Supplementary Materials to: Equivalence Testing With Particle Size Distribution Data: Methods and Applications in the Development of Inhalative Drugs

THOMAS HOFFELDER

Boehringer Ingelheim GmbH & Co. KG, Germany

STEFAN WELLEK

Dept. of Biostatistics, CIMH Mannheim/Univ. of Heidelberg &

Dept. of Medical Biostatistics, Epidemiology and Informatics,

Univ. of Mainz, Germany

November 25, 2015

Notational Preliminaries: Section numbers, parenthesized equation numbers, and table numbers followed by (M) refer to the main paper. Tables reproduced as Supplementary Materials only are indexed by a character string consisting of a capital letter followed by a number (sometimes by a number plus a lower case letter, e.g., F1a). In Section A and B of the Supplementary Materials, equations are assigned bracketed numbers [1], . . . , [12].

Appendix A: PROOFS

A1: Maximum-likelihood estimation procedure for the Dirichlet family of distributions

It suffices to describe the estimation procedure for a single sample $\mathbf{X}_1, \dots, \mathbf{X}_n$ from $\mathcal{D}(\boldsymbol{\alpha})$. Denoting the j -th component of the i -th random vector by X_{ij} and writing for brevity $G_j = \prod_{i=1}^n X_{ij}^{1/n}$ for each $j = 1, \dots, p$, the efficient score for α_k is given by

$$U_k(\boldsymbol{\alpha}) = n \left\{ \psi \left(\sum_{j=0}^p \alpha_j \right) - \psi(\alpha_k) + \ln(G_k) \right\}, \quad k = 0, \dots, p, \quad [1]$$

where $\psi(\cdot)$ stands for the digamma function, i.e., the log-derivative of $\Gamma(\cdot)$. The observed information matrix $\mathbf{I}(\boldsymbol{\alpha})$ has elements

$$I_{kl}(\boldsymbol{\alpha}) = \begin{cases} -\psi' \left(\sum_{j=0}^p \alpha_j \right), & k, l = 0, \dots, p, k \neq l \\ \psi'(\alpha_l) - \psi' \left(\sum_{j=0}^p \alpha_j \right), & k, l = 0, \dots, p, k = l. \end{cases} \quad [2]$$

Since these expressions contain only constants, $\mathbf{I}(\boldsymbol{\alpha})$ coincides with the expected information matrix denoted by $\mathcal{I}(\boldsymbol{\alpha})$. As proven below as a Lemma, its inverse admits the representation

$$\mathcal{I}^{-1}(\boldsymbol{\alpha}) = \text{diag} \left(\frac{1}{\psi'(\alpha_0)}, \dots, \frac{1}{\psi'(\alpha_p)} \right) + \frac{\psi' \left(\sum_{j=0}^p \alpha_j \right)}{1 - \psi' \left(\sum_{j=0}^p \alpha_j \right) \sum_{j=0}^p \frac{1}{\psi'(\alpha_j)}} \begin{pmatrix} \frac{1}{\psi'(\alpha_0)} \\ \vdots \\ \frac{1}{\psi'(\alpha_p)} \end{pmatrix} \begin{pmatrix} \frac{1}{\psi'(\alpha_0)}, \dots, \frac{1}{\psi'(\alpha_p)} \end{pmatrix}. \quad [3]$$

Multiplying Equation [3] with $1/n$ yields the expression displayed in Kotz *and others* (2000) as Equation (49.32), provided an obvious typing error contained in the latter (omission of the factor $1/n$ from the formula for $\hat{\boldsymbol{\alpha}}^{*'}$) is corrected. From well known properties of multiparameter exponential families of distributions (see, e.g., van der Vaart, 2000, Ch. 4.2), it follows that the system of likelihood equations corresponding to [1] has almost surely a unique solution $\hat{\boldsymbol{\alpha}}_n$, say, such that there holds

$$\sqrt{n}(\hat{\boldsymbol{\alpha}}_n - \boldsymbol{\alpha}) \xrightarrow{d} \mathcal{N}(\mathbf{0}, \mathcal{I}^{-1}(\boldsymbol{\alpha})) \quad \text{as } n \rightarrow \infty \quad [4]$$

for every $\boldsymbol{\alpha} \in (0, \infty)^{p+1}$. The ML estimator $\hat{\boldsymbol{\alpha}}_n$ has to be determined numerically by means of the Newton-Raphson algorithm. The literature contains different proposals for specifying an initial value $\boldsymbol{\alpha}^{(0)}$ from which iteration has to be started. The SAS/IML module (see Appendix C below) we used in all calculations and simulations adopts the proposal of Ronning (1989) setting $\alpha_j^{(0)} = \min_{1 \leq i \leq n} X_{ij}$ for all $j = 0, \dots, p$.

The information matrix with (k, l) -element

$$\mathcal{I}_{kl}(\boldsymbol{\alpha}) = \begin{cases} -\psi' \left(\sum_{j=0}^p \alpha_j \right), & k, l = 0, \dots, p, k \neq l \\ \psi'(\alpha_l) - \psi' \left(\sum_{j=0}^p \alpha_j \right), & k, l = 0, \dots, p, k = l \end{cases}$$

can be written

$$\mathcal{I}(\boldsymbol{\alpha}) = \mathbf{D} + \beta \mathbf{a} \mathbf{b}',$$

with

$$\begin{aligned} \mathbf{D} &= -\text{diag}(\psi'(\alpha_0), \dots, \psi'(\alpha_p)), \\ \beta &= \psi' \left(\sum_{l=0}^p \alpha_l \right) \text{ and} \\ \mathbf{a} &= (\underbrace{1, \dots, 1}_{p+1})' = \mathbf{b}. \end{aligned}$$

Applying Lemma A.1 yields

$$\mathcal{I}^{-1}(\boldsymbol{\alpha}) = \mathbf{D}^{-1} + \gamma \mathbf{e} \mathbf{f}',$$

where

$$\begin{aligned} \mathbf{D}^{-1} &= -\text{diag} \left(\frac{1}{\psi'(\alpha_0)}, \dots, \frac{1}{\psi'(\alpha_p)} \right), \\ \gamma &= -\frac{\beta}{1 + \beta \sum_{l=0}^p \frac{a_l b_l}{d_l}} = \frac{\psi'(\sum_{l=0}^p \alpha_l)}{\psi'(\sum_{l=0}^p \alpha_l) \sum_{l=0}^p \frac{1}{\psi'(\alpha_l)} - 1}, \\ \mathbf{e} = \mathbf{D}^{-1} \mathbf{a} &= \begin{pmatrix} -\frac{1}{\psi'(\alpha_0)} \\ \vdots \\ -\frac{1}{\psi'(\alpha_p)} \end{pmatrix} = \mathbf{D}^{-1} \mathbf{b} = \mathbf{f}. \end{aligned} \quad \blacksquare$$

Lemma A.1 Let $\mathbf{C}$ denote an invertible real matrix of order $q \times q$ admitting the representation

$$\mathbf{C} = \mathbf{D} + \beta \mathbf{a} \mathbf{b}',$$

where $\mathbf{a}$ and $\mathbf{b}$ is a q -dimensional vector of real numbers, $\mathbf{D} = \text{diag}(d_1, \dots, d_q)$ a nonsingular diagonal $(q \times q)$ -matrix, and β a real number with

$$\beta \neq -\frac{1}{\sum_{j=1}^q \frac{a_j b_j}{d_j}}.$$

Then

$$\mathbf{C}^{-1} = \mathbf{D}^{-1} + \gamma \mathbf{e} \mathbf{f}',$$

where

$$\gamma = -\frac{\beta}{1 + \beta \sum_{j=1}^q \frac{a_j b_j}{d_j}}, \quad \mathbf{e} = \mathbf{D}^{-1} \mathbf{a}, \quad \mathbf{f} = \mathbf{D}^{-1} \mathbf{b}.$$

Proof. We can write:

$$\begin{aligned} \beta + \gamma + \beta \gamma \sum_{j=1}^q \frac{a_j b_j}{d_j} &= \beta - \frac{\beta}{1 + \beta \sum_{j=1}^q \frac{a_j b_j}{d_j}} - \frac{\beta^2}{1 + \beta \sum_{j=1}^q \frac{a_j b_j}{d_j}} \sum_{j=1}^q \frac{a_j b_j}{d_j} \\ &= \frac{\beta + \beta^2 \sum_{j=1}^q \frac{a_j b_j}{d_j} - \beta - \beta^2 \sum_{j=1}^q \frac{a_j b_j}{d_j}}{1 + \beta \sum_{j=1}^q \frac{a_j b_j}{d_j}} \quad (*) \\ &= 0. \end{aligned}$$

Exploiting this identity yields:

$$\begin{aligned} \mathbf{C} \mathbf{C}^{-1} &= (\mathbf{D} + \beta \mathbf{a} \mathbf{b}') (\mathbf{D}^{-1} + \gamma \mathbf{e} \mathbf{f}') \\ &= \mathbf{D} \mathbf{D}^{-1} + \beta \mathbf{a} \mathbf{b}' \mathbf{D}^{-1} + \mathbf{D} \gamma \mathbf{e} \mathbf{f}' + \beta \gamma \mathbf{a} \mathbf{b}' \mathbf{e} \mathbf{f}' \\ &= \mathbf{I} + \beta \mathbf{a} \mathbf{b}' \mathbf{D}^{-1} + \underbrace{\mathbf{D} \gamma \mathbf{D}^{-1}}_{=\gamma} \mathbf{a} \mathbf{b}' \mathbf{D}^{-1} + \beta \gamma \mathbf{a} \underbrace{\mathbf{b}' \mathbf{D}^{-1} \mathbf{a}}_{\sum_{j=1}^q \frac{a_j b_j}{d_j}} \mathbf{b}' \mathbf{D}^{-1} \\ &= \mathbf{I} + \underbrace{\left(\beta + \gamma + \beta \gamma \sum_{j=1}^q \frac{a_j b_j}{d_j} \right)}_{\stackrel{(*)}{=} 0} \mathbf{a} \mathbf{b}' \mathbf{D}^{-1} \\ &= \mathbf{I} \end{aligned}$$

■

A2: Derivation of the asymptotic variance in § 3.2.

If the function $g^{(k)} : \mathbb{R}_+^{2p+2} \rightarrow \mathbb{R}_+ \cup \{0\}$ is defined by

$$g^{(k)}(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2) = \tilde{\varrho}^{(k)} = \ln \alpha_{2k} + \ln \alpha_{10} - \ln \alpha_{20} - \ln \alpha_{1k},$$

it follows that

$$g^{(k)}(\hat{\boldsymbol{\alpha}}_1, \hat{\boldsymbol{\alpha}}_2) = T_N^{(k)} = \ln \hat{\alpha}_{2k} + \ln \hat{\alpha}_{10} - \ln \hat{\alpha}_{20} - \ln \hat{\alpha}_{1k}.$$

$g^{(k)}$ is totally differentiable. Together with Equation (11) the δ -method states that the following holds true:

$$\sqrt{N} \frac{(T_N^{(k)} - \tilde{\varrho}^{(k)})}{\sqrt{v^{(k)}}} \xrightarrow{d} \mathcal{N}(0, 1) \quad \text{as } N \rightarrow \infty,$$

where the asymptotic variance $v^{(k)}$ has to be calculated with $(\gamma_{10}^{(k)}, \dots, \gamma_{1p}^{(k)})' = \boldsymbol{\gamma}_1^{(\mathbf{k})} \in \mathbb{R}^{p+1}$ and $(\gamma_{20}^{(k)}, \dots, \gamma_{2p}^{(k)})' = \boldsymbol{\gamma}_2^{(\mathbf{k})} \in \mathbb{R}^{p+1}$ as defined by

$$\gamma_{1j}^{(k)} := \frac{\partial g^{(k)}(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)}{\partial \alpha_{1j}} = \frac{\partial \tilde{\varrho}^{(k)}}{\partial \alpha_{1j}} = \begin{cases} \frac{1}{\alpha_{10}} & , \text{ if } j = 0, \\ -\frac{1}{\alpha_{1k}} & , \text{ if } j = k, \\ 0 & , \text{ otherwise,} \end{cases}$$

and

$$\gamma_{2j}^{(k)} := \frac{\partial g^{(k)}(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)}{\partial \alpha_{2j}} = \frac{\partial \tilde{\varrho}^{(k)}}{\partial \alpha_{2j}} = \begin{cases} -\frac{1}{\alpha_{20}} & , \text{ if } j = 0, \\ \frac{1}{\alpha_{2k}} & , \text{ if } j = k, \\ 0 & , \text{ otherwise.} \end{cases}$$

With these partial derivatives the asymptotic variance is given by

$$v^{(k)} = \left(\boldsymbol{\gamma}_1^{(\mathbf{k})} \right)' \frac{\boldsymbol{\Sigma}_1}{c} \boldsymbol{\gamma}_1^{(\mathbf{k})} + \left(\boldsymbol{\gamma}_2^{(\mathbf{k})} \right)' \frac{\boldsymbol{\Sigma}_2}{1-c} \boldsymbol{\gamma}_2^{(\mathbf{k})},$$

with

$$\boldsymbol{\Sigma}_\nu = \mathcal{I}^{-1}(\boldsymbol{\alpha}_\nu), \quad \gamma_{\nu j}^{(k)} = \begin{cases} \frac{(-1)^{\nu-1}}{\alpha_{\nu 0}} & , \text{ if } j = 0, \\ \frac{(-1)^\nu}{\alpha_{\nu k}} & , \text{ if } j = k, \\ 0 & , \text{ otherwise,} \end{cases}$$

for $\nu = 1, 2$. ■

A3: Derivation of the expression for the asymptotic variance in § 3.3.

Let $g : \mathbb{R}_+^{2p+2} \rightarrow \mathbb{R}_+ \cup \{0\}$ be defined by

$$g(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2) = \delta^2 = \sum_{k=1}^p \ln^2(\varrho_k) = \sum_{k=1}^p [\ln \alpha_{2k} + \ln \alpha_{10} - \ln \alpha_{20} - \ln \alpha_{1k}]^2.$$

The partial derivatives of g are

$$\zeta_{1j} := \frac{\partial g(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)}{\partial \alpha_{1j}} = \begin{cases} \frac{1}{\alpha_{10}} \sum_{k=1}^p 2 \ln(\varrho_k) & , \text{ if } j = 0, \\ -\frac{2 \ln(\varrho_j)}{\alpha_{1j}} & , \text{ if } j \neq 0, \end{cases}$$

and

$$\zeta_{2j} := \frac{\partial g(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)}{\partial \alpha_{2j}} = \begin{cases} -\frac{1}{\alpha_{20}} \sum_{k=1}^p 2 \ln(\varrho_k) & , \text{ if } j = 0, \\ \frac{2 \ln(\varrho_j)}{\alpha_{2j}} & , \text{ if } j \neq 0. \end{cases}$$

According to Equation (11) and the δ -method the asymptotic variance appearing in Equation (22) is

$$\begin{aligned} \tau^2(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2) &= \sum_{j=0}^p \sum_{k=0}^p \frac{\sigma_{jk}^{(1)}}{c} \frac{\partial g(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)}{\partial \alpha_{1j}} \frac{\partial g(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)}{\partial \alpha_{1k}} \\ &\quad + \sum_{j=0}^p \sum_{k=0}^p \frac{\sigma_{jk}^{(2)}}{1-c} \frac{\partial g(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)}{\partial \alpha_{2j}} \frac{\partial g(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2)}{\partial \alpha_{2k}} \\ &= \zeta_1' \frac{\boldsymbol{\Sigma}_1}{c} \zeta_1 + \zeta_2' \frac{\boldsymbol{\Sigma}_2}{1-c} \zeta_2. \end{aligned}$$

■

A4: Inversion of the Obenchain transformation.

For $(u_0, \dots, u_p) \in \mathbb{R}^{p+1}$ with $u_k \geq 0 \forall k = 0, \dots, p$ and $\sum_{k=0}^p u_k = 1$ define the transformation

$$(v_1, \dots, v_p)' = \mathbf{g}((u_0, \dots, u_p)') = (\ln(u_1/u_0), \dots, \ln(u_p/u_0))'.$$

Then, it follows that

$$(u_1/u_0, \dots, u_p/u_0)' = (\exp(v_1), \dots, \exp(v_p)) \quad \text{and} \quad u_0 = 1 - \sum_{k=1}^p u_k. \quad (*)$$

Equation $(*)$ states that for $k, l \in \{1, \dots, p\}$, it holds true:

$$\frac{u_k}{u_l} = \exp(v_k - v_l) \implies u_k = \exp(v_k - v_l) u_l. \quad (**)$$

With equations $(*)$ and $(**)$, we get for $l \in \{1, \dots, p\}$

$$\begin{aligned} u_l &= \exp(v_l) \underbrace{\left(1 - \sum_{k=1}^p u_k\right)}_{u_0} \\ &\iff \\ u_l + u_l \exp(v_l) &= \exp(v_l) - \exp(v_l) \sum_{\substack{k=1 \\ k \neq l}}^p u_k \\ &\stackrel{(**)}{=} \exp(v_l) - \exp(v_l) u_l \sum_{\substack{k=1 \\ k \neq l}}^p \exp(v_k - v_l) \\ &= \exp(v_l) - u_l \sum_{\substack{k=1 \\ k \neq l}}^p \exp(v_k) \\ &\iff \\ u_l \left(1 + \exp(v_l) + \sum_{\substack{k=1 \\ k \neq l}}^p \exp(v_k)\right) &= \exp(v_l) \\ &\iff \\ u_l &= \frac{\exp(v_l)}{1 + \sum_{k=1}^p \exp(v_k)}. \end{aligned}$$

Finally, we have

$$\begin{aligned}
u_0 &= 1 - \sum_{l=1}^p u_l \\
&= 1 - \sum_{l=1}^p \frac{\exp(v_l)}{1 + \sum_{k=1}^p \exp(v_k)} \\
&= \frac{1}{1 + \sum_{k=1}^p \exp(v_k)} .
\end{aligned}$$

■

A5: Derivation of the asymptotic variance in § 5.3.

The aggregate distance measure in § 5.3 is

$$\delta^2(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2) = \sum_{k=1}^p \ln^2(\varrho_k) = \sum_{k=1}^p (\ln \mu_{2k} + \ln \mu_{10} - \ln \mu_{20} - \ln \mu_{1k})^2 .$$

The partial derivatives are

$$\Delta_{1j} := \frac{\partial \delta^2(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)}{\partial \mu_{1j}} = \begin{cases} \frac{1}{\mu_{10}} \sum_{k=1}^p 2 \ln(\varrho_k) & , \text{ if } j = 0, \\ -\frac{2 \ln(\varrho_j)}{\mu_{1j}} & , \text{ if } j \neq 0, \end{cases}$$

and

$$\Delta_{2j} := \frac{\partial \delta^2(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)}{\partial \mu_{2j}} = \begin{cases} -\frac{1}{\mu_{20}} \sum_{k=1}^p 2 \ln(\varrho_k) & , \text{ if } j = 0, \\ \frac{2 \ln(\varrho_j)}{\mu_{2j}} & , \text{ if } j \neq 0 . \end{cases}$$

With these derivatives, the δ -method yields for the asymptotic variance appearing in Equation (48) the expression

$$\begin{aligned}
w &= \sum_{j=0}^p \sum_{k=0}^p \frac{\sigma_{jk}^{(1)}}{c} \frac{\partial \delta^2(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)}{\partial \mu_{1j}} \frac{\partial \delta^2(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)}{\partial \mu_{1k}} \\
&\quad + \sum_{j=0}^p \sum_{k=0}^p \frac{\sigma_{jk}^{(2)}}{1-c} \frac{\partial \delta^2(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)}{\partial \mu_{2j}} \frac{\partial \delta^2(\boldsymbol{\mu}_1, \boldsymbol{\mu}_2)}{\partial \mu_{2k}} \\
&= \Delta'_1 \frac{\Sigma_1}{c} \Delta_1 + \Delta'_2 \frac{\Sigma_2}{1-c} \Delta_2 .
\end{aligned}$$

■

Appendix B: DERIVATION OF THE TESTS FOR THE OBENCHAIN MODEL

B1: Component-wise tests for equivalence of two Obenchain distributions

For univariate normally distributed data the $(1 - \alpha)$ -confidence bounds for the non-standardized differences of the expected values $\check{\delta}_k = \check{\mu}_{2k} - \check{\mu}_{1k}$ estimated by $\check{T}_N^{(k)} = \check{Y}_k - \check{X}_k$ are well known to be given by

$$C_{1-\alpha}(\check{\delta}_k) = \check{T}_N^{(k)} - t_{N-2;1-\alpha} \sqrt{\frac{N}{RT}} \check{S}^{(k)}, \quad \bar{C}_{1-\alpha}(\check{\delta}_k) = \check{T}_N^{(k)} + t_{N-2;1-\alpha} \sqrt{\frac{N}{RT}} \check{S}^{(k)}, \quad [5]$$

where $\check{\mathbf{X}}$ and $\check{\mathbf{Y}}$ are the respective mean values of the transformed samples $\check{\mathbf{X}}$ and $\check{\mathbf{Y}}$, $\check{S}^{(k)}$ is the pooled empirical standard deviation in component k , and $t_{N-2;1-\alpha}$ denotes the $(1 - \alpha)$ -quantile of the central t-distribution with $N - 2$ degrees of freedom. Hence, the critical function of a level- α confidence interval inclusion test for the k th elementary equivalence problem is given by

$$\check{\phi}^{(k)}(\mathbf{X}, \mathbf{Y}) = \begin{cases} 1 & , \text{ if } \left(C_{1-\alpha}(\check{\delta}_k), \bar{C}_{1-\alpha}(\check{\delta}_k) \right) \subset \left(-\varepsilon_1^{(k)}, \varepsilon_2^{(k)} \right) \\ 0 & , \text{ if } \left(C_{1-\alpha}(\check{\delta}_k), \bar{C}_{1-\alpha}(\check{\delta}_k) \right) \not\subset \left(-\varepsilon_1^{(k)}, \varepsilon_2^{(k)} \right) \end{cases}. \quad [6]$$

Using these tests as building blocks of an intersection-union test for the combined equivalence problem (5) yields the critical function

$$\check{\phi}(\mathbf{X}, \mathbf{Y}) = \prod_{k=1}^p \check{\phi}^{(k)}(\mathbf{X}, \mathbf{Y}). \quad [7]$$

B2: Aggregate test for equivalence of two Obenchain distributions

Under the homoskedasticity assumption $\check{\sigma}_{kk}^{(1)} = \check{\sigma}_{kk}^{(2)}$, (30) and (6) imply that the Obenchain version of the aggregate equivalence testing problem (7) reads

$$H_0 : \sum_{k=1}^p \left(\check{\mu}_2^{(k)} - \check{\mu}_1^{(k)} \right)^2 \geq \check{\delta}_0^2 \quad \text{versus} \quad H_1 : \sum_{k=1}^p \left(\check{\mu}_2^{(k)} - \check{\mu}_1^{(k)} \right)^2 < \check{\delta}_0^2. \quad [8]$$

In Hoffelder & Wellek (2015) it is shown that for multivariate normally distributed data $(\check{\mathbf{X}}, \check{\mathbf{Y}})$ setting

$$\hat{\delta}_N^2 = \sum_{k=1}^p \left(\check{Y}^{(k)} - \check{X}^{(k)} \right)^2 \quad [9]$$

yields a test statistic for (6) whose asymptotic distribution is given by

$$\frac{\sqrt{N} \left(\hat{\delta}_N^2 - \check{\delta}_0^2 \right)}{\sqrt{\hat{v}}} \xrightarrow{d} \mathcal{N}(0, 1), \quad [10]$$

where

$$\hat{v} = 4N \left(\check{\mathbf{Y}} - \check{\mathbf{X}} \right)' \left(\frac{\check{\mathbf{S}}_1}{R} + \frac{\check{\mathbf{S}}_2}{T} \right) \left(\check{\mathbf{Y}} - \check{\mathbf{X}} \right), \quad [11]$$

and $\check{\mathbf{S}}_1$ and $\check{\mathbf{S}}_2$ are the empirical covariance matrices of the respective samples $\check{\mathbf{X}}$ and $\check{\mathbf{Y}}$. From this result it follows that

$$\check{\psi}(\mathbf{X}, \mathbf{Y}) = \begin{cases} 1 & \text{for } \frac{\sqrt{N}(\hat{\delta}_N^2 - \check{\delta}_0^2)}{\sqrt{\hat{v}}} < u_\alpha, \\ 0 & \frac{\sqrt{N}(\hat{\delta}_N^2 - \check{\delta}_0^2)}{\sqrt{\hat{v}}} \geq u_\alpha \end{cases} \quad [12]$$

is the critical function of an asymptotically valid level- α test for aggregate equivalence of two Obenchain distributions with homoskedastic Gaussian distributions of the transformed observations.

Appendix C: PROGRAM CODE

C1: SAS/IML auxiliary modules

In this section, auxiliary modules are presented which are used as building blocks for the implementations of the equivalence testing procedures in Appendix C2.

With the SAS/IML-function `ESTIMATE_PARAMETER` the Dirichlet parameter is estimated from a sample, given by the matrix `X`, according to the Newton-Raphson algorithm described in §3.1. `max_iter` is the maximum number of iteration steps in the Newton-Raphson algorithm. Assigning to the variable `startestimator` an initial value which is close to the exact value of the estimate considerably reduces the execution time of the iteration procedure.

```
1  START  ESTIMATE_PARAMETER( X , max_iter , startestimator);
2
3  eps_alpha          = 0.01 ;
4  p                  = NCOL(X) ;
5  n                  = NROW(X) ;
6
7  * start vector ;
8  IF startestimator[1] = 0 THEN DO ; * no start vector ;
9    ronning_alpha0    = J(p,1,0) ;
10   DO i=1 TO p ;
11     min              = MIN( X[,i] ) ;
12     IF ( min = 0 ) THEN DO ;
13       print 'Error: Value is 0' ;
14     END ;
15   ELSE DO ; *min > 0;
16     IF ( min < 10##(-10) ) THEN DO ; * eps_data = 10##(-10) ;
17       ronning_alpha0[i] = 10##(-10) ;
18     END ;
19   ELSE DO ;
20     ronning_alpha0[i] = min ;
21   END ;
22 END ;
23 END;
24 alpha              = ronning_alpha0 ;
25 END; * no start vector ;
26 ELSE DO ;
27   alpha              = startestimator ;
28 END;
29
30 ln_Gj              = J(1,p,0) ;
31 IF ANY( X < 0 ) THEN PRINT 'Error: Negative Value in X: ' X ;
32 LOGMATRIX          = LOG(X) ;
33 DO j=1 TO p ;
34   ln_Gj[j]          = LOGMATRIX[+,j] / n ;
35 END;
36
37 paramsequencelength = max_iter + 1 ;
38 Paramsequence       = J(paramsequencelength , p , 0) ;
39 Paramsequence[1,]   = alpha ;
```

```

40  f_of_alpha          = J(p,1,0) ;
41  diag_values         = J(p,1,0) ;
42  trigamma_inv_values = J(p,1,0) ;
43
44  * Iteration loop for calculating parameter estimate ;
45  iterationsteps      = 0 ;
46  convergence_statistic= 1 ;
47  DO UNTIL( iterationsteps = max_iter | convergence_statistic < 10##(-10) ) ;
48    iterationsteps    = iterationsteps + 1 ;
49    sum_of_alphas     = alpha[+] ;
50    DO j=1 TO p ;
51      f_of_alpha[j]   = DIGAMMA( sum_of_alphas ) - DIGAMMA( alpha[j] )
52                      + ln_Gj[j] ;
53    END;
54
55    DO j=1 TO p ;
56      trigamma_inv_values[j] = 1 / ( TRIGAMMA( alpha[j] ) ) ;
57    END;
58    D_inv              = -DIAG( trigamma_inv_values ) ;
59
60    gamma              = TRIGAMMA( sum_of_alphas )
61                      / ( TRIGAMMA( sum_of_alphas )
62                        * ( trigamma_inv_values[+] ) - 1 ) ;
63
64    e_and_f            = trigamma_inv_values ;
65    M                  = D_inv + gamma * e_and_f * e_and_f' ;
66
67    alpha_new          = alpha - M * f_of_alpha ;
68    DO j=1 TO p ;
69      IF ( alpha_new[j] < 0 ) THEN alpha_new[j] = eps_alpha ;
70    END;
71    IF ( MIN(alpha_new) > eps_alpha )
72      THEN convergence_statistic = f_of_alpha' * (-M) * f_of_alpha ;
73    Paramsequence[iterationsteps+1,] = alpha_new' ;
74    alpha                = alpha_new ;
75
76  END ; * DO UNTIL ;
77
78  IF ( iterationsteps < max_iter )
79    THEN Paramsequence = Paramsequence[1:(iterationsteps+2),] ;
80    ELSE PRINT 'estimation process not succesful' ;
81  IF ( MIN(alpha) < 0 ) THEN DO;
82    PRINT X , alpha iterationsteps Paramsequence ;
83  END;
84  return (alpha');
85
86  FINISH ESTIMATE_PARAMETER ;

```

The function `CALC_EMPIRICAL_COV_MAT` calculates the empirical covariance matrix from a sample given by the IML matrix `DATA`.

```

1  START CALC_EMPIRICAL_COV_MAT( DATA );
2
3  p          = NCOL( DATA ) ;

```

```

4   n           = NROW( DATA ) ;
5   Mean        = DATA[:,] ;
6   MEAN_MATRIX = SHAPE( Mean , n , p ) ;
7   S           = 1 / ( n - 1 ) * (DATA - MEAN_MATRIX)' * (DATA - MEAN_MATRIX) ;
8   return (S);
9
10  FINISH CALC_EMPIRICAL_COV_MAT;

```

The function `OBENCHAIN_TRANSFORMATION` calculates the Obenchain transformed sample $\check{\mathbf{X}}$ from an APSD sample $\mathbf{X}$, given by the IML matrix `X`, according to the Obenchain transformation in §4.1.

```

1  START OBENCHAIN_TRANSFORMATION( X );
2
3  p           = NCOL( X ) - 1 ;
4  n           = NROW( X ) ;
5  X_BREVE     = J( n , p , 0 ) ;
6  do i = 1 to n ;
7    do j = 1 to p ;
8      X_BREVE[i,j] = LOG(X[i,j+1] / X[i,1]) ;
9    end;
10 end;
11 return (X_BREVE);
12
13 FINISH OBENCHAIN_TRANSFORMATION;

```

C2: SAS/IML modules for all derived equivalence tests

The function `IU_D_TEST` applies the IU-D test for equivalence of §3.2 to the samples `X` and `Y`. The vectors `epsilon_1` and `epsilon_2` contain the lower and upper equivalence margins for all elementary tests making up the IU-D procedure. `alpha` is the nominal significance level. The parameters `max_iter` and `startestimator` are the parameters of the IML function `ESTIMATE_PARAMETER` explained in Appendix C1.

```

1  START IU_D_TEST( X , Y , epsilon_1 , epsilon_2 , alpha
2                  , max_iter , startestimator);
3
4  Alpha_1_hat = ESTIMATE_PARAMETER( X , max_iter , startestimator )';
5  Alpha_2_hat = ESTIMATE_PARAMETER( Y , max_iter , startestimator )';
6  sum_alpha_1_hat = Alpha_1_hat[+] ;
7  sum_alpha_2_hat = Alpha_2_hat[+] ;
8  p = ncol(X) - 1 ;
9  R = nrow(X) ;
10 T = nrow(Y) ;
11 N = R + T ;
12 Vector_inv_trigamma_1 = J( p + 1 , 1 , 0 ) ;
13 Vector_inv_trigamma_2 = J( p + 1 , 1 , 0 ) ;
14 DO k = 1 TO p + 1 ;
15   Vector_inv_trigamma_1[k] = 1 / TRIGAMMA( Alpha_1_hat[k] ) ;
16   Vector_inv_trigamma_2[k] = 1 / TRIGAMMA( Alpha_2_hat[k] ) ;
17 END ;
18 sum_inv_trigammas_1 = Vector_inv_trigamma_1[+] ;

```

```

19  sum_inv_trigammas_2  = Vector_inv_trigamma_2[+] ;
20
21  * EMPIRICAL COVARIANCEMATRIX FIRST SAMPLE ;
22  D_INV_1      = DIAG(Vector_inv_trigamma_1) ;
23  gamma_1      = TRIGAMMA( sum_alpha_1_hat )
24                / ( 1 - TRIGAMMA( sum_alpha_1_hat ) * sum_inv_trigammas_1 ) ;
25  d_1_hat      = Vector_inv_trigamma_1 ;
26  SIGMA_HAT_1  = D_INV_1 + gamma_1 * d_1_hat * d_1_hat' ;
27
28  * EMPIRICAL COVARIANCEMATRIX SECOND SAMPLE ;
29  D_INV_2      = DIAG(Vector_inv_trigamma_2) ;
30  gamma_2      = TRIGAMMA( sum_alpha_2_hat )
31                / ( 1 - TRIGAMMA( sum_alpha_2_hat ) * sum_inv_trigammas_2 ) ;
32  d_2_hat      = Vector_inv_trigamma_2 ;
33  SIGMA_HAT_2  = D_INV_2 + gamma_2 * d_2_hat * d_2_hat' ;
34
35  * pseudo odds ratios ;
36  p_1_hat      = Alpha_1_hat / sum_alpha_1_hat ;
37  p_2_hat      = Alpha_2_hat / sum_alpha_2_hat ;
38  pi_hat       = J(p,1,0) ;
39  DO k = 1 TO p ;
40    pi_hat[k]   = p_2_hat[k+1] * p_1_hat[1]
41                  / ( p_2_hat[1] * p_1_hat[k+1] ) ;
42  END;
43
44  * TESTSTATISTIC ;
45  T_N          = LOG( pi_hat ) ;
46
47  * asymptotic variances ;
48  GAMMAS_1_HAT = J(p+1,p,0) ;
49  DO k = 1 TO p ;
50    DO j = 1 to p+1;
51      IF j = 1 THEN GAMMAS_1_HAT[j,k] = 1 / Alpha_1_hat[1] ;
52      IF j = k+1 THEN GAMMAS_1_HAT[j,k] = -1 / Alpha_1_hat[k+1] ;
53    END;
54  END;
55  GAMMAS_2_HAT = J(p+1,p,0) ;
56  DO k = 1 TO p ;
57    DO j = 1 to p+1;
58      IF j = 1 THEN GAMMAS_2_HAT[j,k] = -1 / Alpha_2_hat[1] ;
59      IF j = k+1 THEN GAMMAS_2_HAT[j,k] = 1 / Alpha_2_hat[k+1] ;
60    END;
61  END;
62  v_hat       = J(p,1,0) ;
63  DO k = 1 TO p ;
64    gamm_hat_k_1 = GAMMAS_1_HAT[,k] ;
65    gamm_hat_k_2 = GAMMAS_2_HAT[,k] ;
66    v_hat[k]     = N * ( gamm_hat_k_1' * SIGMA_HAT_1 / R * gamm_hat_k_1
67                        + gamm_hat_k_2' * SIGMA_HAT_2 / T * gamm_hat_k_2
68                        ) ;
69  END;
70
71  *CALCULATION TESTRESULT ;

```

```

72  quantil          = PROBIT(1-alpha) ;
73  Element_tests    = J(p,1,-1) ;
74  LCLs_PORD        = J(p,1,-1) ;
75  UCLs_PORD        = J(p,1,-1) ;
76  DO k = 1 TO p ;
77    LCLs_PORD[k]    = T_N[k] - SQRT( v_hat[k] / N ) * quantil ;
78    UCLs_PORD[k]    = T_N[k] + SQRT( v_hat[k] / N ) * quantil ;
79    Element_tests[k] = ( ( epsilon_1[k] < LCLs_PORD[k] )
80                      & ( UCLs_PORD[k] < epsilon_2[k] ) ) ;
81  END;
82  IF ( Element_tests > 0.5 ) THEN result = 1;
83                      ELSE result = 0;
84  return (result);
85  FINISH IU_D_TEST ;

```

The function IU_0_TEST applies the IU-O test for equivalence of §4.2 to the samples X and Y. The vectors `epsilon_1` and `epsilon_2` contain the lower and upper equivalence margins for all elementary tests of the IU-O test. `alpha` is the nominal significance level.

```

1  START IU_0_TEST( X , Y , epsilon_1 , epsilon_2 , alpha );
2
3  X_BREVE          = OBENCHAIN_TRANSFORMATION( X ) ;
4  Y_BREVE          = OBENCHAIN_TRANSFORMATION( Y ) ;
5
6  p                = ncol(X_BREVE);
7  R                = nrow(X_BREVE);
8  T                = nrow(Y_BREVE);
9  N                = R + T ;
10 meanX_breve      = (X[:,])';
11 meanY_breve      = (Y[:,])';
12 T_N_breve        = meanY_breve - meanX_breve ;
13 S_1              = CALC_EMPIRICAL_COV_MAT( X_BREVE ) ;
14 S_2              = CALC_EMPIRICAL_COV_MAT( Y_BREVE ) ;
15 S                = 1/(N-2) * ( (R-1) * S_1 + (T-1) * S_2 ) ;
16
17 width_ci         = J(p,1,1);
18 do j = 1 to p;
19   width_ci[j] = tinv(1-alpha,N-2) * sqrt( N / (R*T) * S[j,j] );
20 end;
21 UCL_NSIU         = T_N_breve + width_ci ; -SP
22 LCL_NSIU         = T_N_breve - width_ci ;
23
24 Element_tests= J(p,1,-1) ;
25 DO k = 1 TO p ;
26   Element_tests[k]
27   = ( ( epsilon_1[k] < LCL_NSIU[k] ) & ( UCL_NSIU[k] < epsilon_2[k] )
28       ) ;
29 END;
30 IF ( Element_tests > 0.5 ) THEN result = 1 ;
31                      ELSE result = 0 ;
32 return (result);
33
34 FINISH IU_0_TEST ;

```

The function IU_SP_TEST applies the intersection-union test of §5.2 to the samples X and Y . The vectors `epsilon_1` and `epsilon_2` contain the lower and upper equivalence margins for all elementary tests of the IU-SP procedure. `alpha` is the nominal significance level.

```

1  START IU_SP_TEST( X , Y , epsilon_1 , epsilon_2 , alpha );
2
3  p          = ncol(X) - 1 ;
4  R          = nrow(X) ;
5  T          = nrow(Y) ;
6  N          = R+T ;
7  Mean_X     = (X[:,])' ;
8  Mean_Y     = (Y[:,])' ;
9  S_1        = CALC_EMPIRICAL_COV_MAT( X ) ;
10 S_2        = CALC_EMPIRICAL_COV_MAT( Y ) ;
11
12 *Pseudo Odds Ratios ;
13 rho_hat     = J(p,1,0) ;
14 DO k = 1 TO p ;
15   rho_hat[k] = Mean_Y[k+1] * Mean_X[1] / ( Mean_Y[1] * Mean_X[k+1] ) ;
16 END ;
17
18 *CALCULATION TESTSTATISTIC ;
19 T_N         = LOG( rho_hat ) ;
20
21 *CALCULATION asymptotic Variance ;
22 ETA_1       = J(p+1,p,0) ;
23 DO k = 1 TO p ;
24   DO j = 1 to p+1 ;
25     IF j = 1 THEN ETA_1[j,k] = 1 / Mean_X[1] ;
26     IF j = k+1 THEN ETA_1[j,k] = -1 / Mean_X[k+1] ;
27   END;
28 END;
29
30 ETA_2       = J(p+1,p,0) ;
31 DO k = 1 TO p ;
32   DO j = 1 to p+1 ;
33     IF j = 1 THEN ETA_2[j,k] = -1 / Mean_Y[1] ;
34     IF j = k+1 THEN ETA_2[j,k] = 1 / Mean_Y[k+1] ;
35   END;
36 END;
37
38 w          = J(p,1,0) ;
39 DO k = 1 TO p ;
40   eta_k_1   = ETA_1[,k] ;
41   eta_k_2   = ETA_2[,k] ;
42   w[k]      = N * ( eta_k_1' * S_1 / R * eta_k_1
43                   + eta_k_2' * S_2 / T * eta_k_2 ) ;
44 END;
45
46 *CALCULATION TESTRESULT ;
47 quantil     = PROBIT(1-alpha) ;
48 Element_tests= J(p,1,-1) ;
49 LCLs        = J(p,1,-1) ;

```



```

50   UCLs          = J(p,1,-1) ;
51   DO k = 1 TO p ;
52     LCLs[k]      = T_N[k] - SQRT( w[k] / N ) * quantil ;
53     UCLs[k]      = T_N[k] + SQRT( w[k] / N ) * quantil ;
54     Element_tests[k]
55     = ( ( epsilon_1[k] < LCLs[k] ) & ( UCLs[k] < epsilon_2[k] )
56         ) ;
57   END;
58   IF ( Element_tests > 0.5 ) THEN result = 1 ;
59                                   ELSE result = 0 ;
60   return (result);
61
62   FINISH IU_SP_TEST ;

```

The function AG_D_TEST applies the aggregate test of §3.3 to the samples X and Y. $\delta_{0,2}$ is the equivalence margin, α is the nominal significance level, and $\max_iter$ and startestimator are again the parameters of the IML function ESTIMATE_PARAMETER.

```

1   START AG_D_TEST( X , Y , delta_0_2 , alpha , max_iter , startestimator);
2
3   Alpha_1_hat      = ESTIMATE_PARAMETER( X , max_iter , startestimator )';
4   Alpha_2_hat      = ESTIMATE_PARAMETER( Y , max_iter , startestimator )';
5   sum_alpha_1_hat   = Alpha_1_hat[+] ;
6   sum_alpha_2_hat   = Alpha_2_hat[+] ;
7   p                = ncol(X) - 1 ;
8   R                = nrow(X) ;
9   T                = nrow(Y) ;
10  N                = R + T ;
11  Vector_inv_trigamma_1 = J( p + 1 , 1 , 0 ) ;
12  Vector_inv_trigamma_2 = J( p + 1 , 1 , 0 ) ;
13  DO k = 1 TO p + 1 ;
14    Vector_inv_trigamma_1[k] = 1 / TRIGAMMA( Alpha_1_hat[k] ) ;
15    Vector_inv_trigamma_2[k] = 1 / TRIGAMMA( Alpha_2_hat[k] ) ;
16  END ;
17  sum_inv_trigammas_1 = Vector_inv_trigamma_1[+] ;
18  sum_inv_trigammas_2 = Vector_inv_trigamma_2[+] ;
19
20  * EMPIRICAL COVARIANCEMATRIX FIRST SAMPLE ;
21  D_INV_1      = DIAG(Vector_inv_trigamma_1) ;
22  gamma_1      = TRIGAMMA( sum_alpha_1_hat )
23                / ( 1 - TRIGAMMA( sum_alpha_1_hat ) * sum_inv_trigammas_1 ) ;
24  d_1_hat      = Vector_inv_trigamma_1 ;
25  SIGMA_HAT_1 = D_INV_1 + gamma_1 * d_1_hat * d_1_hat' ;
26
27  * EMPIRICAL COVARIANCEMATRIX SECOND SAMPLE ;
28  D_INV_2      = DIAG(Vector_inv_trigamma_2) ;
29  gamma_2      = TRIGAMMA( sum_alpha_2_hat )
30                / ( 1 - TRIGAMMA( sum_alpha_2_hat ) * sum_inv_trigammas_2 ) ;
31  d_2_hat      = Vector_inv_trigamma_2 ;
32  SIGMA_HAT_2 = D_INV_2 + gamma_2 * d_2_hat * d_2_hat' ;
33
34  *Proportions and pseudo odds ratios ;
35  p_1_hat      = Alpha_1_hat / sum_alpha_1_hat ;
36  p_2_hat      = Alpha_2_hat / sum_alpha_2_hat ;

```

```

37  pi_hat      = J(p,1,0) ;
38  DO k = 1 TO p ;
39    pi_hat[k]  = p_2_hat[k+1] * p_1_hat[1] / ( p_2_hat[1] * p_1_hat[k+1] ) ;
40  END;
41
42  * TESTSTATISTIC ;
43  hat_delta_0_2  = SSQ( LOG( pi_hat ) ) ;
44
45  * asymptotic variance ;
46  zeta_1         = J(p+1,1,0) ;
47  zeta_1[1]      = 2 / Alpha_1_hat[1] * LOG( pi_hat ) [+] ;
48  DO j = 2 TO p+1 ;
49    zeta_1[j]    = -2 / Alpha_1_hat[j] * LOG( pi_hat[j-1] ) ;
50  END;
51  zeta_2         = J(p+1,1,0) ;
52  zeta_2[1]      = -2 / Alpha_2_hat[1] * LOG( pi_hat ) [+] ;
53  DO j = 2 TO p+1 ;
54    zeta_2[j]    = 2 / Alpha_2_hat[j] * LOG( pi_hat[j-1] ) ;
55  END;
56  tau_hat       = N * ( zeta_1' * SIGMA_HAT_1 / R * zeta_1
57                      + zeta_2' * SIGMA_HAT_2 / T * zeta_2
58                      ) ;
59
60  *CALCULATION TESTRESULT ;
61  frac          = SQRT(N) * ( hat_delta_0_2 - delta_0_2 ) / SQRT(tau_hat) ;
62  quantil       = PROBIT(alpha) ;
63  IF ( frac < quantil ) THEN result = 1;
64                      ELSE result = 0;
65  return (result);
66
67  FINISH AG_D_TEST;

```

The function `EDNE_TEST` applies the EDNE test for equivalence for normally distributed data according to Hoffelder et al. (2015) (see Equations [8] to [11]) to the samples `X` and `Y`. `deltaq_edne` is the equivalence margin and `alpha` is the nominal significance level. The function `EDNE_TEST` is used in the implementation of the aggregate test for samples for Obenchain distributions ($\rightarrow$ **B2**).

```

1  START EDNE_TEST( X , Y , deltaq_edne , alpha ) ;
2
3  R      = nrow(X) ;
4  T      = nrow(Y) ;
5  N      = R+T ;
6  Mean_X = (X[:,])';
7  Mean_Y = (Y[:,])';
8  Mean_Diff = Mean_X - Mean_Y ;
9  S_1     = CALC_EMPIRICAL_COV_MAT( X ) ;
10 S_2     = CALC_EMPIRICAL_COV_MAT( Y ) ;
11 Q_EDNE  = Mean_Diff' * Mean_Diff ;
12 v_hat   = 4 * N * Mean_Diff' * ( S_1/R + S_2/T ) * Mean_Diff ;
13 frac    = SQRT(N) * ( Q_EDNE - deltaq_edne ) / SQRT(v_hat) ;
14 quantil = PROBIT(alpha) ;
15 IF      ( frac < quantil ) THEN result = 1;
16                      ELSE result = 0;

```

```

17   return (result);
18
19   FINISH EDNE_TEST ;

```

The function AG_O_TEST applies the AG-O test of §4.3 to the samples X and Y. breve_delta_0_2 is the equivalence margin and alpha is the nominal significance level.

```

1   START AG_O_TEST( X , Y , breve_delta_0_2 , alpha ) ;
2
3   X_BREVE          = OBENCHAIN_TRANSFORMATION( X ) ;
4   Y_BREVE          = OBENCHAIN_TRANSFORMATION( Y ) ;
5   result           = EDNE_TEST( X_BREVE , Y_BREVE , breve_delta_0_2 , alpha ) ;
6   return (result);
7
8   FINISH AG_O_TEST ;

```

The function AG_SP_TEST implements the asymptotically distribution-free aggregate test of §5.3 with the samples X and Y, where delta_0_2 is the equivalence margin and alpha is the nominal significance level.

```

1   START AG_SP_TEST( X , Y , delta_0_2 , alpha );
2
3   p                = ncol(X) - 1 ;
4   R                = nrow(X) ;
5   T                = nrow(Y) ;
6   N                = R+T ;
7   Mean_X           = (X[:,])';
8   Mean_Y           = (Y[:,])';
9   S_1              = CALC_EMPIRICAL_COV_MAT( X ) ;
10  S_2               = CALC_EMPIRICAL_COV_MAT( Y ) ;
11
12  *pseudo odds ratios ;
13  pi_hat            = J(p,1,0) ;
14  DO k = 1 TO p ;
15    pi_hat[k]       = Mean_Y[k+1] * Mean_X[1] / ( Mean_Y[1] * Mean_X[k+1] ) ;
16  END;
17
18  *TESTSTATISTIC ;
19  delta_N_2         = SSQ( LOG( pi_hat ) ) ;
20
21  *asymptotic variance ;
22  Delta_1            = J(p+1,1,0) ;
23  Delta_1[1]         = 2 / Mean_X[1] * LOG( pi_hat ) [ + ] ;
24  DO j = 2 TO p+1 ;
25    Delta_1[j]       = - 2 * LOG( pi_hat[j-1] ) / Mean_X[j] ;
26  END;
27  Delta_2            = J(p+1,1,0) ;
28  Delta_2[1]         = - 2 / Mean_Y[1] * LOG( pi_hat ) [ + ] ;
29  DO j = 2 TO p+1 ;
30    Delta_2[j]       = 2 * LOG( pi_hat[j-1] ) / Mean_Y[j] ;
31  END;
32  w_hat             = N * ( Delta_1' * S_1 / R * Delta_1
33                        + Delta_2' * S_2 / T * Delta_2
34                        ) ;

```

```

35
36  *CALCULATION TESTRESULT ;
37  frac          = SQRT(N) * ( delta_N_2 - delta_0_2 ) / SQRT(w_hat) ;
38  quantil       = PROBIT(alpha) ;
39  IF ( frac < quantil ) THEN result = 1;
40                      ELSE result = 0;
41  return (result);
42
43  FINISH AG_SP_TEST ;

```

Appendix D: SIMULATION SCENARIOS, ADDITIONAL DETAILS

Table D1: Simulation scenarios used for studying the type-I error risk of the testing procedures under the Obenchain model.

		Obenchain Parameters of Reference and Test Distribution							
Component		Scenario 1		Scenario 2		Scenario 3		Scenario 4	
# (k)	Name	$\check{\mu}_{1k}$	$\check{\mu}_{2k}$	$\check{\mu}_{1k}$	$\check{\mu}_{2k}$	$\check{\mu}_{1k}$	$\check{\mu}_{2k}$	$\check{\mu}_{1k}$	$\check{\mu}_{2k}$
1	SIP	2.377	2.377	2.377	2.377	2.377	2.667	2.377	2.688
2	Stage 0	2.375	2.375	2.375	2.375	2.375	2.665	2.375	2.220
3	Stage 1	3.053	3.053	3.053	3.053	3.053	3.426	3.053	3.453
4	Stage 2	3.120	3.120	3.120	3.720	3.120	3.501	3.120	2.916
5	Stage 3	3.655	3.655	3.655	3.056	3.655	4.101	3.655	4.133
6	Stage 4	3.058	3.058	3.058	3.058	3.058	2.871	3.058	2.858
7	Stage 5	2.107	2.107	2.107	2.107	2.107	1.978	2.107	2.382
8	Stage 6	-0.259	0.340	-0.259	-0.259	-0.259	-0.243	-0.259	-0.242
9	Stage 7	-1.619	-2.219	-1.619	-1.619	-1.619	-1.521	-1.619	-1.831
10	Filter	1.193	1.193	1.193	1.193	1.193	1.121	1.193	1.115

Table D2: Componentwise equivalence margins used for studying the type-I error risk of the intersection-union tests under the Dirichlet model.

k	Scenario 1 $(-\varepsilon_1^{(k)}, \varepsilon_1^{(k)})$	Scenario 2 $(-\varepsilon_1^{(k)}, \varepsilon_1^{(k)})$	Scenario 3 $(-\varepsilon_1^{(k)}, \varepsilon_1^{(k)})$	Scenario 4 $(-\varepsilon_1^{(k)}, \varepsilon_1^{(k)})$
1	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.3789, 0.3789)$
2	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$
3	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.3789, 0.3789)$
4	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$
5	$(-0.8473, 0.8473)$	$(-0.6179, 0.6179)$	$(-0.2280, 0.2280)$	$(-0.3789, 0.3789)$
6	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.3650, 0.3650)$	$(-0.8473, 0.8473)$
7	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.3650, 0.3650)$	$(-0.3789, 0.3789)$
8	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$
9	$(-0.7355, 0.7355)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.3789, 0.3789)$
10	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$

Table D3 : Componentwise equivalence margins used for studying the type-I error risk of the intersection-union tests under the Obenchain model.

k	Scenario 1 $(-\varepsilon_1^{(k)}, \varepsilon_1^{(k)})$	Scenario 2 $(-\varepsilon_1^{(k)}, \varepsilon_1^{(k)})$	Scenario 3 $(-\varepsilon_1^{(k)}, \varepsilon_1^{(k)})$	Scenario 4 $(-\varepsilon_1^{(k)}, \varepsilon_1^{(k)})$
1	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.3110, 0.3110)$
2	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.1553, 0.1553)$
3	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.3727, 0.3727)$	$(-0.3994, 0.3994)$
4	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.3809, 0.3809)$	$(-0.2041, 0.2041)$
5	$(-0.8473, 0.8473)$	$(-0.5991, 0.5991)$	$(-0.4461, 0.4461)$	$(-0.4782, 0.4782)$
6	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.1866, 0.1866)$	$(-0.2000, 0.2000)$
7	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.1286, 0.1286)$	$(-0.2756, 0.2756)$
8	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.0170, 0.0170)$
9	$(-0.5991, 0.5991)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.2119, 0.2119)$
10	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.8473, 0.8473)$	$(-0.0781, 0.0781)$

Table D4: Summary statistics for the sample of APSDs from which the parameters used under POWSCEN2 were determined. [The S_{jk}^* are the (co-)variances S_{jk} multiplied by 10^7 .]

	Particle Size Category $k =$										
	0	1	2	3	4	5	6	7	8	9	10
$\bar{X}_k$	0.0100	0.1424	0.0707	0.1380	0.1718	0.2444	0.1434	0.0521	0.0048	0.0012	0.0213
S_{0k}^*	174.1	886.2	-104.2	-109.2	-326.0	-490.5	62.4	-73.8	3.9	-6.9	-15.9
S_{1k}^*	886.2	41439.2	-174.0	-3234.8	-12680	-15206	-5526.8	-4230.0	-144.5	-55.7	-1073.1
S_{2k}^*	-104.2	-174.0	3277.8	733.8	-2628.8	-221.2	-524.8	-180.6	-7.0	-6.4	-164.6
S_{3k}^*	-109.2	-3234.8	733.8	4740.7	-1586.9	895.1	-1188.8	-20.1	14.7	-12.9	-231.5
S_{4k}^*	-326.0	-12680	-2628.8	-1586.9	10066.6	3882.3	1713.3	1108.0	45.1	30.5	376.2
S_{5k}^*	-490.5	-15206	-221.2	895.1	3882.3	6911.3	1990.4	1757.6	54.8	29.7	396.8
S_{6k}^*	62.4	-5526.8	-524.8	-1188.8	1713.3	1990.4	2311.3	786.9	15.5	4.3	356.3
S_{7k}^*	-73.8	-4230.0	-180.6	-20.1	1108.0	1757.6	786.9	649.3	15.5	6.6	180.5
S_{8k}^*	3.9	-144.5	-7.0	14.7	45.1	54.8	15.5	15.5	3.2	-0.1	-1.0
S_{9k}^*	-6.9	-55.7	-6.4	-12.9	30.5	29.7	4.3	6.6	-0.1	1.5	9.3
S_{10k}^*	-15.9	-1073.1	-164.6	-231.5	376.2	396.8	356.3	180.5	-1.0	9.3	167.0

Table D5: Summary statistics for the Obenchain-transformed raw data underlying Table D4.

	Particle Size Category $k =$									
	1	2	3	4	5	6	7	8	9	10
$\check{X}_k$	2.634	1.995	2.675	2.881	3.257	2.724	1.704	-0.672	-2.069	0.803
$\check{S}_{1k}^*$	0.156	0.065	0.049	0.027	0.039	0.035	0.023	0.038	0.084	0.034
$\check{S}_{2k}^*$	0.065	0.217	0.162	0.151	0.173	0.145	0.167	0.148	0.203	0.152
$\check{S}_{3k}^*$	0.049	0.162	0.191	0.163	0.170	0.137	0.164	0.146	0.192	0.148
$\check{S}_{4k}^*$	0.027	0.151	0.163	0.279	0.203	0.177	0.205	0.171	0.246	0.199
$\check{S}_{5k}^*$	0.039	0.173	0.170	0.203	0.193	0.164	0.192	0.161	0.227	0.180
$\check{S}_{6k}^*$	0.035	0.145	0.137	0.177	0.164	0.147	0.166	0.136	0.196	0.161
$\check{S}_{7k}^*$	0.023	0.167	0.164	0.205	0.192	0.166	0.200	0.159	0.226	0.187
$\check{S}_{8k}^*$	0.038	0.148	0.146	0.171	0.161	0.136	0.159	0.146	0.190	0.145
$\check{S}_{9k}^*$	0.084	0.203	0.192	0.246	0.227	0.196	0.226	0.190	0.370	0.241
$\check{S}_{10k}^*$	0.034	0.152	0.148	0.199	0.180	0.161	0.187	0.145	0.241	0.197

Appendix E: DETAILED DISCUSSION OF SIMULATION RESULTS

As regards the behavior of the first of the intersection-union tests (constructed under the Dirichlet model and denoted IU-D test for brevity), the entries in Column 4 of Table 6(M) support the following conclusions: When the data satisfy the model assumptions, there is no reason to suspect that marked exceedances of the target significance level might occur. This holds true even for very small sample sizes and despite the fact that by theory the IU-D test, like both other variants of the intersection-union procedure we have investigated, maintains the significance level only asymptotically. However, the rate at which the rejection probability of the IU-D test converges to α from below can be very low, reflecting the general tendency of intersection-union tests to overconservatism. With samples taken from Obenchain distributions, inferences made by means of the IU-D test may be strongly misleading since then the rejection probability under the null hypothesis is not even bounded away from 100%, as becomes obvious from the results obtained under Scenario 1. Similarly, the results shown in Table 6(M) admit the conclusion that the Obenchain variant of the IU test (IU-O) has acceptable level properties only when the test is applied to data satisfying the model for which it is tailored. Regarding the IU-SP test, i.e., the asymptotically distribution-free version of the intersection-union test constructed under the semi-parametric location-scale model of § 5.1, the level was found to be maintained acceptably well.

The results displayed in Table 7(M) show that in all three tests constructed for the aggregate criterion of equivalence, the sample sizes needed for ensuring that the discrepancy between target significance level and size of the critical region keeps acceptably small, are fairly large even when the respective test is carried out with data satisfying the model assumptions under which it has been derived. For the scenarios involving Dirichlet distributed data, the parametric AG-D test (aggregate test derived under the Dirichlet model) yielded rejection probabilities under the null hypothesis that were similar to those obtained for the asymptotically distribution-free semi-parametric version of the test (AG-SP test). Convergence of the rejection probability to the nominal value .05 of α could be verified empirically in all scenarios. The speed of convergence was particularly slow in Scenario 2 where sample sizes of 10,000 and 100,000 were additionally examined in order to check that convergence eventually takes place. In Scenario 1, the AG-D test and the AG-SP test both turned out to be conservative. In Scenarios 3 and 4, both tests showed a markedly anticonservative behavior. Applying the test in the AG-O version being asymptotically valid under the Obenchain model, to Dirichlet distributed data gave rejection probabilities converging to zero with increasing sample sizes.

With data sampled from Obenchain distributions, the size of the AG-D test increased with increasing sample sizes. For the AG-SP test, its semi-parametric analogue, the convergence of the size to $\alpha = 0.05$ did not become as obvious in the simulations with Obenchain distributed data as was the case with the other data distributions. In Scenarios 1 and 2, the AG-SP test turned out conservative whereas in Scenarios 3 and 4 the differences between size and target level went in the opposite direction. The behavior of the AG-O version of the test resembled that of the AG-SP procedure in that respect. In Scenarios 3 and 4, the convergence of its rejection probability to α was somewhat faster than that of the AG-SP test. In Scenario 1, however, convergence was somewhat slower, and in Scenario 2 even considerably slower.

Inspecting the entries in Tables 8(M) and 9(M) immediately reveals that the power of both the IU and AG tests strongly depends on the specific alternative one is interested in: Given the sample size, the testing procedure and the type of distributions under comparison, the power attained under POWSCEN1 was always considerably larger than that under POWSCEN2, in accordance with the fact that the latter involved some components with markedly increased variability.

The major conclusions to be drawn from the results obtained for the power of the three tests for aggregate equivalence are twofold: Expectedly, for both parametric families of distributions consisting of vectors of the structure of an APSD, the test constructed under the assumption of the respective model is more powerful than the other two versions. However, the gain in power attained by using the appropriate parametric version instead of the asymptotically distribution-free procedure turned out to be modest in the case of the Dirichlet distribution and almost negligible with Obenchain distributed data. In the former, the largest difference shown by the entries in Table 9(M) in favor of AG-D is 15.6%, as compared with 0.7% for AG-O vs. AG-SP for the Obenchain model.

Comparisons across Tables 8(M) and 9(M) reveal that in the majority of investigated settings the parametric IU test attained higher power than its aggregate counterpart. These differences, which in some cases were $\geq 30\%$ on an absolute scale, reflect the fact that the cubic equivalence regions used in the IU tests were proper supersets of the spherical regions specified by the alternative hypotheses to be established by means of the aggregate tests. In fact, a test which is tailored for some null hypothesis contained in that of some other testing procedure, will typically be more powerful, and this effect turned out to overcompensate by far the tendency towards conservatism inherent in the IU approach.

Appendix F: INDIVIDUAL APSD’S MEASURED IN THE METHOD TRANSFER STUDY ANALYZED IN SECTION 7

Table F1a: Illustrating example, raw data, Sample 1.

Obs	Adapter	SIP	Stage 0	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Stage 6	Stage 7	Filter
1	0.0073	0.1001	0.0758	0.1519	0.1655	0.2699	0.1399	0.0611	0.0054	0.0010	0.0221
2	0.0075	0.0735	0.0807	0.1617	0.1740	0.2763	0.1405	0.0565	0.0056	0.0013	0.0225
3	0.0063	0.0727	0.0786	0.1627	0.1671	0.2805	0.1441	0.0570	0.0043	0.0012	0.0255
4	0.0059	0.0724	0.0555	0.1297	0.1685	0.3004	0.1661	0.0687	0.0064	0.0015	0.0249
5	0.0084	0.0651	0.0609	0.1440	0.1757	0.2941	0.1535	0.0641	0.0066	0.0014	0.0261
6	0.0078	0.0561	0.0582	0.1399	0.1802	0.2985	0.1610	0.0648	0.0065	0.0015	0.0254
7	0.0079	0.0936	0.0714	0.1335	0.1531	0.2823	0.1554	0.0641	0.0070	0.0031	0.0286
8	0.0095	0.0815	0.0759	0.1424	0.1611	0.2771	0.1535	0.0635	0.0070	0.0020	0.0264
9	0.0114	0.0716	0.0812	0.1481	0.1666	0.2697	0.1539	0.0637	0.0074	0.0015	0.0251
10	0.0060	0.0816	0.0906	0.1519	0.1658	0.2750	0.1488	0.0533	0.0051	0.0014	0.0205
11	0.0074	0.0629	0.0804	0.1618	0.1604	0.2822	0.1572	0.0576	0.0051	0.0010	0.0240
12	0.0064	0.0842	0.0942	0.1682	0.1566	0.2664	0.1455	0.0529	0.0044	0.0009	0.0203
13	0.0072	0.0887	0.0912	0.1667	0.1626	0.2680	0.1442	0.0501	0.0039	0.0010	0.0165
14	0.0044	0.1072	0.0864	0.1615	0.1609	0.2648	0.1420	0.0499	0.0047	0.0010	0.0172
15	0.0062	0.0821	0.0828	0.1604	0.1626	0.2757	0.1531	0.0534	0.0053	0.0011	0.0174
16	0.0061	0.0670	0.0600	0.1258	0.1512	0.3077	0.1845	0.0679	0.0064	0.0014	0.0221
17	0.0060	0.0619	0.0594	0.1336	0.1517	0.2928	0.1762	0.0714	0.0075	0.0041	0.0353
18	0.0089	0.1064	0.0947	0.1628	0.1525	0.2487	0.1396	0.0543	0.0048	0.0017	0.0255
19	0.0084	0.0962	0.0931	0.1595	0.1545	0.2575	0.1455	0.0549	0.0045	0.0013	0.0247
20	0.0069	0.0674	0.0776	0.1578	0.1573	0.2762	0.1606	0.0605	0.0055	0.0014	0.0289
21	0.0075	0.0587	0.0926	0.1784	0.1663	0.2641	0.1462	0.0543	0.0048	0.0015	0.0255

Table F1b: Illustrating example, raw data, Sample 2.

Obs	Adapter	SIP	Stage 0	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Stage 6	Stage 7	Filter
1	0.0080	0.0846	0.0873	0.1588	0.1901	0.2412	0.1598	0.0469	0.0037	0.0020	0.0176
2	0.0110	0.0876	0.0842	0.1628	0.1832	0.2514	0.1581	0.0459	0.0030	0.0009	0.0120
3	0.0092	0.0900	0.0821	0.1618	0.1850	0.2491	0.1560	0.0463	0.0038	0.0010	0.0157
4	0.0084	0.1231	0.0710	0.1394	0.1835	0.2390	0.1645	0.0501	0.0038	0.0009	0.0164
5	0.0107	0.0952	0.0716	0.1494	0.1832	0.2577	0.1629	0.0489	0.0034	0.0009	0.0161
6	0.0072	0.0755	0.0801	0.1550	0.1889	0.2552	0.1638	0.0501	0.0040	0.0019	0.0181
7	0.0082	0.1373	0.0863	0.1639	0.1741	0.2311	0.1391	0.0407	0.0031	0.0009	0.0153
8	0.0097	0.0952	0.0983	0.1755	0.1926	0.2280	0.1422	0.0404	0.0035	0.0010	0.0138
9	0.0105	0.0970	0.0996	0.1829	0.1807	0.2320	0.1356	0.0401	0.0032	0.0028	0.0156
10	0.0121	0.0676	0.0660	0.1452	0.1956	0.2696	0.1752	0.0546	0.0038	0.0011	0.0093
11	0.0051	0.0598	0.0678	0.1450	0.2060	0.2659	0.1779	0.0518	0.0038	0.0013	0.0156
12	0.0109	0.0762	0.0661	0.1427	0.1903	0.2786	0.1705	0.0484	0.0034	0.0013	0.0117
13	0.0126	0.0594	0.0590	0.1331	0.1633	0.2830	0.1838	0.0662	0.0088	0.0104	0.0203
14	0.0097	0.0751	0.0751	0.1523	0.1904	0.2543	0.1721	0.0491	0.0031	0.0014	0.0174
15	0.0110	0.0713	0.0762	0.1517	0.1983	0.2582	0.1638	0.0487	0.0035	0.0009	0.0164
16	0.0059	0.0680	0.0694	0.1415	0.2022	0.2672	0.1686	0.0517	0.0041	0.0013	0.0201
17	0.0099	0.0630	0.0634	0.1362	0.1781	0.2763	0.1772	0.0618	0.0076	0.0038	0.0227
18	0.0093	0.0793	0.0817	0.1578	0.1883	0.2539	0.1606	0.0477	0.0031	0.0013	0.0170
19	0.0044	0.0653	0.0841	0.1690	0.2032	0.2516	0.1558	0.0437	0.0038	0.0018	0.0174
20	0.0086	0.0705	0.0798	0.1539	0.2080	0.2527	0.1588	0.0447	0.0034	0.0018	0.0177
21	0.0060	0.0612	0.0755	0.1501	0.1905	0.2672	0.1667	0.0524	0.0060	0.0027	0.0216

Table F2: Empirical covariance matrix for the data of Table F1b. [The S_{jk}^* are the (co-)variances S_{jk} multiplied by 10^7 .]

	Particle Size Category $k =$										
	0	1	2	3	4	5	6	7	8	9	10
S_{0k}^*	50.7	50.8	-34.5	-33.2	-131.5	50.3	24.2	33.0	4.5	13.1	-27.4
S_{1k}^*	50.8	4131.6	998.9	974.6	-867.0	-2368.8	-1697.7	-725.5	-138.7	-155.6	-202.6
S_{2k}^*	-34.5	998.9	1141.9	1311.4	83.4	-1447.4	-1225.3	-584.3	-90.3	-82.6	-71.2
S_{3k}^*	-33.2	974.6	1311.4	1622.8	119.2	-1558.9	-1421.4	-690.5	-108.7	-93.8	-121.6
S_{4k}^*	-131.5	-867.0	83.4	119.2	1179.1	14.2	66.8	-213.7	-77.3	-125.2	-48.0
S_{5k}^*	50.3	-2368.8	-1447.4	-1558.9	14.2	2390.3	1606.7	815.2	142.4	147.7	118.3
S_{6k}^*	24.2	-1697.7	-1225.3	-1421.4	66.8	1696.7	1550.6	699.3	105.1	103.5	98.3
S_{7k}^*	33.0	-725.5	-584.3	-690.5	-213.7	815.2	699.3	406.0	81.6	90.5	88.5
S_{8k}^*	4.5	-138.7	-90.3	-108.7	-77.3	142.4	105.1	81.6	23.2	27.1	31.1
S_{9k}^*	13.1	-155.6	-82.6	-93.8	-125.2	147.7	103.5	90.5	27.1	43.5	32.0
S_{10k}^*	-27.4	-202.6	-71.2	-121.6	-48.0	118.3	98.3	88.5	31.1	32.0	102.7

Table F3: Empirical covariance matrix for the Obenchain-transformed sample of Table F1b.

	Particle Size Category $k =$									
	1	2	3	4	5	6	7	8	9	10
$\check{S}_{1k}$	0.104	0.088	0.077	0.071	0.055	0.054	0.044	0.031	-0.012	0.076
$\check{S}_{2k}$	0.088	0.110	0.100	0.096	0.078	0.076	0.066	0.062	0.054	0.106
$\check{S}_{3k}$	0.077	0.100	0.093	0.094	0.079	0.078	0.070	0.070	0.064	0.103
$\check{S}_{4k}$	0.071	0.096	0.094	0.103	0.090	0.091	0.083	0.083	0.070	0.112
$\check{S}_{5k}$	0.055	0.078	0.079	0.090	0.083	0.084	0.082	0.091	0.091	0.105
$\check{S}_{6k}$	0.054	0.076	0.078	0.091	0.084	0.087	0.085	0.093	0.091	0.107
$\check{S}_{7k}$	0.044	0.066	0.070	0.083	0.082	0.085	0.086	0.105	0.113	0.107
$\check{S}_{8k}$	0.031	0.062	0.070	0.083	0.091	0.093	0.105	0.165	0.217	0.139
$\check{S}_{9k}$	-0.012	0.054	0.064	0.070	0.091	0.091	0.113	0.217	0.446	0.171
$\check{S}_{10k}$	0.076	0.106	0.103	0.112	0.105	0.107	0.107	0.139	0.171	0.171

References

- Hoffelder, T., Gössl, R., and Wellek, S. (2015). Multivariate Equivalence Tests for Use in Pharmaceutical Development. *Journal of Biopharmaceutical Statistics* 25 (3), 417–437.
- Kotz, S., Johnson, N. L., and Balakrishnan, N. (2000) *Continuous Multivariate Distributions. Volume 1: Models and Applications. Second Edition*. New York: John Wiley & Sons, Inc.
- Ronning, G. (1989). Maximum likelihood estimation of Dirichlet distributions. *Journal of Statistical Computation and Simulation* 32 (4), 215–221.
- van der Vaart, A. W. (2000). *Asymptotic Statistics* . Cambrdige: Cambridge University Press.
- Wellek, S. (2010) *Testing Statistical Hypotheses of Equivalence and Noninferiority. Second Edition*. Boca Raton: Chapman & Hall / CRC.