and compliance settings.

## 5.2 Generating operating characteristic curves with `OCdata()`

The `OCdata()` function facilitates the generation of OC curve data from either optimized sampling plans—such as those produced by `optPlan()` or user-defined parameters via `manualPlan()`. It accommodates both attributes and variables sampling plans by adapting calculations to the specified distribution, and computes acceptance probabilities across a specified range of nonconforming proportions. The output is a structured `OCdata` object that encapsulates key vectors such as `pd` (nonconforming proportions), `paccept` (acceptance probabilities), and, for variables sampling plans, `process_means` along with sample size and plan-specific parameters. This organization facilitates seamless downstream visualization and analysis.

## 5.3 Visualizing Sampling Plans with `plot()`

Visualization of sampling plans and their corresponding OC curves is supported via a unified `S3 plot()` method. The method adapts automatically based on the class of the input object—`AttrPlan`, `VarPlan`, or `OCdata`—and sets appropriate axis labels, reference lines, and graphical elements for each context. The x-axis scale can be controlled using the `by` argument, allowing users to plot against the proportion nonconforming or, where applicable, the process mean. The function uses base R graphics and supports further customization through standard graphical parameters.

# 6 Practical examples

This section demonstrates how to use the [AccSamplingDesign](#) package in R to design AS plans. Examples cover both attributes and variables sampling plans. We also compare user-defined and optimal plans, evaluate the performance of Normal vs. Beta-based AS methods, and demonstrate

how to use `OCdata()` to create custom plans and plot OC curves by nonconforming proportion ( $pd$ ) or process mean. Before running the examples, ensure the package is loaded into the R environment.

```
library(AccSamplingDesign)
```

## 6.1 Sampling plan design for defective rates in packaged goods

This example illustrates an attributes sampling plan for a packaged goods manufacturer aiming to control the defect rate. Because the data are discrete (e.g., counts of defective units), a Binomial distribution assumption is appropriate. The plan is designed to ensure a low probability of rejecting acceptable batches with a defect rate of 1% ( $PRQ = 0.01$ )—reflecting the producer’s risk—and a low probability of accepting unacceptable batches with a defect rate of 5% ( $CRQ = 0.05$ )—reflecting the consumer’s risk. The risk constraints are set to  $\alpha = 0.02$  and  $\beta = 0.15$ . The optimal plan is obtained using the generic `optPlan()` function.

```
optPlan(distribution = "binomial", PRQ = 0.01, CRQ = 0.05, alpha = 0.02, beta = 0.15)

#> AttrPlan object:
#> Distribution: binomial
#> Sample size (n): 144
#> Acceptance number (c): 4
#>
#> NOTE: summary(plan) for detail report,
#>       plot(plan) for quick OC visualization,
#>       OCdata(plan) to extract data for evaluation and custom plots.
```

This plan recommends inspecting 144 items from each batch and accepting the batch if no more than 4 defective items are found. It balances the producer’s and consumer’s risks, offering a practical guideline for quality control in production. Users can explore the plan further using `summary()`, `plot()`, or `OCdata()` to evaluate performance under other defect rates or to create customized visualizations.

## 6.2 Sampling plan for moisture composition in milk products

This example illustrates a variables sampling plan for monitoring moisture content in milk products, where the data are continuous and represent proportions. Such fractional data can be modeled using either a Normal or a Beta distribution. We use this example to compare the performance of both approaches. The upper specification limit (USL) is set at 0.05. The plan targets acceptance of high-quality batches with a defect rate below 0.5% ( $PRQ = 0.005$ ), and rejection of low-quality batches with a defect rate above 3% ( $CRQ = 0.03$ ), while controlling producer’s risk at  $\alpha = 0.05$  and consumer’s risk at  $\beta = 0.10$ . We first design a Normal-based plan, followed by a Beta-based plan assuming a known precision parameter  $\theta = 500$ .

```
normal_plan <- optPlan(PRQ = 0.005, CRQ = 0.03, alpha = 0.05, beta = 0.10,
                      distribution = "normal", sigma_type = "known")
summary(normal_plan)

#> Variables Acceptance Sampling Plan
#> Distribution: normal
#> Sample Size (n, rounded up): 18 [raw n = 17.72779 ]
#> Acceptability Constant (k): 2.185
#> Population Standard Deviation: known
#> Producer's Risk (PR = 0.05 ) at PRQ = 0.005
#> Consumer's Risk (CR = 0.1 ) at CRQ = 0.03

beta_plan <- optPlan(PRQ = 0.005, CRQ = 0.03, alpha = 0.05, beta = 0.10, USL = 0.05,
                    distribution = "beta", theta = 500, theta_type = "known")
summary(beta_plan)

#> Variables Acceptance Sampling Plan
#> Distribution: beta
#> Sample Size (n, rounded up): 16 [raw n = 15.22157 ]
#> Acceptability Constant (k): 2.485
```

```
#> Population Precision Parameter (theta): known
#> Producer's Risk (PR = 0.05072676 ) at PRQ = 0.005
#> Consumer's Risk (CR = 0.1004133 ) at CRQ = 0.03
#> Upper Specification Limit (USL): 0.05
```

It is worth noting that, in this case, the Beta-based sampling plan requires a smaller sample size ( $n = 16$ ) compared to the Normal-based sampling plan ( $n = 18$ ). This aligns with the findings of [Govindaraju and Kissling \(2016\)](#).

### 6.3 Sampling plan for Vitamin-A in a food product

This example designs a variables sampling plan for Vitamin-A content, where data are heavily skewed and close to zero—characteristics well handled by the Beta distribution. Using a known precision parameter  $\theta = 6.6 \times 10^8$  and a lower specification limit (LSL) of  $5.65 \times 10^{-6}$ , the plan controls producer's risk at  $\alpha = 0.05$  for a preferred quality level (PRQ = 2.5%) and consumer's risk at  $\beta = 0.10$  for a worse quality level (CRQ = 10%). We begin by computing the optimal Beta-based sampling plan under these conditions:

```
vita_plan <- optPlan(PRQ = 0.025, CRQ = 0.10, alpha = 0.05, beta = 0.10,
                    distribution = "beta", theta_type = "known",
                    theta = 6.6e8 , LSL = 5.65e-6)
summary(vita_plan)

#> Variables Acceptance Sampling Plan
#> Distribution: beta
#> Sample Size (n, rounded up): 19 [raw n = 18.66251 ]
#> Acceptability Constant (k): 1.571
#> Population Precision Parameter (theta): known
#> Producer's Risk (PR = 0.05087403 ) at PRQ = 0.025
#> Consumer's Risk (CR = 0.1005144 ) at CRQ = 0.1
#> Lower Specification Limit (LSL): 5.65e-06
```

The `plot()` method visualizes Beta-based AS plans. By default, it shows the acceptance probability against the nonconforming proportion  $pd$  (Figure 3).

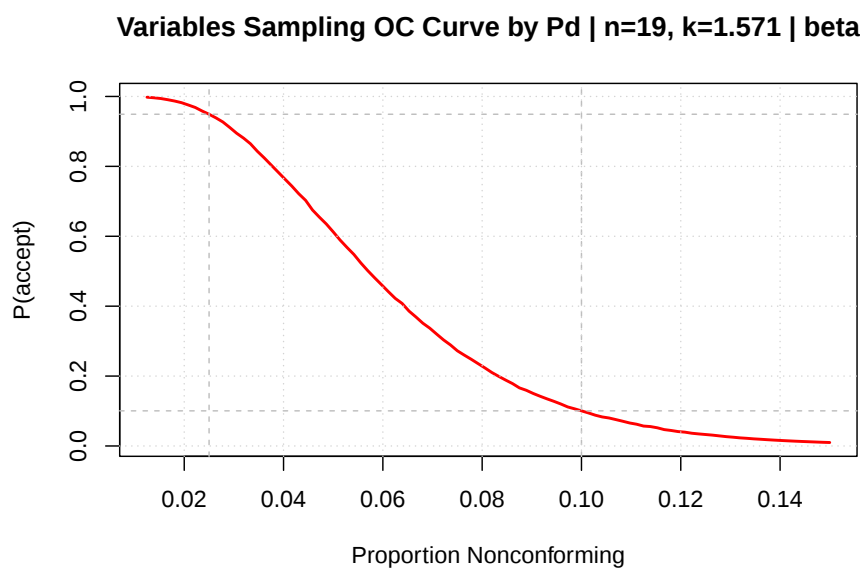
```
plot(vita_plan)
```

Setting `by = "mean"` changes the x-axis to the process mean (Figure 4).

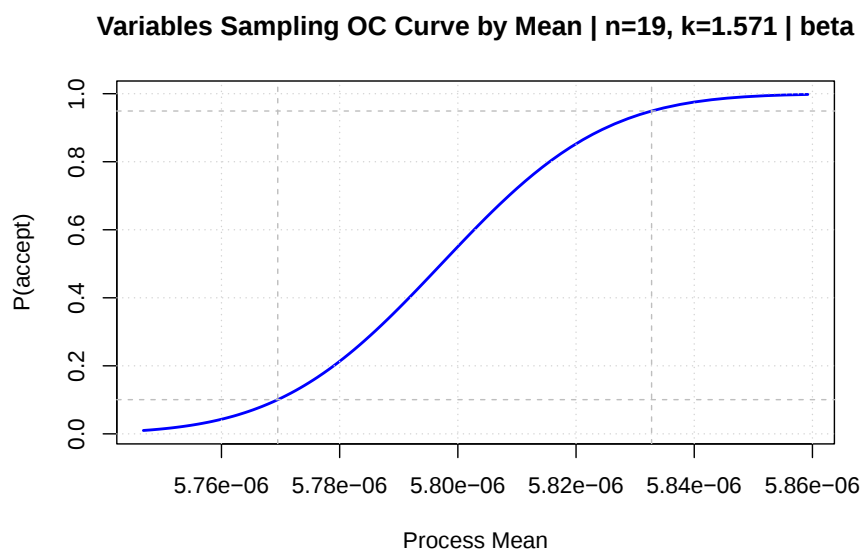
```
plot(vita_plan, by = "mean")
```

**Sensitivity analysis** can be performed by varying the sample size  $n$  or the acceptability constant  $k$  using `manualPlan()` and `OCdata()`, based on the optimal plan generated by `optPlan()`. The evaluated plans are then compared to the optimal plan using OC curves. The visualization was produced using [ggplot2](#) ([Wickham, 2016](#)), and the interactive version was generated with [plotly](#) ([Sievert, 2020](#)) (see Figure 5).

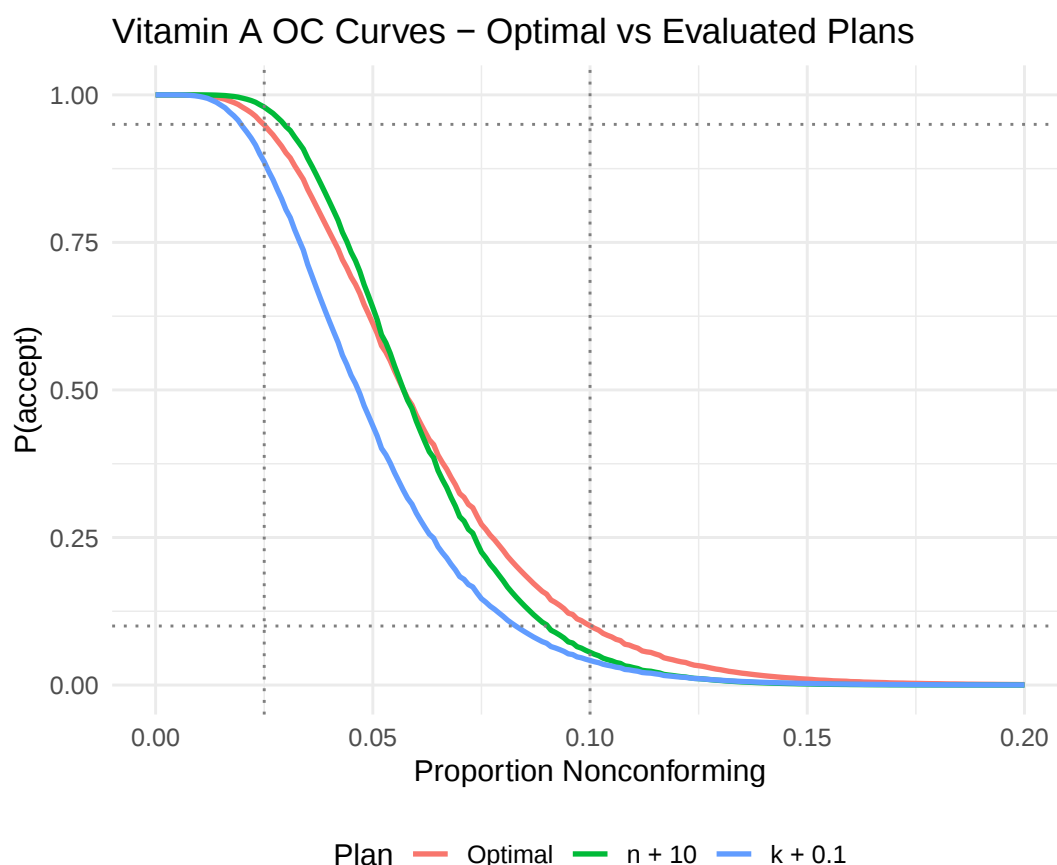
```
# Sequence of defect rates for OC curve
pd <- seq(0, 0.2, by = 0.001)
# Create manual plans to compare
plan_n_up <- manualPlan(n = vita_plan$sample_size + 10, k = vita_plan$k,
                        distribution = "beta", LSL = 5.65e-6, theta = 6.6e8)
plan_k_up <- manualPlan(n = vita_plan$sample_size, k = vita_plan$k + 0.1,
                        distribution = "beta", LSL = 5.65e-6, theta = 6.6e8)
# Combine OC data into a data frame
oc_df <- bind_rows(
  as.data.frame(OCdata(vita_plan, pd = pd)) |> mutate(Plan = "Optimal"),
  as.data.frame(OCdata(plan_n_up, pd = pd)) |> mutate(Plan = "n + 10"),
  as.data.frame(OCdata(plan_k_up, pd = pd)) |> mutate(Plan = "k + 0.1")
) |> mutate(Plan = factor(Plan, levels = c("Optimal", "n + 10", "k + 0.1")))
# Plot OC curves
plt_plan_compare <- ggplot(oc_df, aes(x = pd, y = paccept, color = Plan)) +
```



**Figure 3:** OC curve for Vitamin A Beta-based sampling showing the probability of acceptance across proportions nonconforming. Vertical dashed lines mark the producer's risk quality (PRQ = 2.5%) and consumer's risk quality (CRQ = 10%) levels. Horizontal dashed lines represent the corresponding acceptance probability constraints: at least 95% at PRQ (producer's risk  $\leq 5\%$ ) and at most 10% at CRQ (consumer's risk  $\leq 10\%$ ).



**Figure 4:** OC curve for Vitamin A Beta-based sampling plotted against the process mean, showing the probability of acceptance across different process mean values. Vertical dashed lines mark the mean levels corresponding to the producer's risk quality (PRQ) and consumer's risk quality (CRQ). Horizontal dashed lines represent the corresponding acceptance probability constraints: at least 95% at PRQ (producer's risk  $\leq 5\%$ ) and at most 10% at CRQ (consumer's risk  $\leq 10\%$ ).



**Figure 5:** OC curves for Vitamin A Beta-based acceptance sampling. The optimal plan is compared with two evaluated plans: one with increased sample size ( $n + 10$ ) and another with a higher acceptability constant ( $k + 0.1$ ). The plot illustrates how changes in  $n$  or  $k$  affect the shape and slope of the curve, thereby influencing the producer's and consumer's risks.

```
geom_line(linewidth = 1) +
  geom_vline(xintercept = c(0.025, 0.10), linetype = "dotted", color = "gray50") +
  geom_hline(yintercept = c(0.95, 0.10), linetype = "dotted", color = "gray50") +
  labs(title = "Vitamin A OC Curves – Optimal vs Evaluated Plans",
       x = "Proportion Nonconforming", y = "P(accept)") +
  theme_minimal(base_size = 12) + theme(legend.position = "bottom")
```

**For case of unknown  $\theta$ ,** the sampling plan is designed with the following commands:

```
vita_plan2 <- optPlan(PRQ = 0.025, CRQ = 0.10, alpha = 0.05, beta = 0.10,
                     distribution = "beta", LSL = 5.65e-6,
                     theta = 6.6e8, theta_type = "unknown")
summary(vita_plan2)
```

```
#> Variables Acceptance Sampling Plan
#> Distribution: beta
#> Sample Size (n, rounded up): 59 [raw n = 58.84879 ]
#> Acceptability Constant (k): 1.571
#> Population Precision Parameter (theta): unknown
#> Producer's Risk (PR = 0.04934987 ) at PRQ = 0.025
#> Consumer's Risk (CR = 0.0985057 ) at CRQ = 0.1
#> Lower Specification Limit (LSL): 5.65e-06
```

## 7 Computational performance and benchmarking

All computations were performed in R version 4.5.1 (R Core Team, 2025) on a Mac system with an Apple M2 chip, 8 GB RAM, and macOS 15.4.1. To assess computational efficiency, we benchmarked

**Table 2:** Comparison of AS plans under the Normal model with unknown standard deviation. For each PRQ/CRQ pair, the table reports AS plan (n, k), and computation time from two R packages: *AccSamplingDesign* (using NLP) and *AcceptanceSampling* (grid search). The sample size (n) is rounded up to the next integer for practical application.

| PRQ    | CRQ  | Package                   | Time (ms) | n   | k     |
|--------|------|---------------------------|-----------|-----|-------|
| 0.0050 | 0.03 | <i>AccSamplingDesign</i>  | 0.79      | 62  | 2.192 |
| 0.0050 | 0.03 | <i>AcceptanceSampling</i> | 10.56     | 62  | 2.194 |
| 0.0075 | 0.03 | <i>AccSamplingDesign</i>  | 0.76      | 93  | 2.127 |
| 0.0075 | 0.03 | <i>AcceptanceSampling</i> | 15.80     | 93  | 2.127 |
| 0.0100 | 0.03 | <i>AccSamplingDesign</i>  | 1.49      | 138 | 2.079 |
| 0.0100 | 0.03 | <i>AcceptanceSampling</i> | 24.35     | 138 | 2.079 |
| 0.0125 | 0.03 | <i>AccSamplingDesign</i>  | 1.41      | 205 | 2.041 |
| 0.0125 | 0.03 | <i>AcceptanceSampling</i> | 41.75     | 205 | 2.041 |
| 0.0150 | 0.03 | <i>AccSamplingDesign</i>  | 1.32      | 309 | 2.009 |
| 0.0150 | 0.03 | <i>AcceptanceSampling</i> | 63.68     | 309 | 2.009 |
| 0.0175 | 0.03 | <i>AccSamplingDesign</i>  | 1.56      | 491 | 1.982 |
| 0.0175 | 0.03 | <i>AcceptanceSampling</i> | 79.08     | 491 | 1.982 |
| 0.0200 | 0.03 | <i>AccSamplingDesign</i>  | 1.75      | 836 | 1.957 |
| 0.0200 | 0.03 | <i>AcceptanceSampling</i> | 90.53     | 836 | 1.957 |

*AccSamplingDesign* version 0.0.8 against existing tools for designing variables AS plans, focusing on models based on the Normal and Beta distributions.

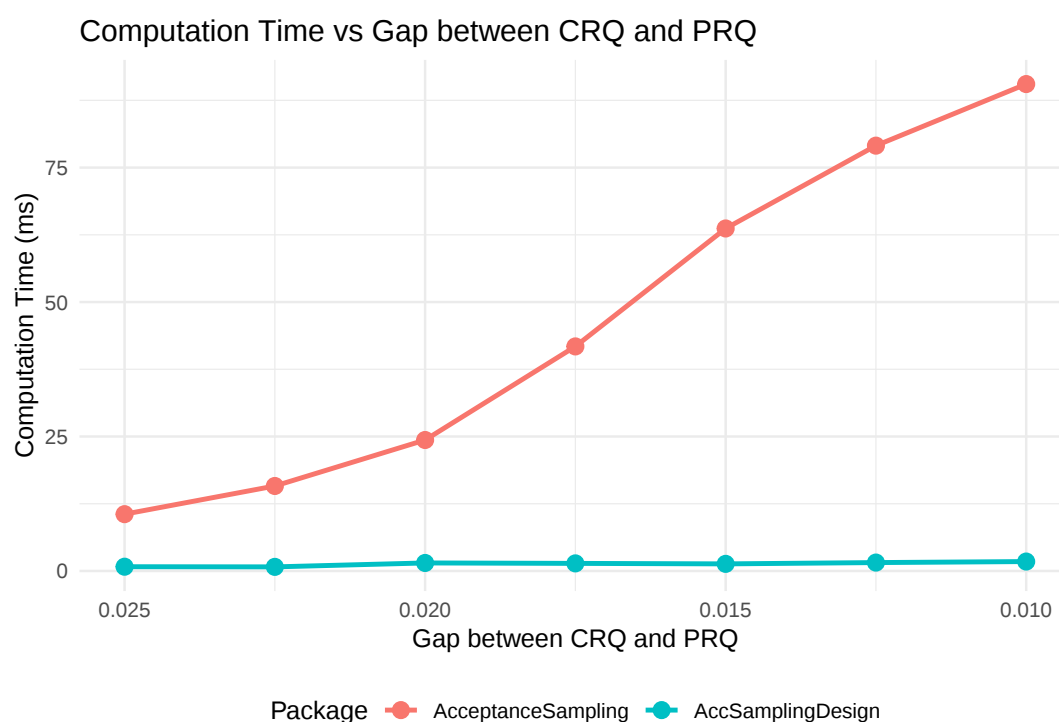
For Normal-based AS plans with unknown standard deviation, we reanalyzed the *moisture sampling plan* example and compared results between our NLP-based method, implemented in *AccSamplingDesign*, and the grid search approach used by the *AcceptanceSampling* package (version 1.0-10), across varying PRQ values (see Table 2). As shown by Duarte and Saraiva (2013), NLP outperforms grid search in both feasibility and efficiency. Our findings support this observation: while both approaches produced nearly identical plans (n,k), the NLP method was significantly faster, with speed improvements of approximately 50× when PRQ approached CRQ. It also demonstrated consistent processing time and precision across repeated runs, regardless of the PRQ value (see Figure 6).

To the best of our knowledge, no existing R package provides automated support for the design of Beta-based AS plans. We evaluated the performance of *AccSamplingDesign* against the original R scripts developed by Govindaraju and Kissling (2016). In the case study on *vitamin A sampling*, their grid search procedure required over 20 seconds to identify an optimal plan, whereas our implementation achieved the same result in under 0.02 seconds, operating over 1000 times faster and with substantially reduced memory usage. Benchmarking was performed using the *bench* package (Hester et al., 2025). Complete results and reproducible code are provided in the supplementary materials (see *beta\_plans\_benchmarking.Rmd*).

## 8 Conclusion

The *AccSamplingDesign* package makes several important contributions to AS design and evaluation within R. First, it introduces support for Beta-based variables sampling plans, which are particularly well suited for bounded quality characteristics such as proportions and concentrations commonly encountered in food, pharmaceutical, and biomedical industries. Compared to Normal-based plans under equivalent settings, Beta-based plans typically require smaller sample sizes, thereby enhancing inspection efficiency. Second, it uses a closed-form expression for the acceptance probability, avoiding computationally intensive Monte Carlo simulation (Govindaraju and Kissling, 2016), and successfully applies NLP for AS plan optimization. This approach outperforms traditional grid search for both Normal- and Beta-based plans, achieving over 1000-fold speed gains for Beta-based plans, making them feasible for real-time use in interactive tools, dashboards, and automated quality control pipelines. Third, the package supports ISO standards and offers a user-friendly design that facilitates practical application of AS plan development in industry settings. It includes functions for visualizing and comparing OC curves, helping users evaluate and communicate plan performance effectively.

Overall, *AccSamplingDesign* fills a notable gap in the R ecosystem by offering a flexible, risk-based framework for AS plan design that aligns with ISO standards. It caters to statisticians, quality control



**Figure 6:** Computation time comparison between `AccSamplingDesign` and `AcceptanceSampling` for the Normal model with unknown standard deviation, as the CRQ–PRQ gap narrows (harder cases on the right). `AccSamplingDesign` is consistently faster and maintains stable performance, while `AcceptanceSampling` slows down.

practitioners, and industry users, promoting accessibility and practical adoption. Future development plans include extending the package to support multivariate sampling plans and measurement error adjustment for compositional data, thereby expanding its utility in advanced quality control settings.

## References

- Andreas Kiermeier. Visualizing and assessing acceptance sampling plans: The R package `AcceptanceSampling`. *Journal of Statistical Software*, 26(6):1–20, 2008. doi: 10.18637/jss.v026.i06. [p368]
- G. H. Brown. Acceptance sampling plans for material in bulk form using a Beta model. *Journal of Quality Technology*, 17:134–139, 1985. doi: 10.1080/00224065.1985.11978952. [p368]
- 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. [p372]
- E. L. Cano, J. M. Moguerza, and M. P. Corcoba. *Quality Control with R: An ISO Standards Approach*. Springer, 2015. ISBN 978-3-319-23454-1. doi: 10.1007/978-3-319-24046-6. [p368]
- CXG 50-2004. General guidelines on sampling (CXG 50-2004), 2004. URL <https://www.fao.org/fao-who-codexalimentarius/en/>. Rome: FAO/WHO. [p369]
- H. F. Dodge and H. G. Romig. *Sampling Inspection Tables: Single and Double Sampling*. John Wiley & Sons, 1944. URL <https://archive.org/details/samplinginspecti00dodg>. [p369]
- B. P. M. Duarte and P. M. Saraiva. Optimal design of acceptance sampling plans by variables for nonconforming proportions when the standard deviation is unknown. *Communications in Statistics - Simulation and Computation*, 42(6):1318–1342, 2013. doi: 10.1080/03610918.2012.665548. [p368, 378]
- K. Govindaraju and R. C. Kissling. Sampling plans for Beta distributed compositional fractions. *Chemometrics and Intelligent Laboratory Systems*, 151:103–107, 2016. doi: 10.1016/j.chemolab.2015.12.009. [p368, 371, 372, 375, 378]
- R. Harman and L. Filova. *OptimalDesign: A Toolbox for Computing Efficient Designs of Experiments*, 2025. URL <http://www.iam.fmph.uniba.sk/design/>. R package version 1.0.2.1. [p368]

- J. Hester, D. Vaughan, and D. Schmidt. *bench: High Precision Timing of R Expressions*, 2025. URL <https://CRAN.R-project.org/package=bench>. R package version 1.1.4. [p378]
- ISO 2859-1. *Sampling Procedures for Inspection by Attributes — Part 1: Sampling Schemes Indexed by Acceptance Quality Limit (AQL) for Lot-by-Lot Inspection*. International Organization for Standardization, Geneva, Switzerland, 1999. URL <https://www.iso.org/standard/1141.html>. The primary standard for attribute-based acceptance sampling plans. [p368]
- ISO 3534-2. *Statistics – vocabulary and symbols – part 2: Applied statistics*, 2006. URL <https://www.iso.org/standard/40189.html>. [p369]
- ISO 3951-1. *Sampling Procedures for Inspection by Variables — Part 1: Specification for Single Sampling Plans Indexed by Acceptance Quality Limit (AQL) for Lot-by-Lot Inspection*. International Organization for Standardization, Geneva, Switzerland, 2022. URL <https://www.iso.org/standard/74706.html>. Complementary to ISO 2859 for variable-based acceptance sampling. [p368]
- J. M. Juran and F. M. Gryna. *Juran's Quality Control Handbook*. McGraw-Hill, 4th edition, 1988. URL <https://search.worldcat.org/en/title/17546189>. [p368]
- G. J. Lieberman and G. J. Resnikoff. Sampling plans for inspection by variables. *Journal of Quality Technology*, July 15 1955. doi: 10.2307/2280972. [p369]
- H. J. Mittag. Measurement error effects on the performance of process capability indices. In H. J. Lenz and P. T. Wilrich, editors, *Frontiers in Statistical Quality Control*, volume 5. Physica, Heidelberg, 1997. doi: 10.1007/978-3-642-59239-3\_15. [p368]
- 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. [p372]
- R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2025. URL <https://www.R-project.org/>. [p377]
- E. G. Schilling and D. V. Neubauer. *Acceptance Sampling in Quality Control*. CRC Press, 3rd edition, 2017. doi: 10.4324/9781315120744. [p368, 370]
- C. Sievert. *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC, 2020. ISBN 9781138331457. URL <https://plotly-r.com>. [p375]
- H. Truong, V. Miranda, and R. Kissling. *AccSamplingDesign: Acceptance Sampling Plans Design*, 2025. URL <https://CRAN.R-project.org/package=AccSamplingDesign>. R package version 0.0.8. [p368]
- B. Wheeler and J. Braun. *AlgDesign: Algorithmic Experimental Design*, 2025. URL <https://CRAN.R-project.org/package=AlgDesign>. R package version 1.2.1.2. [p368]
- H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016. ISBN 978-3-319-24277-4. URL <https://ggplot2.tidyverse.org>. [p375]
- H. Wickham, J. Hester, W. Chang, and J. Bryan. *devtools: Tools to Make Developing R Packages Easier*, 2022. URL <https://CRAN.R-project.org/package=devtools>. R package version 2.4.5. [p372]
- P. T. Wilrich. Single sampling plans for inspection by variables under a variance component situation. In H. J. Lenz and P. T. Wilrich, editors, *Frontiers in Statistical Quality Control 7*, volume 7 of *Frontiers in Statistical Quality Control*. Physica, Heidelberg, 2004. doi: 10.1007/978-3-7908-2674-6\_4. [p368, 370, 371, 372]
- T. W. Yee. *VGAM: Vector Generalized Linear and Additive Models*, 2025. URL <https://CRAN.R-project.org/package=VGAM>. R package version 1.1-13. [p372]

Ha Truong  
Auckland University of Technology  
55 Wellesley Street East, Auckland City, New Zealand  
<https://github.com/vietha>  
ORCID: 0009-0008-2986-1425  
[ha.truong@aut.ac.nz](mailto:ha.truong@aut.ac.nz)

Victor Miranda  
Auckland University of Technology  
55 Wellesley Street East, Auckland City, New Zealand

<https://academics.aut.ac.nz/victor.miranda>  
*ORCID:* 0000-0001-8920-3573  
[victor.miranda@aut.ac.nz](mailto:victor.miranda@aut.ac.nz)

*Roger Kissling*  
*Fonterra Co-operative*  
*New Zealand*  
[roger.kissling@fonterra.com](mailto:roger.kissling@fonterra.com)