

# Inference with Linear Equality and Inequality Constraints Using R: The Package `ic.infer`

Ulrike Grömping

BHT Berlin – University of Applied Sciences

---

## Abstract

In linear models and multivariate normal situations, prior information in linear inequality form may be encountered, or linear inequality hypotheses may be subjected to statistical tests. R package `ic.infer` has been developed to support inequality-constrained estimation and testing for such situations. This article gives an overview of the principles underlying inequality-constrained inference that are far less well-known than methods for unconstrained or equality-constrained models, and describes their implementation in the package.

*Keywords:* inequality constraints, quadratic programming, order-restricted linear model, `ic.infer`, `mvtnorm`.

---

## 1. Introduction

Inequality-constrained estimation and testing are relevant in various fields. Development of R package `ic.infer` has been instigated by customer satisfaction research: researchers believe that it is unreasonable if increase in an individual satisfaction aspect goes with decrease in overall satisfaction. Multicollinearity among individual satisfaction aspects can cause some estimated coefficients to be negative by chance, even if the true underlying coefficients are all non-negative. Thus, [Lipovetsky and Conklin \(2001\)](#) – in the context of the so-called “Shapley value regression” – have suggested an adjustment method for the coefficients that (among other purposes) is supposed to establish non-negativity. The author believes that an explicit restriction to non-negativity is more appropriate in such contexts (cf. also [Grömping and Landau 2009](#)). A further application of inequality-constrained inference lies in linear models with ordered categorical predictors, where it can be reasonable to a priori restrict estimated effects to be monotone in the expected direction. For example, if a monotone increase is expected, this can be implemented either by ensuring increasing size of effects, or – with different coding of dummy variables – by ensuring non-negativity of all affected coefficients. Attention to inequality-constrained inference is particularly high in econometrics, cf. e.g., [Gourieroux and Monfort \(1995, Chapter 21\)](#). This is also reflected by the fact that software implementation of inequality-constrained inference is found in econometric software (e.g. `GAUSS`, [Aptech Systems, Inc. 2009](#)) or in the econometric parts of general statistical software (e.g. the product `ETS` in `SAS`, [SAS Institute, Inc. 2009](#)), if at all. In R, constrained estimation of means or linear model coefficients can be handled by optimization packages, such as `quadprog` ([Turlach](#)

and Weingessel 2007). However, direct usage of optimization routines forces the data analyst to concentrate on numerical instead of statistical considerations and may lead to erroneous application in applied statistics. In addition, optimization routines do not automatically yield the ingredients for inference in terms of hypothesis tests. Distributional results for inequality-constrained estimates are far more complicated than standard distributional results. The recommended likelihood ratio approach to testing under normality requires calculation of mixing weights for mixtures of  $\chi^2$  or Beta distributions. While these can in principle be calculated using package **mvtnorm** (Genz, Bretz, Miwa, Mi, Leisch, Scheipl, and Hothorn 2009), literature on their calculation is not widely known, so that it is a severe impediment against application of inequality-related testing if they are not readily available. R package **ic.infer** implements inequality-constrained estimation and likelihood ratio tests for normal means and linear models. It is the first *statistical* package in R that accomodates general-purpose inequality restrictions (there are already packages that offer specific methods like the **Bioconductor** package **SAGx** (Broberg 2009) that offers the non-parametric Jonckheere-Terpstra trend test). The package is available from the Comprehensive R Archive Network at <http://CRAN.R-project.org/package=ic.infer>.

Literature on inequality-constrained likelihood ratio testing dates back to Bartholomew (1961), who proposed a test for detecting trends in group means. Kudô (1963) introduced the theory on testing the null hypothesis of mean 0 within the non-negative orthant as the parameter space under multivariate normality. Shapiro (1988) gave a review of the work on inequality-constrained testing up to that point in time, generalizing tests in the spirit of Kudô as well as tests of an inequality restriction against an unrestricted alternative. Sasabuchi (1980) introduced a likelihood ratio test for the reversed test problem, where the inequality restriction (with strict inequalities only) constitutes the alternative hypothesis. The latter test coincides with the intersection-union test (cf. Gleser 1973), which rejects its union null hypothesis, a union of individual null hypotheses, at level  $\alpha$  if and only if all individual null hypotheses are rejected by level  $\alpha$  tests (i.e., the  $p$  value is the maximum  $p$  value from individual tests). The monographs by Robertson, Wright, and Dykstra (1988) and Silvapulle and Sen (2004) give a broad overview of inequality-constrained inference.

This article discusses inference under linear inequality and equality constraints in normal distribution situations, both concerning the expectation vector in a homogeneous multivariate normal population and the parameters in normal linear models. Like in the linear model, estimation is based on projection and is applicable even without the normality assumption. For testing, however, a normality assumption is required. Nevertheless, for linear models, normality of the random error is not necessary; it is also sufficient to request multivariate normality of the coefficient estimators, which holds asymptotically in large samples. Even non-linear models can be treated based on the asymptotic normal distribution of the estimated coefficients (cf. e.g., chapter 4 of Silvapulle and Sen 2004).

Important theoretical foundations are polyhedral cones and translated versions thereof, probabilities that a multivariate normal random vector lies in a polyhedral cone – or the non-negative orthant –, and projections onto polyhedral cones. These are discussed in Section 2, along with the normal distribution probabilities that projections onto cones lie in linear spaces of certain dimensions and distributions of quadratic forms involving inequality-constrained estimates. Section 3 discusses the various test problems and likelihood ratio tests for them, emphasizing their different nature that has already been mentioned in the brief literature review above. Section 4 transfers inequality-restricted estimation and testing to the normal

linear model. Here, decomposition of  $R^2$  into contributions by individual regressors is also treated, in a spirit similar to Chevan and Sutherland (1991). Functionality in R package **ic.infer** is introduced in Section 5 using two examples: Robertson *et al.* (1988) data on grade point averages of Iowa university first-years are analysed by two-way analysis of variance, and female body fat values are analysed by multiple regression, based on a small dataset published by Kutner, Nachtsheim, and Neter (2004). Apart from pointing out the possibilities of **ic.infer**, Section 5 also gives an idea about the computational burden that comes with calculation of  $R^2$  decomposition and weights for hypothesis tests. The final section discusses plans for extending the package.

Throughout this article, all inequality relations between vectors are understood component wise.

## 2. Cones, cone probabilities, and projections onto cones

This section introduces polyhedral cones, probabilities for polyhedral cones under multivariate normality, and projections onto cones. Probabilities that projections onto cones lie in linear spaces of certain dimensions and distributions of quadratic forms are also discussed. The idea is to give the reader a basic understanding of the principles behind inequality-constrained estimation and testing for the situations covered in R package **ic.infer**. Readers interested in more detail are referred to Silvapulle and Sen (2004) or other cited literature.

### 2.1. Polyhedral cones

Any set of the form

$$C_R = \{y \in \mathbb{R}^p : Ry \geq 0\} \quad (1)$$

with a row-regular  $m \times p$  matrix  $R$  is called a polyhedral cone. The linear space, for which all restrictions  $Ry \geq 0$  are fulfilled with equality (nomenclature: are “active”), is called the vertex of the cone. The margins, for which some of the inequalities are active while others are fulfilled with inequality are called “faces” of the cone. The vertex is the lowest-dimensional face, and the interior of the cone can be seen as the highest-dimensional face. Even when adding additional restrictions that are always requested to hold with equality, the resulting set

$$C_{R1,R2} = \{y \in \mathbb{R}^p : R_1y \geq 0, R_2y = 0\}$$

is still a polyhedral cone, however with the highest-dimensional face having a dimension lower than  $p$ . It is a defining property of cones that  $y \in C$  implies  $\lambda y \in C$  for any  $\lambda > 0$ .

If the “0” on the right-hand side of the relations defining the cone is replaced by some possibly non-zero  $m \times 1$  vector  $r$ , (1) becomes

$$C_{R,r} = \{y \in \mathbb{R}^p : Ry \geq r\},$$

and  $C_{R1,R2,r1,r2}$  is analogously defined. These latter sets are translated cones, i.e., their vertex no longer is or includes the origin of  $p$ -dimensional space. Most estimation results hold for restrictions to cones and translated cones, while some testing results are not easily transferred from cones to translated cone restrictions, at least not in the current implementation of **ic.infer**. Most results in this article are presented for cones only (i.e.,  $r = 0$ ).

If a positive definite matrix  $\Sigma$  is given, the “polar” or “dual” cone of  $C_R$  is defined as the set  $C_R^{o,\Sigma} = \{y : -\Sigma^{-1}y \in C_R\} = C_{-R\Sigma^{-1}}$ . The top row of Figure 1 shows the positive orthant (cone  $C_I$ ) with its polar cone for two different covariance matrices  $\Sigma$  in bivariate space.

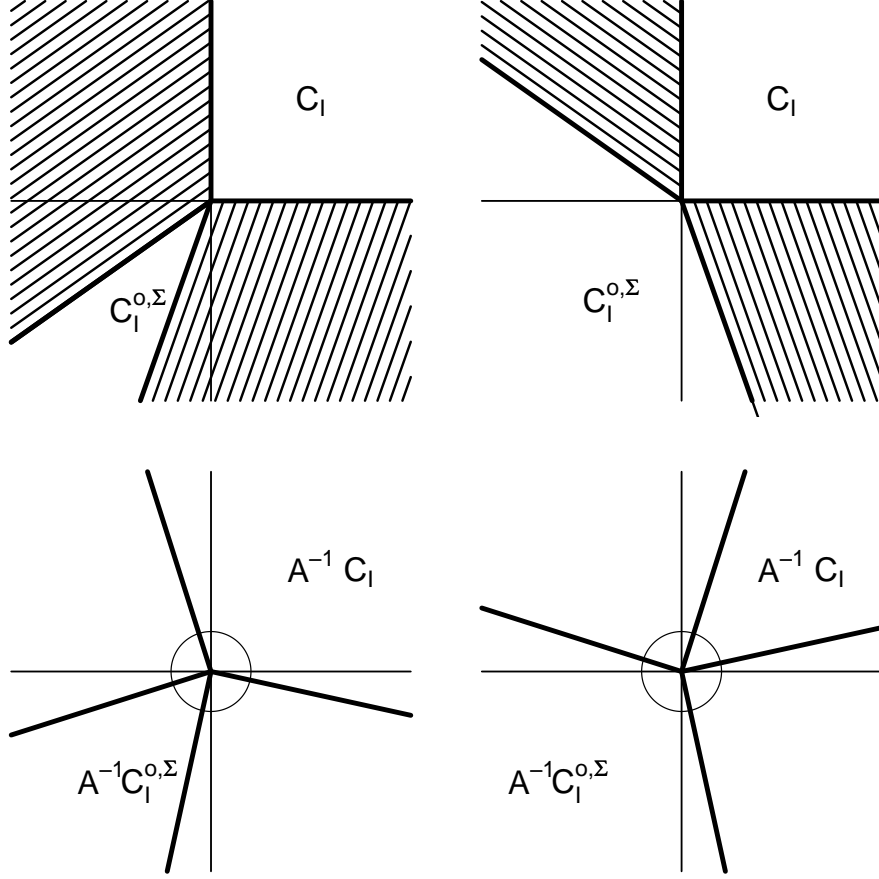


Figure 1: Positive orthant with its polar cone,  $\sigma_{11} = 1, \sigma_{22} = 2$ . Left:  $\sigma_{12} = 1/\sqrt{2}$ , Right:  $\sigma_{12} = -1/\sqrt{2}$ . Top: original scale. Bottom: transformed to identity covariance matrix

## 2.2. Multivariate normal cone probabilities

The non-negative orthant is the special polyhedral cone  $C_I$  with the identity matrix  $I$ . Let  $y \sim N(0, \Sigma)$  for a  $p$ -dimensional positive definite covariance matrix  $\Sigma$ . Then, there is a regular matrix  $A$  with  $AA^\top = \Sigma$ . We will denote the probability  $P(y \in C_I)$  as  $p(\Sigma)$ . Then  $p(\Sigma) = P(y \in C_I) = P(A^{-1}y \in C_A)$  for the normal random vector  $A^{-1}y$  with covariance matrix  $I$ . For low-dimensional random vectors, the latter probability can geometrically be obtained as the proportion of the unit sphere that is part of the cone  $C_A$ . The bottom row of Figure 1 illustrates the probabilities associated with the positive orthant for the two examples from the top row of the figure ( $C_A = A^{-1}C_I$ ): The probability for the non-negative orthant is  $1/3$  for  $\sigma_{12} = 1/\sqrt{2}$  and  $1/6$  for  $\sigma_{12} = -1/\sqrt{2}$  (the probabilities for the respective polar cones

are vice versa). Calculation of rectangle probabilities such as  $p(\Sigma)$  is implemented in **ic.infer** through access to package **mvtnorm** that uses Monte Carlo methods for this calculation.

For a row-regular matrix  $R$ , with  $\Sigma$  and  $A$  as defined above,  $P(y \in C_R) = P(A^{-1}y \in C_{RA}) = P(RAA^{-1}y \in C_I) = P(\tilde{y} \in C_I)$  for  $\tilde{y} \sim N(0, R\Sigma R^\top)$ . Thus, the probability  $P(y \in C_R)$  can be obtained as the orthant probability  $p(R\Sigma R^\top)$  (and thus again be calculated using **mvtnorm**). The probability for the polar of  $C_I$ ,  $C_I^{o,\Sigma}$ , can be obtained by choosing  $R = -\Sigma^{-1}$  and is thus  $p(\Sigma^{-1})$ ; analogously, the probability for the polar cone  $C_R^{o,\Sigma}$  is  $p((R\Sigma R^\top)^{-1})$ .

If the probabilities for translated cones are considered, things become notationally more complicated, but the probabilities remain unchanged when considering appropriately translated distributions.

The orthant probabilities together with the results of the following section on projections onto polyhedral cones yield all necessary ingredients for deriving null distributions of test statistics for test problems involving inequality constraints.

### 2.3. Projections onto cones

The projection of a  $p \times 1$  vector  $y$  onto the polyhedral cone  $C_R$  along the positive definite  $p \times p$ -matrix  $\Sigma$  is defined as the vector  $\hat{\mu}^*$  that minimizes the quadratic form

$$(y - \mu)^\top \Sigma^{-1} (y - \mu) \quad (2)$$

w.r.t.  $\mu \in C_R$ . If  $y \in C_R$ ,  $\hat{\mu}^* = y$ , if  $y \in C_R^{o,\Sigma}$ ,  $\hat{\mu}^* = y - \Sigma R^\top (R\Sigma R^\top)^{-1} Ry$ , which is the projection of  $y$  along  $\Sigma$  onto the  $(p - m)$ -dimensional vertex of cone  $C_R$ . (In particular, if  $m = p$ ,  $\hat{\mu}^* = 0$  for  $y \in C_R^{o,\Sigma}$ .) Otherwise,  $\hat{\mu}^*$  lies in one of the faces of  $C_R$  with intermediate dimension and is the projection onto the smallest linear space that contains this face.  $\hat{\mu}^*$  lies in a particular face of the cone  $C_R$ , if  $y$  lies in a particular cone. This can be illustrated by looking at Figure 1: The hatched regions in the top row are the cones of  $y$ -values that project onto one-dimensional faces of the non-negative orthant. Projection of any  $y$ -vector from these regions onto the non-negative orthant moves the  $y$ -vector parallel to the hatch-lines onto the margin of the non-negative orthant.

We have already seen that  $\hat{\mu}^* \in C_R$  if and only if  $y \in C_R$ , and that  $R\hat{\mu}^* = 0$  if and only if  $y \in C_R^{o,\Sigma}$ . Cones that lead to  $\hat{\mu}^*$  with intermediate dimensions are now investigated. Suppose the row-regular  $m \times p$  matrix  $R$  is composed of the matrices  $R_1$  (the first  $m_1$  rows) and  $R_2$  (the remaining  $m_2$  rows). The projection of  $y$  onto the linear space defined by  $R_2\mu = 0$  is known to be  $\tilde{\mu} = y - \Sigma R_2^\top (R_2 \Sigma R_2^\top)^{-1} R_2 y$ , so that

$$y - \tilde{\mu} = \Sigma R_2^\top (R_2 \Sigma R_2^\top)^{-1} R_2 y. \quad (3)$$

It is known from optimization theory that  $\tilde{\mu}$  minimizes (2), if and only if the component wise minimum of  $R\tilde{\mu}$  and  $-(R\Sigma R^\top)^{-1}(Ry - R\tilde{\mu})$  is a vector of zeroes. Straightforward calculations show that the first  $m_1$  elements of  $-(R\Sigma R^\top)^{-1}(Ry - R\tilde{\mu})$  are 0 and the last  $m_2$  elements of  $R\tilde{\mu}$  are 0. Hence,  $\tilde{\mu}$  is the projection, if and only if  $R_1\tilde{\mu} \geq 0$  and the last  $m_2$  elements of  $-(R\Sigma R^\top)^{-1}(Ry - R\tilde{\mu})$  are non-negative. Inserting (3) into the latter formula yields the last  $m_2$  elements as  $-(R_2 \Sigma R_2^\top)^{-1} R_2 y$ . Thus,  $\tilde{\mu}$  is the projection of  $y$  onto  $C_R$  if and only if

$$R_1\tilde{\mu} \geq 0 \quad \text{and} \quad -(R_2 \Sigma R_2^\top)^{-1} R_2 y \geq 0. \quad (4)$$

This is an important intermediate result for deriving the null distributions of quadratic forms.

## 2.4. Distribution of minimized quadratic forms

Under  $y \sim N(\mu, \Sigma)$ , the size of (2) is independent of which cone  $y$  is an element of, i.e., which face  $\hat{\mu}^*$  belongs to:  $(y - \mu)^\top \Sigma^{-1}(y - \mu) \sim \chi^2(p)$ . For the rest of this subsection, it will be assumed that the true expectation vector  $\mu$  lies in the vertex of cone  $C_R$ . (This setup is the least favourable configuration for those test statistics which depend on quadratic forms, cf. (8) and (9) below.) Given that  $\hat{\mu}^*$  lies in a particular  $i$ -dimensional face of the cone, it can be written as the projection of  $y$  onto the adequate  $i$ -dimensional linear space. This projection is normally distributed, and the fact whether or not it fulfills the conditions (4), i.e., is the minimizer of (2), is again independent of the size of its quadratic forms because of the assumption that  $\mu$  is in the vertex of  $C_R$ . Thus, whenever  $\hat{\mu}^*$  lies in a particular  $i$ -dimensional face of  $C_R$ ,

$$(y - \mu)^\top \Sigma^{-1}(y - \mu) = (y - \hat{\mu}^*)^\top \Sigma^{-1}(y - \hat{\mu}^*) + (\hat{\mu}^* - \mu)^\top \Sigma^{-1}(\hat{\mu}^* - \mu),$$

where the first summand is distributed as  $\chi^2(p - i)$  and the second as  $\chi^2(i)$ . ( $\chi^2(0)$  is the one-point distribution on “0”.)

If  $y \in C_R, i = p$ . This happens with probability  $p(R\Sigma R^\top)$ . On the other hand, if  $y \in C_R^{o,\Sigma}, i = p - m$ , which happens with probability  $p((R\Sigma R^\top)^{-1})$ . For determining the probabilities of faces of intermediate dimensions, the probability of condition (4) needs to be calculated. Since the two vectors involved in (4) are independent, the probability that  $\hat{\mu}^*$  lies in a particular face of the form  $R_1\hat{\mu}^* > 0, R_2\hat{\mu}^* = 0$  is given as the product of the probabilities for the two conditions, i.e.,

$$P(R_1\tilde{\mu} \geq 0)P(-(R_2\Sigma R_2^\top)^{-1}R_2y \geq 0). \quad (5)$$

The second factor in the product (5) is a cone probability and can be calculated as the orthant probability  $p((R_2\Sigma R_2^\top)^{-1})$ . The first factor is obtainable as the orthant probability  $p(V_{1.2})$  based on the matrix

$$V_{1.2} = R_1\Sigma R_1^\top - R_1\Sigma R_2^\top(R_2\Sigma R_2^\top)^{-1}R_2\Sigma R_1^\top, \quad (6)$$

which is the covariance matrix of  $R_1\tilde{\mu}$  and coincides with the conditional covariance matrix  $\text{COV}(R_1Y|R_2Y)$ . (Note that  $V_{1.2}$  is also the inverse of the upper left block of the appropriately partitioned matrix  $(R\Sigma^{-1}R^\top)^{-1}$ .) Thus, with rows of  $R$  arranged in appropriate order and  $m_2$  chosen as large as possible, the probability that  $\hat{\mu}^*$  lies in the face  $R_1\hat{\mu}^* > 0$  and  $R_2\hat{\mu}^* = 0$  is given as

$$p(V_{1.2})p((R_2\Sigma R_2^\top)^{-1}).$$

Analogously to Kudô (1963), the probability that  $\hat{\mu}^*$  lies in any face of a specified dimension  $i$  is then obtained as the sum of all probabilities for individual faces of that dimension, i.e., it is

$$w_{i-(p-m)}(m, R\Sigma R^\top) = \sum_{J: \|J\|=p-i} p((R_J\Sigma R_J^\top)^{-1})p(V_{J^c,J}) \quad (7)$$

with

- $i \in \{p - m, \dots, p\}$ ,
- $J \subset \{1, \dots, m\}$ ,
- $R_J$  the  $(p - i) \times p$  sub matrix of  $R$  with rows indexed by elements of  $J$  (matrix for active restrictions in current face),

- $J^c$  the complement of  $J$  in  $\{1, \dots, m\}$ ,
- $V_{J^c, J}$  constructed analogously to Equation (6),
- and  $w_j(p, \Sigma)$  the conventional notation for the probability that the projection along  $\Sigma$  of a  $p$ -variate random vector  $y \sim N(0, \Sigma)$  onto the non-negative orthant is of dimension  $j$  (e.g. Shapiro 1988; Silvapulle and Sen 2004).

Remember that the quadratic forms  $(y - \hat{\mu}^*)^\top \Sigma^{-1} (y - \hat{\mu}^*)$  and  $(\hat{\mu}^* - \mu)^\top \Sigma^{-1} (\hat{\mu}^* - \mu)$  follow a  $\chi^2$  distribution with  $p - i$  or  $i$  degrees of freedom, respectively, for  $\hat{\mu}^*$  in a given  $i$ -dimensional face of the restriction cone independent of the event which face  $\hat{\mu}^*$  belongs to. Thus, it is now verified that the overall distribution of these quadratic forms is a mixture of  $\chi^2$  distributions with different degrees of freedom, and mixing probabilities are obtained according to (7). Analogously, with  $\hat{\mu}_=$  denoting the equality-constrained estimate ( $R\hat{\mu}_= = 0$ ), the  $\chi^2(p - m)$  quadratic form  $(y - \hat{\mu}_=)^\top \Sigma^{-1} (y - \hat{\mu}_=)$  can be decomposed as

$$(y - \hat{\mu}_=)^\top \Sigma^{-1} (y - \hat{\mu}_=) = (y - \hat{\mu}^*)^\top \Sigma^{-1} (y - \hat{\mu}^*) + (\hat{\mu}^* - \hat{\mu}_=)^\top \Sigma^{-1} (\hat{\mu}^* - \hat{\mu}_=).$$

The summands are distributed as  $\chi^2(p - i)$  and  $\chi^2(i - p + m)$  respectively for  $\hat{\mu}^*$  in an  $i$ -dimensional face of  $C_R$ , and the overall distributions of the summands are again mixtures of  $\chi^2$  distributions with weights according to (7).

### 3. Hypothesis tests and linear inequality constraints

In the following, all theoretical discussions assume a normal distribution, initially  $y \sim N(\mu, \Sigma)$  with known  $\Sigma$ , then relaxed to  $\Sigma = \sigma^2 \Sigma_0$  with known  $\Sigma_0$  and finally a normal linear model with unknown positive variance parameter. As mentioned in the introduction, it is possible to cover more general situations for large samples, where coefficient estimates often approximately follow a multivariate normal distribution.

#### 3.1. Three test problems

As has been mentioned in the introduction, there are three basic test problems that can be considered in connection with an inequality constraint ( $\neg$  denotes “not”):

$$H_{0A} : R\mu = r \quad vs \quad H_{1A} : R\mu \geq r, \quad (\text{TP1})$$

$$H_{0B} : R\mu \geq r \quad vs \quad H_{1B} : \neg R\mu \geq r, \quad (\text{TP2})$$

$$H_{0C} : \neg R\mu > r \quad vs \quad H_{1C} : R\mu > r. \quad (\text{TP3})$$

In all three test problems,  $R$  is a row-regular  $m \times p$  matrix,  $r$  an  $m \times 1$  vector, and inequalities are to be understood component wise. Likelihood ratio tests for these three test problems (cf. Shapiro 1988; Sasabuchi 1980) are based on

$$LR = \frac{\max_{\mu \in H_0} (f(y; \mu))}{\max_{\mu \in H_0 \cup H_1} (f(y; \mu))},$$

where  $f(\cdot)$  denotes the appropriate probability density function.

### 3.2. Likelihood ratio tests for known $\Sigma$

For known  $\Sigma$ , the resulting test statistics simplify to

$$T_1 = (\hat{\mu}^* - \hat{\mu}_=)^\top \Sigma^{-1} (\hat{\mu}^* - \hat{\mu}_=), \quad (8)$$

$$T_2 = (y - \hat{\mu}^*)^\top \Sigma^{-1} (y - \hat{\mu}^*), \quad (9)$$

and

$$T_3 = \min_{j=1, \dots, m} \frac{(Ry - r)_j}{\sqrt{(R\Sigma R^\top)_{j,j}}}, \quad (10)$$

where  $\hat{\mu}^*$  is the inequality-constrained estimate and  $\hat{\mu}_=$  is the equality-constrained estimate. The critical value for  $T_3$  is easily obtained as the  $(1 - \alpha)$  quantile from the standard normal distribution, and the  $p$  value is accordingly the maximum of the one-sided  $p$  values of tests for individual restrictions. It is known that the least favourable configuration for the likelihood ratio tests of **TP1** and **TP2** is given as  $R\mu = r$ . According to Section 2.4, the null distributions of  $T_1$  and  $T_2$  are mixtures of  $\chi^2$  distributions under this least favorable configuration, and  $p$  values can be easily calculated as a weighted sum of  $\chi^2$ - $p$  values, once the mixing weights are known (cf. Equation (7)).

Figure 2 (page 9) shows the null hypothesis and rejection regions for two simple two-dimensional examples (restriction with  $R = I_2$ ,  $r = 0_2$ , i.e., non-negative orthant) for the three test problems. Note that the shape of the critical region strongly depends on the correlation structure between the components of  $y$  for **TP1** and **TP2**, while it does not for **TP3**. In particular, note that the null hypothesis of **TP1** is equality *only*. Rejection of this null hypothesis does not provide evidence for validity of the inequality restriction in any way, although formulation of the test problem has mislead many researchers to interpret it in this way. The inequality restriction comes in as a *prior belief*. The test thus concentrates its power as much as possible in the area where the restriction holds; however, even situations with all constraints strictly (and statistically significantly) violated can lead to rejection of  $H_{0A}$  (cf. Silvapulle 1997, for an example). This is also reflected by the rejection region for **TP1** for positive correlation (top left in Figure 2) that does extend into the negative orthant. Some authors (e.g., Cohen, Kemperman, and Sackrowitz 2000) criticize the likelihood ratio principle for this behavior, others, e.g., Perlman and Wu (2003) or Silvapulle (1997) defend the likelihood ratio principle and correctly place the fault where it belongs: interpretation of test results in a restricted parameter space must account for the *a-priori* nature of the restriction, and must realize that the only guaranteed null hypothesis is the stated null hypothesis (i.e., equality in case of **TP1**).

The power of the test for **TP3** is generally poor, especially in case of a relatively large number of restrictions. It can be far lower than the chosen significance level even on relevant parts of the alternative hypothesis. Thus, the role of **TP3** is rather to intellectually complete the set of tests (and to prevent erroneous interpretation of significant results from testing **TP1**) than to actually provide a useful test for applications. Of course, there are occasionally applications where this test rejects its null hypothesis, for example when investigating the monotonicity of diamond prices w.r.t. color and clarity in the diamond data published by Chu (2001). **TP3** has motivated researchers to develop tests that are uniformly more powerful than the likelihood ratio test (e.g. Berger 1989; Liu and Berger 1995). Perlman and Wu (1999) passionately

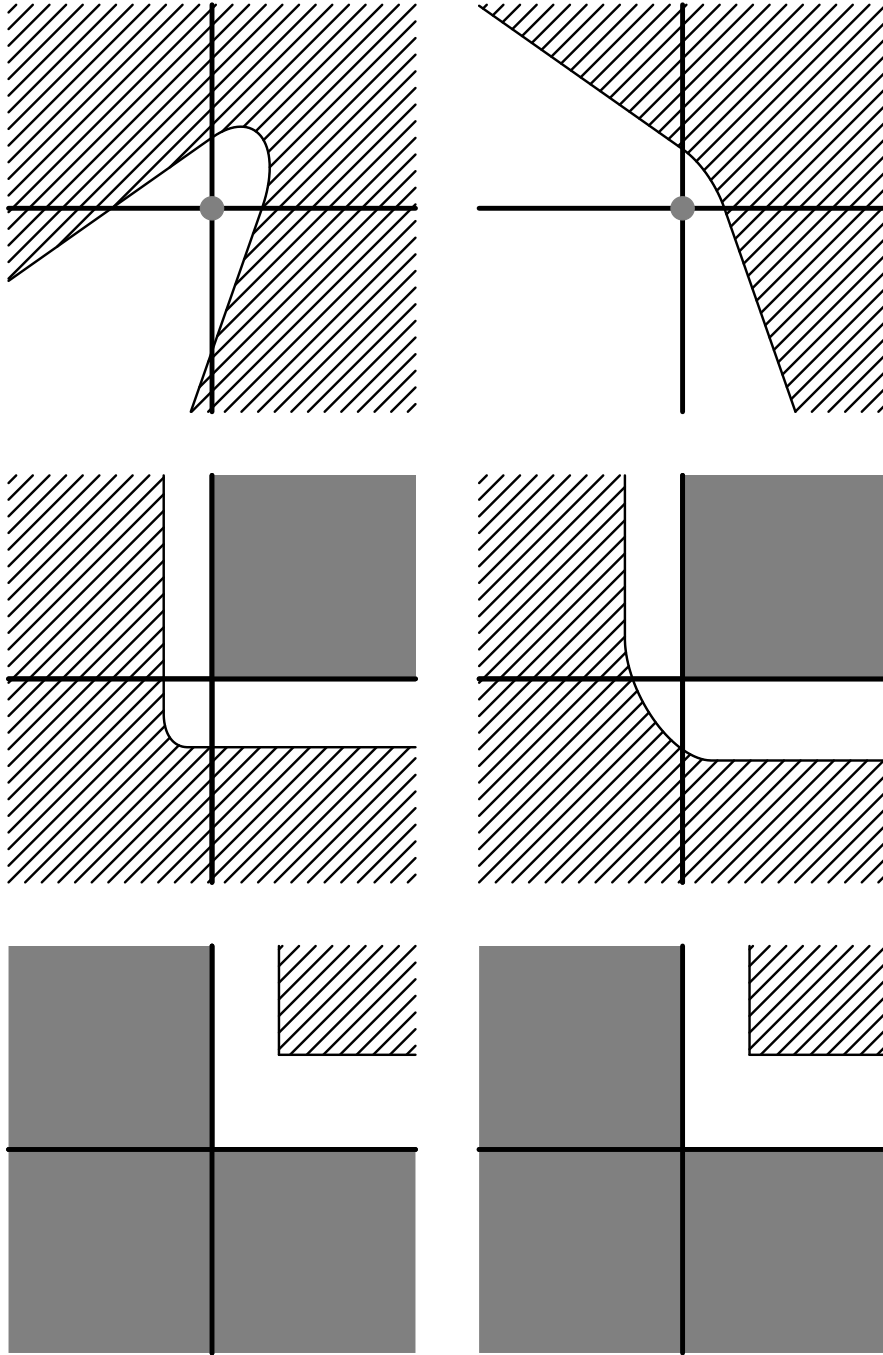


Figure 2: Restriction  $\mu \geq 0$ ,  $\sigma_{11} = 1$ ,  $\sigma_{22} = 2$ . Left:  $\sigma_{12} = 1/\sqrt{2}$ , Right:  $\sigma_{12} = -1/\sqrt{2}$ . Top: Null space (grey dot) and rejection region (hatched) for TP1. Middle: Null space (grey) and rejection region (hatched) for TP2. Bottom: Null space (grey) and rejection region (hatched) for TP3.

defend the likelihood ratio principle against these “Emperor’s new tests” – the author agrees with their assessment, cf. also Grömping (1996).

### 3.3. Likelihood ratio tests for unknown $\sigma^2$

The covariance matrix  $\Sigma$  has so far been assumed completely known. In this section, it is given as  $\Sigma = \sigma^2 \Sigma_0$  with known  $\Sigma_0$  but unknown  $\sigma^2 > 0$ . Furthermore, it is assumed that an estimate  $s^2$  for  $\sigma^2$  is available that is distributed as  $\sigma^2 \chi^2(df_{error})$  independently of  $y$ . This is for example the case, if a researcher has access to a vector of estimated coefficients, the estimated covariance matrix and the variance estimate from a normal linear model.

Then, the test statistic (10) for TP3 – with  $\Sigma$  replaced by  $s^2 \Sigma_0$  – has to be compared to the  $(1 - \alpha)$  quantile of a  $t(df_{error})$  distribution instead of the standard normal quantile, and  $p$  values are again obtained as the maximum  $p$  value from  $m$  individual t-tests.

The test statistics (8) and (9) for test problems TP1 and TP2 have to be modified as follows:

$$T_{unknown} = \frac{T_{\Sigma_0}}{T_{\Sigma_0} + df_{error}s^2}, \quad (11)$$

where  $T_{\Sigma_0}$  denotes the statistic (8) or (9) based on the known matrix  $\Sigma_0$ . Then,  $T_{unknown}$  is under the least favourable configuration (equality for all restrictions) distributed as a mixture of Beta distributions with the same mixing weights as in case of known  $\sigma^2$  (these can be determined without knowledge or estimation of  $\sigma^2$ , since constant multipliers do not affect the probabilities). The first parameter for the Beta distributions is half the degrees of freedom of the corresponding  $\chi^2$  distribution for the known- $\Sigma$  case, the second parameter is  $df_{error}/2$ . Thus,  $p$  values for TP1 and TP2 can again be easily obtained, if mixing weights are available.

## 4. Estimation in the linear model

In this section, estimation of linear model coefficients in the multivariate normal situation is discussed. Although normality is assumed throughout, since the estimates are derived as maximum likelihood estimates, estimation is valid for more general situations. The most simple form of the linear model is given as

$$y = X\beta + \varepsilon,$$

with  $\varepsilon$  assumed to be a vector of uncorrelated errors with homogeneous variances. The ordinary least squares (OLS) method estimates  $\beta$  by unconstrained minimization of  $(y - X\beta)^\top (y - X\beta)$ , which yields the well-known formula  $\hat{\beta}_{OLS} = (X^\top X)^{-1} X^\top y$ , and the fits  $\hat{y} = X\hat{\beta}_{OLS}$  are the orthogonal projection of  $y$  on the space spanned by  $X$ . Under the normality assumption, the linear model can be formulated as

$$y \sim N(X\beta, \sigma^2 V_0),$$

with  $X$  a known  $n \times p$  matrix,  $\beta$  an unknown  $p \times 1$  vector,  $V_0$  a known positive definite  $n \times n$  matrix,  $\sigma^2 > 0$  either known or unknown. Then, the unconstrained estimate is  $\hat{\beta} = \hat{\beta}_{GLS} = (X^\top V_0^{-1} X)^{-1} X^\top V_0^{-1} y$ , and the vector of unconstrained fits  $\hat{y}$  is the projection of  $y$  along  $V_0$  (or, equivalently,  $\sigma^2 V_0$ ) onto the space defined by  $X$ , i.e.,  $\hat{y} = X\hat{\beta}$  with  $\hat{\beta}$  the minimizer of the quadratic form  $(y - X\beta)^\top V_0^{-1} (y - X\beta)$  w.r.t.  $\beta \in \mathbb{R}^p$ . Although package **ic.infer** only

covers diagonal  $V_0$  (i.e., heterogeneity of variances, cf. e.g., the grades example in Section 5.1), general positive definite  $V_0$  are used here.

If linear inequality constraints of the form  $R\beta \geq r$  are introduced into the linear model, where  $R$  is a matrix with linearly independent rows so that the restrictions define a (translated) cone in the parameter space  $\mathbb{R}^p$  (cf. 2.1), the thus-constrained fit  $\hat{y}^* = X\hat{\beta}^*$  is obtained by minimizing  $(y - X\beta)^\top V_0^{-1}(y - X\beta)$  w.r.t.  $\beta \in C_{R,r}$ . Its target space is also a (potentially translated) convex polyhedral cone, and the fit is the projection onto that (translated) cone. It is easily verified that the minimum of

$$(y - X\beta)^\top V_0^{-1}(y - X\beta)$$

is the sum

$$(y - X\hat{\beta})^\top V_0^{-1}(y - X\hat{\beta}) + (\hat{\beta} - \hat{\beta}^*)^\top X^\top V_0^{-1}X(\hat{\beta} - \hat{\beta}^*),$$

i.e., the minimizing parameter vector can be simply obtained by projecting  $\hat{\beta}$  onto the (translated) cone  $C_{R,r}$  along the matrix  $(X^\top V_0^{-1}X)^{-1}$ . Hence, application of the results from Sections 2 and 3 is straightforward, when replacing  $y$  with  $\hat{\beta}$ , and  $\mu$ ,  $\hat{\mu}^*$ , and  $\hat{\mu}_=$  with  $\beta$ ,  $\hat{\beta}^*$ , and  $\hat{\beta}_=$ , respectively.

It is of course possible to obtain an  $R^2$  value for the constrained model in much the same way as for the unconstrained model (if constraints restrict the intercept so that the residuals do not sum to zero, implausible results can occur).  $R^2$  of the constrained model is never larger than  $R^2$  of the unconstrained model, and a small reduction by introduction of constraints is an informal indication that constraints are compatible with the data.

In unconstrained linear models, decomposition of  $R^2$  has been considered as a method for assessing relative importance of regressors. This approach has first been proposed by Lindeman, Merenda, and Gold (1980) and has been suggested in more general contexts by Chevan and Sutherland (1991). It has been reinvented many times, e.g., by Kruskal (1987) or as “Shapley value regression” by Lipovetsky and Conklin (2001). It is implemented in R packages **hier.part** (Walsh and Mac Nally 2008) and **relaimpo** (Grömping 2006), where these two packages have quite different emphasis. The statistical properties of  $R^2$  decomposition in unrestricted linear models have been investigated in Grömping (2007).

$R^2$  decomposition based on the Shapley value can also be applied for constrained regression models. However, there are two caveats: First, implementation for inequality-constrained models is more computationally demanding than for the (already often challenging) unconstrained case, since individual estimation steps are more demanding (cf. Section 5.7). The second and more important caveat refers to restrictions: Automatic consideration of sub models requires automatic omission of columns from the restriction matrix  $R$ . Some types of restrictions, e.g., non-negativity of all parameters, easily translate to sub models. Other restrictions, however, e.g., the monotonicity restrictions for the parameters of a factor, would lead to nonsensical results if some of the dummy variables for the factor were omitted for a sub model. It is good practice to consider all dummy variables for a factor as an unseparable entity, but there might also be other situations, for which automatic generation of restrictions for sub models by deletion of columns in  $R$  leads to nonsensical results.

## 5. Functionality in R package **ic.infer**

In this section, functionality in R package **ic.infer** is described. Functions `ic.est` (Section 5.3) and `orlm` (Section 5.6) take care of inequality-constrained estimation of normal means or linear model coefficients, respectively. Function `ic.test` (Section 5.4) implements tests of various test problems (TP1 to TP3 and generalized versions of TP1 and TP2). Supplementary functions `ic.weights`, `pchibar` and `pbetabar` (Section 5.5) are used within function `ic.test` for evaluating the null distributions of test statistics. The `summary` method for `orlm` objects (Section 5.6) uses function `ic.test` for implementation of overall tests of model parameters and restrictions. Function `or.relimp` (Section 5.7) decomposes  $R^2$  in the restricted model into contributions of the various regressors. Bootstrapping is implemented in package **ic.infer** through access to package **boot** (Canty and Ripley 2009) and is explained in Section 4.

Notation for constraints in the package uses `ui` for the matrix  $R$  and `ci` for the vector  $r$ .

### 5.1. Example data

Two data examples are used in this section. The first example, taken from Table 1.3.1 in Robertson *et al.* (1988), concerns first-year grade point averages from 2397 Iowa university first-years (available as data frame `grades` in package **ic.infer**) as a function of two ordinal variables with 9 categories each, High-School-Ranking percentiles and ACT Classification<sup>1</sup>. Suppose that an admission policy is to be developed based on these figures. Of course, in order to appear just, an admission policy should be monotone in the sense that admission of a particular person implies that all persons who are better on one criterion and not worse on the other are also admitted. Thus, the predicted function must be monotone in both variables. Using this motivation, Robertson *et al.* (1988) demonstrate isotonic regression on these data. In this article, a two-way analysis of variance without interaction is fit to the data. The unrestricted linear model (cf. below) does contain reversals w.r.t. HSR, where applicants with HSR 41% to 50% would be assessed better than those with HSR 51% to 60%, and similarly applicants with HSR < 20% better than those with HSR 21% to 40%. Note that estimates for the categories of HSR for which unrestricted estimates are reversed are not significantly different from 0. The restricted analyses in Sections 5.3 and 5.6 will restrict parameters for the factor HSR to be monotone. Note that this is an example of a model with a known diagonal (but not identity)  $V_0$ : assuming an unknown positive variance  $\sigma^2$  of the grade points of each student, the variances of the grade means are proportional to the inverse number of students in each class. This can be easily accommodated in function `lm` by using the number of students `n` in the `weights` option (cf. the code below).

```
R> limo.grades <- lm(meanGPA ~ HSR + ACTC, grades, weights = n)
R> summary(limo.grades)
```

Call:

```
lm(formula = meanGPA ~ HSR + ACTC, data = grades, weights = n)
```

Residuals:

Min	1Q	Median	3Q	Max
-2.224	-0.494	-0.149	0.433	1.776

---

<sup>1</sup>ACT is an organization that offers – among other things – college entrance exams in the US; up to 1996, ACT stood for “American College Testing”.

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	1.5009	0.2367	6.34	6.4e-08	***
HSR21-30	-0.1251	0.2540	-0.49	0.62456	
HSR31-40	-0.0272	0.2279	-0.12	0.90533	
HSR41-50	0.1489	0.2131	0.70	0.48796	
HSR51-60	0.0947	0.2077	0.46	0.65059	
HSR61-70	0.3129	0.2055	1.52	0.13419	
HSR71-80	0.4290	0.2044	2.10	0.04092	*
HSR81-90	0.5612	0.2045	2.74	0.00839	**
HSR>=91	0.9703	0.2043	4.75	1.8e-05	***
ACTC13-15	0.2937	0.1625	1.81	0.07662	.
ACTC16-18	0.4565	0.1455	3.14	0.00286	**
ACTC19-21	0.5332	0.1402	3.80	0.00039	***
ACTC22-24	0.6193	0.1391	4.45	4.7e-05	***
ACTC25-27	0.6698	0.1396	4.80	1.5e-05	***
ACTC28-30	0.8223	0.1454	5.66	7.5e-07	***
ACTC31-33	0.9214	0.1846	4.99	7.7e-06	***
ACTC34-36	1.0389	0.4959	2.09	0.04127	*

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.951 on 50 degrees of freedom

Multiple R-squared: 0.908, Adjusted R-squared: 0.878

F-statistic: 30.7 on 16 and 50 DF, p-value: &lt;2e-16

The second example uses a data set from [Kutner \*et al.\* \(2004\)](http://www.ats.ucla.edu/stat/sas/examples/alsm/alsmsasch7.htm) (online also at <http://www.ats.ucla.edu/stat/sas/examples/alsm/alsmsasch7.htm>) that contains observations on 20 females with body fat as the target variable and three explanatory variables all of which can be expected to be associated with an increase in body fat:

- triceps skinfold thickness
- thigh circumference
- mid arm circumference.

These data are analysed as a regression model with all coefficients restricted to be non-negative. This example is similar in spirit to the customer satisfaction applications that instigated development of **ic.infer**, but much smaller, publicly available and included in the package. It also permits to demonstrate application of the simple  $R^2$  decomposition function that is offered within **ic.infer**. The unrestricted linear model estimates for two of the three variables are negative, and in spite of high  $R^2$  and rejection of the overall null hypothesis, no individual coefficient is statistically significant:

```
R> limo.bodyfat <- lm(BodyFat ~ ., bodyfat)
R> summary(limo.bodyfat)
```

Call:

lm(formula = BodyFat ~ ., data = bodyfat)

Residuals:

Min	1Q	Median	3Q	Max
-3.726	-1.611	0.392	1.466	4.128

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )
(Intercept)	117.08	99.78	1.17	0.26
Triceps	4.33	3.02	1.44	0.17
Thigh	-2.86	2.58	-1.11	0.28
Midarm	-2.19	1.60	-1.37	0.19

Residual standard error: 2.48 on 16 degrees of freedom  
 Multiple R-squared: 0.801, Adjusted R-squared: 0.764  
 F-statistic: 21.5 on 3 and 16 DF, p-value: 7.34e-06

## 5.2. Utilities for monotonicity situations

One of the most important special cases of inequality-related setups is the investigation of monotonic behavior of the expectation for a factor with ordered categories. In this subsection, package **ic.infer**'s support for this situation is described.

### *Difference contrasts*

The interpretation of coefficients for factors always depends on the factor coding. In R, default coding for conventional factors (as opposed to ordered factors) is a reference coding with the first factor level being the base category (called `contr.treatment`). For factors declared to be ordered, the default contrasts are polynomial. Alternative contrast codings are, among others, `contr.SAS`, `contr.helmert` and `contr.sum`. Among these, the polynomial and the Helmert coding do not allow simple assessment of monotonicity based on the estimated coefficients, while the others do.

There is one particular factor coding that is not routinely available in R but particularly suitable for assessing monotonicity for factors with ordered levels: each coefficient corresponds to the difference in expectation to the next lower category, implying that monotonicity corresponds to the same sign for all coefficients. The corresponding contrast function `contr.diff` has been implemented in package **ic.infer**.

For illustration, the unconstrained linear model for the grades data is re-calculated with this coding below. The contrast matrix shows that the expectation for the lowest level does not contain any of the coefficients, the expectation for the second level contains the first coefficient, the expectation for the third level the first two coefficients and so forth, until all the eight coefficients are contained in the expectation model for the highest level. The coefficients thus measure the average increase from each level to the next higher one.

```
R> grades.diff <- grades
R> contrasts(grades.diff$HSR) <- "contr.diff"
R> contrasts(grades.diff$ACTC) <- "contr.diff"
R> contrasts(grades.diff$HSR)
```

	21-30-<=20	31-40-21-30	41-50-31-40	51-60-41-50	61-70-51-60	71-80-61-70
<=20	0	0	0	0	0	0
21-30	1	0	0	0	0	0
31-40	1	1	0	0	0	0
41-50	1	1	1	0	0	0
51-60	1	1	1	1	0	0
61-70	1	1	1	1	1	0
71-80	1	1	1	1	1	1
81-90	1	1	1	1	1	1

```

>=91      1      1      1      1      1      1
      81-90-71-80 >=91-81-90
<=20      0      0
21-30      0      0
31-40      0      0
41-50      0      0
51-60      0      0
61-70      0      0
71-80      0      0
81-90      1      0
>=91      1      1

```

```

R> limo.grades.diff <- lm(meanGPA ~ HSR + ACTC, grades.diff, weights = n)
R> summary(limo.grades.diff)

```

Call:

```
lm(formula = meanGPA ~ HSR + ACTC, data = grades.diff, weights = n)
```

Residuals:

```

      Min       1Q   Median       3Q      Max
-2.224 -0.494 -0.149  0.433  1.776

```

Coefficients:

```

              Estimate Std. Error t value Pr(>|t|)
(Intercept)    1.5009    0.2367   6.34 6.4e-08 ***
HSR21-30-<=20  -0.1251    0.2540  -0.49  0.6246
HSR31-40-21-30  0.0978    0.1942   0.50  0.6165
HSR41-50-31-40  0.1761    0.1363   1.29  0.2021
HSR51-60-41-50 -0.0542    0.0990  -0.55  0.5860
HSR61-70-51-60  0.2182    0.0814   2.68  0.0099 **
HSR71-80-61-70  0.1161    0.0719   1.62  0.1123
HSR81-90-71-80  0.1322    0.0655   2.02  0.0489 *
HSR>=91-81-90   0.4091    0.0593   6.90 8.5e-09 ***
ACTC13-15-1-12  0.2937    0.1625   1.81  0.0766 .
ACTC16-18-13-15 0.1628    0.1114   1.46  0.1503
ACTC19-21-16-18 0.0767    0.0759   1.01  0.3168
ACTC22-24-19-21 0.0861    0.0622   1.38  0.1727
ACTC25-27-22-24 0.0505    0.0570   0.89  0.3801
ACTC28-30-25-27 0.1524    0.0653   2.34  0.0236 *
ACTC31-33-28-30 0.0991    0.1329   0.75  0.4591
ACTC34-36-31-33 0.1174    0.4913   0.24  0.8121
---

```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Residual standard error: 0.951 on 50 degrees of freedom

Multiple R-squared: 0.908, Adjusted R-squared: 0.878

F-statistic: 30.7 on 16 and 50 DF, p-value: <2e-16

### *Utility function for creating a monotonicity restriction matrix*

Generally, the restriction matrix  $ui$  has to be tailored to the situation at hand. Depending on the coding of a factor, it can be quite tedious to define the appropriate  $ui$  for hypotheses related to the relation of expectations between factor levels.

For the frequent situation, where monotonicity of factors with several ordered levels is of interest, package **ic.infer** provides the convenience function `make.mon.ui` for creating the appropriate restriction matrix  $ui$ . The function can be used whenever the current coding

permits assessment of monotonicity in a simple way, i.e., for contrasts `contr.treatment` (currently with first category as baseline only), `contr.SAS`, `contr.diff` and `contr.sum`). The output below shows the matrix  $ui$  for two different factor codings: The matrix  $ui$  for the treatment contrasts calculates the first coefficient (=difference of second category to the first (=reference) category) and all differences between coefficients for next higher to next lower level. The matrix  $ui$  for the difference contrasts simply calculates each coefficient. For both codings, monotonicity constraints are of the form  $ui\beta \geq 0$  (or  $-ui\beta \geq 0$  for monotone decrease).

```
R> ui.treat <- make.mon.ui(grades$HSR)
R> ui.treat
```

```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
[1,]     1     0     0     0     0     0     0     0
[2,]    -1     1     0     0     0     0     0     0
[3,]     0    -1     1     0     0     0     0     0
[4,]     0     0    -1     1     0     0     0     0
[5,]     0     0     0    -1     1     0     0     0
[6,]     0     0     0     0    -1     1     0     0
[7,]     0     0     0     0     0    -1     1     0
[8,]     0     0     0     0     0     0    -1     1
```

```
R> ui.diff <- make.mon.ui(grades.diff$HSR)
R> ui.diff
```

```
      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8]
[1,]     1     0     0     0     0     0     0     0
[2,]     0     1     0     0     0     0     0     0
[3,]     0     0     1     0     0     0     0     0
[4,]     0     0     0     1     0     0     0     0
[5,]     0     0     0     0     1     0     0     0
[6,]     0     0     0     0     0     1     0     0
[7,]     0     0     0     0     0     0     1     0
[8,]     0     0     0     0     0     0     0     1
```

Function `make.mon.ui` can also be used for creating the matrix  $ui$  for investigating the monotonicity of a multivariate normal mean without using a linear model based on a factor. In this case, the first argument to `make.mon.ui` is the dimension of the multivariate normal distribution, and `type = "mean"` must be specified. The resulting matrix  $ui$  then calculates differences of neighbouring means:

```
R> make.mon.ui(5, type = "mean")
```

```
      [,1] [,2] [,3] [,4] [,5]
[1,]    -1     1     0     0     0
[2,]     0    -1     1     0     0
[3,]     0     0    -1     1     0
[4,]     0     0     0    -1     1
```

### 5.3. Estimation

Function `ic.est` for inequality-constrained estimation of normal means uses the routine `solve.QP` from R-package **quadprog** to determine the constrained estimate. It is possible to declare the first few rows of the restrictions  $ui\mu \geq ci$  to be equality restrictions (via the

parameter `meq`). It has been pointed out above that estimation of  $\beta$  in the restricted linear model is equivalent to estimation of  $\beta$  based on the multivariate normal distribution of the unrestricted estimate  $\hat{\beta}$ . Thus, we can illustrate function `ic.est` using the estimates from one of the linear models above. For example, one can estimate the coefficients of the factor `HSR` with treatment contrasts under the restriction of non-decreasing behavior, i.e.,  $\beta_1 \geq 0, \beta_2 - \beta_1 \geq 0, \dots, \beta_9 - \beta_8 \geq 0$  (contrast matrix `ui.treat` defined in 5.2.2):

```
R> HSRmon <- ic.est(coef(limo.grades)[2:9], ui = ui.treat, Sigma = vcov(limo.grades)[2:9,
+ 2:9])
R> HSRmon
```

Constrained estimate:

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
0.0000	0.0492	0.1918	0.1918	0.3892	0.5055	0.6377	1.0469

It is also possible to indicate that the first few restrictions (number given by option `meq`) are equality restrictions. For example, the code below declares that the first three restrictions are equality instead of inequality restrictions:

```
R> HSRreq <- ic.est(coef(limo.grades)[2:9], ui = ui.treat, Sigma = vcov(limo.grades)[2:9,
+ 2:9], meq = 3)
R> HSRreq
```

Constrained estimate:

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
0.000	0.000	0.000	0.038	0.256	0.372	0.505	0.914

A summary-function on objects of class `orest` – as generated by function `ic.est` – gives more detailed information, showing also the restrictions, which of them are active, and indicating which estimates are subject to a restriction (regardless whether active or not). For the monotonicity-restricted estimate, we get

```
R> summary(HSRmon)
```

Order-restricted estimated mean with restrictions of coefficients of

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
----------	----------	----------	----------	----------	----------	----------	---------

Inequality restrictions:

	HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91		
1: A	1	0	0	0	0	0	0	0	%%colnames	>= 0
2:	-1	1	0	0	0	0	0	0	%%colnames	>= 0
3:	0	-1	1	0	0	0	0	0	%%colnames	>= 0
4: A	0	0	-1	1	0	0	0	0	%%colnames	>= 0
5:	0	0	0	-1	1	0	0	0	%%colnames	>= 0
6:	0	0	0	0	-1	1	0	0	%%colnames	>= 0
7:	0	0	0	0	0	-1	1	0	%%colnames	>= 0
8:	0	0	0	0	0	0	-1	1	%%colnames	>= 0

Note: Restrictions marked with A are active.

Restricted estimate:

R	HSR21-30	R	HSR31-40	R	HSR41-50	R	HSR51-60	R	HSR61-70	R	HSR71-80	R	HSR81-90	R	HSR>=91
	0.00000		0.04917		0.19181		0.19181		0.38920		0.50551		0.63772		1.04688

Note: Estimates marked with R are involved in restrictions.

While it would be possible to determine the estimate even for linearly dependent rows of the constraint matrix  $R$ , this is not permitted in package **ic.infer** – if the package encounters linearly dependent rows in **ui** (the package notation for  $R$ ), it aborts with an error message that suggests a subset of independent rows of **ui**.

## 5.4. Hypothesis testing

Package **ic.infer** implements the likelihood ratio tests for test problems **TP1**, **TP2**, and **TP3** in function **ic.test**. The principal argument to function **ic.test** is an object of class **orest** as output by function **ic.est**; an object of class **orlm** output by function **orlm** can also be processed, since it inherits from class **orest**. Among other things, the input object contains information on the restrictions that were used for estimation. The type of test problem is indicated to function **ic.test** via option **TP**. **TP = 1**, **TP = 2**, and **TP = 3** refer to the test problems introduced in Section 3.1. Three extensions of these problems are additionally implemented:

- For **TP = 1** and **TP = 2**, the first few restrictions can be declared equality instead of inequality restrictions – this is implemented in function **ic.test** through access to the **meq**-element of the input object. This modification requires different calculation of the weights for the null distributions of the test statistics: these weights depend on the conditional covariance matrix given the equality constraints are true, cf. Shapiro (1988, formula (5.9)) and Section 5.5. The test statistics continue to be given by (8), (9) or their modification for unknown  $\sigma^2$  (11), but with  $\hat{\mu}^*$  observing equality- and inequality restrictions.
- Additional equality restrictions can be included in the null hypothesis of **TP1**. For these, the alternative hypothesis is not directional. This test problem is implemented in the package as **TP = 11**, and the additional restrictions are handed to function **ic.test** through arguments **ui0.11** and **ci0.11**. **TP = 11** is, for example, used in the summary function for class **orlm**, when testing the null hypothesis that all coefficients except the intercept are 0 in the presence of constraints  $R\beta \geq 0$  that do not affect all elements of  $\beta$ . Again, the test statistic for this test problem is already given as (8) or its modification (11) above by making  $\hat{\mu}_=$  observe the additional equality restrictions as well. Here, the weights are the same as without equality restrictions, but the degrees of freedom of the distributions in the mixture need to be adjusted.
- Some equality restrictions can be maintained in the alternative hypothesis of **TP2**. This is implemented as **TP = 21** using option **meq.alt**, which indicates the number of the first few equality-restrictions that are to be maintained under the alternative hypothesis. **meq.alt** must not be larger than the **meq**-element of the input object of function **ic.test**. Here, the test statistic (9) (or (11)) has to use the restricted estimated under the maintained equality restrictions  $\hat{\mu}_{=,alt}$  instead of  $y$ .

A few examples are shown below. First, the equality- and inequality-restricted estimate **HSReq** of the HSR coefficients is subjected to a test of type **TP1**. We see that equality of all restrictions is clearly rejected; note that option **brief=FALSE** requests detailed information on constraints that is not shown per default.

```
R> summary(ic.test(HSReq), brief = FALSE)
```

Order-related hypothesis test:

Type 1 Test:

H0: all restrictions active(=)  
vs.

H1: at least one restriction strictly true (>)

Test statistic            p-value  
                         215            <0.0001

	HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91	
1: A 1	0	0	0	0	0	0	0	0	***colnames == 0
2: A -1	1	0	0	0	0	0	0	0	***colnames == 0
3: A 0	-1	1	0	0	0	0	0	0	***colnames == 0
4: 0	0	-1	1	0	0	0	0	0	***colnames >= 0
5: 0	0	0	-1	1	0	0	0	0	***colnames >= 0
6: 0	0	0	0	-1	1	0	0	0	***colnames >= 0
7: 0	0	0	0	0	-1	1	0	0	***colnames >= 0
8: 0	0	0	0	0	0	-1	1	0	***colnames >= 0

Restricted estimate under H0:

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
0	0	0	0	0	0	0	0

Restricted estimate under union of H0 and H1 :

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
0.000	0.000	0.000	0.038	0.256	0.372	0.505	0.914

Now we test the null hypothesis that restrictions hold vs. the alternative that they are violated. We see that this null hypothesis is not rejected, i.e., the data do not provide proof that these restrictions are not all true.

```
R> summary(ic.test(HSReq, TP = 2))
```

Order-related hypothesis test:

Type 2 Test:

H0: all restrictions true(>=)  
vs.

H1: at least one restriction violated (<)

Test statistic            p-value  
                         3.36            0.7948

Restricted estimate under H0:

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
0.000	0.000	0.000	0.038	0.256	0.372	0.505	0.914

Unrestricted estimate:

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
-0.1251	-0.0272	0.1489	0.0947	0.3129	0.4290	0.5612	0.9703

The next test operates on all coefficients of the analysis of variance model from Section 5.1. This TP = 11-type test tests the null hypothesis that all coefficients except for the intercept are zero vs. the alternative that the HSR coefficients follow the restriction outlined above, i.e., coefficients in positions 2 to 9 of the coefficient vector (index = 2:9) follow the indicated restrictions, while all other coefficients are free. Here, the null hypothesis is again clearly rejected.

```
R> HSReq.large <- ic.est(coef(limo.grades), ui = ui.treat, Sigma = vcov(limo.grades),
+   index = 2:9, meq = 3)
R> summary(ic.test(HSReq.large, TP = 11, ui0.11 = cbind(rep(0, 16), diag(1, 16))))
```

Order-related hypothesis test:

Type 11 Test:

H0: all original restrictions active plus additional equality restrictions  
vs.

H1: original restrictions hold

Test statistic	p-value
488	<0.0001

Restricted estimate under union of H0 and H1 :

(Intercept)	HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80
1.552	0.000	0.000	0.000	0.038	0.256	0.372
HSR81-90	HSR>=91	ACTC13-15	ACTC16-18	ACTC19-21	ACTC22-24	ACTC25-27
0.505	0.914	0.300	0.467	0.535	0.624	0.676
ACTC28-30	ACTC31-33	ACTC34-36				
0.827	0.927	1.044				

Restricted estimate under H0:

(Intercept)	HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80
2.63	0.00	0.00	0.00	0.00	0.00	0.00
HSR81-90	HSR>=91	ACTC13-15	ACTC16-18	ACTC19-21	ACTC22-24	ACTC25-27
0.00	0.00	0.00	0.00	0.00	0.00	0.00
ACTC28-30	ACTC31-33	ACTC34-36				
0.00	0.00	0.00				

The last example demonstrates  $TP = 21$ : the null hypothesis has three equality restrictions (estimate object `HSReq`), and the first two of these are maintained for the alternative hypothesis (`meq.alt=2`). Note that – as the alternative is unrestricted apart from the first two equality restrictions – a reversal occurs in the estimate under the alternative hypothesis. Nevertheless, like for  $TP = 2$ , the validity of the restrictions is not rejected.

```
R> summary(ic.test(HSReq, TP = 21, meq.alt = 2))
```

Order-related hypothesis test:

Type 21 Test:

H0: all restrictions true(>= or =)  
vs.

H1: at least one restriction violated (<), some =-restrictions maintained

Test statistic	p-value
3.03	0.6134

Restricted estimate under H0:

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
0.000	0.000	0.000	0.038	0.256	0.372	0.505	0.914

Restricted estimate under H1:

HSR21-30	HSR31-40	HSR41-50	HSR51-60	HSR61-70	HSR71-80	HSR81-90	HSR>=91
0.000	0.000	0.198	0.144	0.362	0.478	0.610	1.020

### 5.5. Calculation of weights and $p$ values for the test problems

Function `ic.weights` calculates the mixing weights for a given covariance matrix, using the probabilities for certain faces of the cone as derived in Section 2.4. Since it is known that even and odd weights sum to 0.5 each (cf. e.g., [Silvapulle and Sen 2004](#), Proposition 3.6.1, Number 3), the two most demanding weights (in terms of most summands in (7)) can always be inferred as the difference of 0.5 to the sum of the other even or odd weights. Even exploiting this possibility, calculation of weights remains computer-intensive for large covariance matrices; for example, it takes about 9 seconds CPU time for a matrix with dimension 10, and already 1265 seconds (about 21 minutes) for a matrix with dimension 15.

Orthant probabilities that are needed for the weights according to (7), are calculated using package `mvtnorm` by Monte-Carlo methods, i.e., the weights are subject to slight variation. Because of numerical inaccuracies, it is even possible that calculated  $p$  values become slightly negative. Printing and summary functions of package `ic.infer` report all  $p$  values below 0.0001 as “<0.0001”, since more accuracy should normally not be needed.

For the test problems implemented in function `ic.test`, choice of the covariance matrices for obtaining the weights follows the formulae by [Shapiro \(1988\)](#), based on the `meq`-, the `ui`-, and the `Sigma`-element of the input object: Whenever `meq=0`, the covariance matrix to use is `ui%*%Sigma%*%t(ui)` (assuming that `ui` has  $p$  columns if the data are  $p$ -dimensional, otherwise think of `ui` as suitably enlarged by zero columns (`uiw` in package code)). If `meq>0`, the conditional covariance of the last  $m - \text{meq}$  rows given the first `meq` rows of `ui%*%y` must be used instead for calculation of mixing weights (formula (5.9) in [Shapiro 1988](#)).

Degrees of freedom corresponding to the weights depend on the test problem at hand and are determined in function `ic.test`, if not provided by the user. Functions `pchibar` and `pbetabar` calculate  $p$  values from given vectors of weights and degrees of freedom. Function `pchibar` has been taken from package `ibdreg` by [Sinnwell and Schaid \(2007\)](#), and function `pbetabar` has been analogously defined.

### 5.6. Estimation in the linear model

Function `orlm` uses the other functions in package `ic.infer` for providing a convenient overall analysis of order-restricted linear models. Starting from an unconstrained linear model object (class `lm`) or a covariance matrix of response (first position) and all regressors, the function determines the constrained estimate,  $R^2$  for the constrained model and – if requested – bootstraps the estimates of coefficients (the latter is valid only in the implemented case of uncorrelated errors and of course only possible if the input is a linear model with embedded data).

#### *Postprocessing the output object*

The output object of class `orlm` can be processed with several S3 methods provided in package `ic.infer`: A `plot` method provides a residual plot, a `print` method gives a brief printout, and a `summary` method gives a more extensive overview on the object, involving bootstrap confidence intervals and overall model and restriction tests, if not suppressed; tests can be suppressed because their calculation may take up substantial time in case of many restrictions because of calculation of weights, cf. also the previous subsection. Furthermore, a `coef` method extracts the coefficients from the object. In addition to these specially-defined methods, some general

methods for model objects do also work: functions `fitted` and `residuals` provide fitted values and residuals. Other methods for class `lm` (`predict`, `effects`, `vcov`) do not work on `orlm` objects. Note that model diagnostics cannot be simply transferred to restricted models, as the restricted estimation modifies the distributional properties of the residuals in not easily foreseeable ways. The `plot` method only provides a simple plot of raw residuals vs. fitted values, as it is not even possible to standardize the residuals. Further research might improve the availability of diagnostics on the restricted model. As long as this has not been conducted, model diagnostics, e.g., for normality, can be done on the unrestricted model, which is of course still valid even though it does not exploit the prior knowledge about a restriction.

### *Linear model analysis for the two example data sets*

For the grades data, with two ordinal factors, restricting only HSR (because ACTC is automatically in the correct order; indicated by `index=2:9` for the position of HSR-coefficients in the overall coefficient vector) function `orlm` works as follows (contrast matrix `ui.treat` defined in 5.2.2):

```
R> orlimo.grades <- orlm(limo.grades, ui = ui.treat, index = 2:9)
R> summary(orlimo.grades, brief = TRUE)
```

Order-restricted linear model with restrictions of coefficients of  
HSR21-30 HSR31-40 HSR41-50 HSR51-60 HSR61-70 HSR71-80 HSR81-90 HSR>=91

Coefficients from order-restricted model:

(Intercept)	R HSR21-30	R HSR31-40	R HSR41-50	R HSR51-60	R HSR61-70	R HSR71-80
1.42444	0.00000	0.04917	0.19181	0.19181	0.38920	0.50551
R HSR81-90	R HSR>=91	ACTC13-15	ACTC16-18	ACTC19-21	ACTC22-24	ACTC25-27
0.63772	1.04688	0.29496	0.45714	0.53311	0.61918	0.66945
ACTC28-30	ACTC31-33	ACTC34-36				
0.82222	0.92091	1.03868				

Note: Coefficients marked with R are involved in restrictions.

Hypothesis tests ( 50 error degrees of freedom ):

Overall model test under the order restrictions:

Test statistic: 0.9075, p-value: <0.0001

Type 1 test: H0: all restrictions active(=)

vs. H1: at least one restriction strictly true (>)

Test statistic: 0.8132, p-value: <0.0001

Type 2 test: H0: all restrictions true

vs. H1: at least one restriction false

Test statistic: 0.01074, p-value: 0.9887

Type 3 test: H0: at least one restriction false or active (=)

vs. H1: all restrictions strictly true (>)

Test statistic: -0.5481, p-value: 0.7070

Type 3 test based on t-distribution (one-sided),

all other tests based on mixture of beta distributions

Option `brief` suppresses information on restrictions (that has been shown in Section 5.3). For this example,  $R^2$  is only slightly reduced by introducing the restriction, and the estimates

(not surprisingly) coincide with those from Section 5.3. The overall model test and the test of TP1 clearly reject their respective null hypothesis, while the data are compatible with validity of the restriction according to the test for TP2 but do not prove strict validity of the inequality restriction (TP3).

The grades example has been about analysis of variance and has worked with aggregated data, which makes bootstrapping useless. The rest of this section uses the body fat data for illustrating functionality for order-restricted regression, including bootstrap confidence intervals:

```
R> orlimo.bodyfat <- orlm(limo.bodyfat, ui = diag(1, 3), boot = TRUE)
R> summary(orlimo.bodyfat)
```

Order-restricted linear model with restrictions of coefficients of  
Triceps Thigh Midarm

```
Inequality restrictions:
      Triceps Thigh Midarm
1:    1      0      0      %%%colnames >= 0
2:    0      1      0      %%%colnames >= 0
3: A 0      0      1      %%%colnames >= 0
```

Note: Restrictions marked with A are active.

Restricted model: R2 reduced from 0.8014 to 0.778

```
Coefficients from order-restricted model
with 95 pct bootstrap confidence intervals( perc ):
      Coeff.   Lower   Upper
(Intercept) -19.1742 -32.7604  -3.2911
R Triceps    0.2224   0.0000   0.9749
R Thigh      0.6594   0.0000   0.9479
R Midarm     0.0000   0.0000   0.3024
```

Note: Coefficients marked with R are involved in restrictions.

Hypothesis tests ( 16 error degrees of freedom ):

Overall model test under the order restrictions:

Test statistic: 0.7966, p-value: <0.0001

Type 1 test: H0: all restrictions active(=)

vs. H1: at least one restriction strictly true (>)

Test statistic: 0.7966, p-value: <0.0001

Type 2 test: H0: all restrictions true

vs. H1: at least one restriction false

Test statistic: 0.105, p-value: 0.4100

Type 3 test: H0: at least one restriction false or active (=)

vs. H1: all restrictions strictly true (>)

Test statistic: -1.370, p-value: 0.9052

Type 3 test based on t-distribution (one-sided),  
all other tests based on mixture of beta distributions

Again,  $R^2$  is not dramatically reduced, the overall test – in this case identical to the test

for **TP1** – is clearly significant, while the other two tests do not reject their null hypothesis. While the unrestricted model had two negative estimated coefficients, the restricted model has one active restriction. The other previously negative coefficient has now been estimated to be positive. Note that still none of the individual coefficients is significantly different from 0, since all bootstrap confidence intervals include this boundary value.

### *Bootstrapping regression models*

Confidence intervals in **ic.infer** are obtained via the bootstrap. The implemented bootstrap is valid for uncorrelated observations only, since observations are independently sampled. When bootstrapping regression models, there are two principally different reasonable approaches (cf. e.g. [Davison and Hinkley 1997](#); [Fox 2002](#)): The regressors can be considered fixed in some situations, e.g., for experimental data. In this case, only the error terms are random. Contrary, in observational studies, like e.g., customer satisfaction surveys, it makes far more sense to consider also the regressors as random, since the observations are a random sample from a larger population. These two scenarii prompt two different approaches for bootstrapping: For fixed regressors, bootstrapping is based on repeated sampling from the residuals of the regression model, while for random regressors, the complete observation rows – consisting of regressors and response – are resampled. **ic.infer** offers both possibilities, defaulting to random regressors (`fixed = FALSE`). Bootstrapping in **ic.infer** is implemented in function `orlm` based on the function `boot` from R package **boot**. Bootstrap confidence intervals are then calculated by the summary method for the output object from function `orlm`, relying on function `boot.ci` of package **boot**. Percentile intervals, BCa intervals, normal intervals and basic intervals are supported (default: percentile intervals). For further information on bootstrapping in general, cf. e.g., [Davison and Hinkley \(1997\)](#).

### *Overall tests*

As mentioned above and shown in the example output, the summary method for objects of class `orlm` calculates an overall model test, similar to the overall  $F$  test in the unconstrained linear model, and several tests for or against the restrictions. These can be suppressed, because their calculation can be very time-consuming in case of large sets of restrictions.

If they are not suppressed, function `summary.orlm` calculates an overall test that all parameters except the intercept are 0 ( $H_0$ ) vs. the restriction set (this is a test of type **TP** = 11 or **TP** = 1, depending on whether or not the original restrictions refer to all parameters in the model). In addition, all tests for the three test problems **TP1** to **TP3** are calculated. (Test problem **TP3** is only applicable if there are no equality restrictions (i.e., `meq=0`).) Note that the time-consuming aspect is calculation of weights for the null distributions of test statistics. These are calculated only once and are then handed to function `ic.test` for the further tests. Nevertheless, calculation of weights for large problems takes a long time or is even impossible because of storage space restrictions.

It would be desirable to have a function for sequential testing of sources, analogous to `anova`, for order-restricted linear models. However, this would require the possibility to test a cone-shaped null hypothesis vs. a larger cone-shaped alternative hypothesis, which is far from trivial. So far, it has not been figured out how to implement such a test.

### 5.7. $R^2$ decomposition

It has been mentioned earlier that function `or.relimp` decomposes  $R^2$  into contributions of individual regressors. The method is implemented by handing the  $2^p$ -vector of  $R^2$  values for all sub-models to function `Shapley.value` from R package **kappalab** (Grabisch, Kojadinovic, and Meyer 2009). The result is illustrated for the body fat example:

```
R> or.relimp(limo.bodyfat, ui = diag(1, 3))
```

```
Triceps    Thigh    Midarm
0.354115 0.416395 0.007542
```

Note that – in this example – although the coefficients are quite different from those of the unrestricted model, the  $R^2$  decomposition is very similar (**relaimpo** must be loaded for the following calculation):

```
R> calc.relimp(limo.bodyfat)$lmg
```

```
Triceps    Thigh    Midarm
0.37439 0.39914 0.02782
```

So far, such similarity has been observed for all examples for which the restrictions employed were plausible and adequate.

It has been mentioned in Section 5.3 that automatic generation of restrictions for sub models is naturally done by deleting the respective columns from the restriction matrix ( $R$  or  $ui$ , respectively). It is emphasized here once more that this is not adequate for all conceivable situations. *It is in the responsibility of the user to ensure that restrictions for sub models are sensible and meaningful.*

Decomposition of  $R^2$  requires calculation of  $2^p$  *constrained* estimates. This involves significantly higher computational burden than for the unconstrained case: For example, calculations on a 2.4GHz Dual Core Windows XP machine in `calc.relimp` took 0.5 seconds for 10 regressors, about 17 seconds for 15 regressors and about 580 seconds for 20 regressors. For the same scenarios, calculations in `or.relimp` with all non-intercept coefficients restricted to be non-negative took 2.5 seconds for 10 regressors, about 109 seconds for 15 regressors, and about 14800 seconds for 20 regressors. In case of fewer restrictions than regressors, computing time is somewhat reduced; for example, when restricting only 10 of the 15 coefficients in the 15 regressor situation, `or.relimp` computing time was about 90 seconds. Given that unconstrained models gave very similar  $R^2$  decompositions in all reasonable applications that have so far been examined, decompositions from unconstrained models may very well be used at least as first checks.

## 6. Final remarks

Inequality-constrained inference and its implementation in R package **ic.infer** have been explained and illustrated in this article. While **ic.infer** offers the most important possibilities for normal means and linear models, some wishes remain to be fulfilled with future developments. These will be discussed below.

Within the linear model context, it would be desirable to implement some factor-related functionality for function `orlm`, supporting e.g., an overall test of significance for a factor as

a whole or hypothesis tests corresponding to sequential analysis of variance (analogously to function `anova`). It has been mentioned before that these topics may prove difficult because they will often require testing a cone-shaped null hypothesis within a larger cone-shaped alternative. Their feasibility will be investigated, and even if not all situations can be covered, some may prove feasible (e.g., no restrictions on the factor, inequality restrictions on the factor but on nothing else, ...).

For non-linear models with asymptotically normal parameter estimates, users can apply inequality-restricted inference on the coefficients through functions `ic.est` or `ic.test`. A more direct approach would be desirable. It is intended to extend coverage of the package to (selected) non-normal situations with linear equality and inequality restrictions, for which it is known that the asymptotic distribution of the likelihood ratio test statistic is also a mixture of  $\chi^2$  distributions (cf. section 4 of [Silvapulle and Sen 2004](#)). Of course, inference is local and less robust, if we leave the linear model.

Calculation of weights is a computational road block in case of many restrictions. It will be explored if direct calculation of weights using Monte-Carlo methods is more efficient than using Equation (7) together with package `mvtnorm`.

Function `or.relimp` is currently restricted to linear models without factors. It would be possible to include factors by grouping their dummies, like in `relaimpo`. Also, it might be possible to enable usage of `or.relimp` for larger problems than currently possible by a different programming approach – however, as long as no reasonable examples have been encountered for which constrained and unconstrained decompositions make a relevant difference, improvements on `or.relimp` have low priority.

## Acknowledgments

My colleague Frank Hausser helped overcome initial numerical problems. The package uses code by John Fox (functions `RREF` and `GaussianElimination`), Wolfgang Huber (function `nchoosek`), and [Sinnwell and Schaid \(2007\)](#), function `pchibar`). A referee made very useful comments that improved both the paper and the software.

## References

- Aptech Systems, Inc (2009). *GAUSS 10.0 User Guide*. Black Diamond WA.
- Bartholomew D (1961). “Ordered Tests in the Analysis of Variance.” *Biometrika*, **48**, 325–332.
- Berger R (1989). “Uniformly More Powerful Tests for Hypotheses Concerning Linear Inequalities and Normal Means.” *Journal of the American Statistical Association*, **84**, 192–199.
- Broberg P (2009). *SAGx: Statistical Analysis of the GeneChip*. R package version 1.9-7, URL <http://www.bioconductor.org/packages/2.1/bioc/html/SAGx.html>.
- Canty A, Ripley B (2009). *boot: Bootstrap R (S-Plus) Functions*. R package version 1.2-41, URL <http://CRAN.R-project.org/package=boot>.
- Chevan A, Sutherland M (1991). “Hierarchical Partitioning.” *The American Statistician*, **45**, 90–96.

- Chu S (2001). “Pricing the C’s of Diamond Stones.” *Journal of Statistics Education*, **9**.
- Cohen A, Kemperman H, Sackrowitz J (2000). “Properties of Likelihood Inference for Order-Restricted Models.” *Journal of Multivariate Analysis*, **72**, 50–77.
- Davison A, Hinkley D (1997). *Bootstrap Methods and Their Application*. Cambridge University Press, Cambridge.
- Fox J (2002). “Bootstrapping Regression Models.” In *An R and S-PLUS Companion to Applied Regression: A Web Appendix to the Book*. Sage, Thousand Oaks, CA. URL <http://cran.r-project.org/doc/contrib/Fox-Companion/appendix-bootstrapping.pdf>.
- Genz A, Bretz F, Miwa T, Mi X, Leisch F, Scheipl F, Hothorn T (2009). *mvtnorm: Multivariate Normal and t Distributions*. R package version 0.9-8, URL <http://CRAN.R-project.org/package=mvtnorm>.
- Gleser L (1973). “On a Theory of Intersection Union Tests (Abstract).” *Institute of Mathematical Statistics Bulletin*, **2**, 233–233.
- Gourieroux C, Monfort A (1995). *Statistics and Econometric Models, Vol.II*. Cambridge University Press, Cambridge.
- Grabisch M, Kojadinovic I, Meyer P (2009). *kappalab: Non-Additive Measure and Integral Manipulation Functions*. R package version 0.4-4, URL <http://CRAN.R-project.org/package=kappalab>.
- Grömping U (1996). *Tests for a Monotone Dose-Response Relation in Models With Ordered Categorical Dose with Emphasis on Likelihood Ratio Tests for Linear Inequalities on Normal Means*. Ph.D. thesis, Department of Statistics, Dortmund University, Germany.
- Grömping U (2006). “Relative Importance for Linear Regression in R: The Package **relaimpo**.” *Journal of Statistical Software*, **17**(1), 1–27. URL <http://www.jstatsoft.org/v17/i01/>.
- Grömping U (2007). “Estimators of Relative Importance in Linear Regression Based on Variance Decomposition.” *The American Statistician*, **61**, 139–147.
- Grömping U (2010). *ic.infer: Inequality Constrained Inference in Linear Normal Situations*. R package version 1.1-3, URL <http://CRAN.R-project.org/package=ic.infer>.
- Grömping U, Landau S (2009). “Do Not Adjust Coefficients in Shapley Value Regression.” *Preprint*, Applied Stochastic Models in Business and Industry. doi:10.1002/asmb.773. Online early view.
- Kruskal W (1987). “Relative Importance by Averaging Over Orderings.” *The American Statistician*, **41**, 6–10.
- Kudô A (1963). “A Multivariate Analogue of the One-Sided Test.” *Biometrika*, **50**, 403–418.
- Kutner M, Nachtsheim C, Neter J (2004). *Applied Linear Statistical Models*. 4th edition. McGraw-Hill, Boston.
- Lindeman R, Merenda P, Gold R (1980). *Introduction to Bivariate and Multivariate Analysis*. Scott, Foresman, Glenview IL.

- Lipovetsky S, Conklin M (2001). “Analysis of Regression in Game Theory Approach.” *Applied Stochastic Methods in Business and Industry*, **17**, 319–330.
- Liu H, Berger R (1995). “Uniformly More Powerful One-Sided Tests for Hypotheses About Linear Inequalities.” *The Annals of Statistics*, **23**, 55–72.
- Perlman M, Wu L (1999). “The Emperor’s New Tests.” *Statistical Science*, **14**, 355–381.
- Perlman M, Wu L (2003). “The Validity of the Likelihood Ratio and Maximum Likelihood Methods.” *Journal of Statistical Planning and Inference*, **117**, 59–81.
- R Development Core Team (2009). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.
- Robertson T, Wright F, Dykstra R (1988). *Order-Restricted Inference*. John Wiley & Sons, New York.
- SAS Institute, Inc (2009). *SAS/ETS User’s Guide*. Cary, NC.
- Sasabuchi S (1980). “A Test of a Multivariate Normal Mean With Composite Hypotheses Determined by Linear Inequalities.” *Biometrika*, **67**, 429–439.
- Shapiro A (1988). “Towards a Unified Theory in Inequality-Constrained Testing in Multivariate Analysis.” *International Statistical Review*, **56**, 49–62.
- Silvapulle M (1997). “A Curious Example Involving the Likelihood Ratio Test Against One-Sided Hypotheses.” *The American Statistician*, **51**, 178–180.
- Silvapulle M, Sen P (2004). *Constrained Statistical Inference*. John Wiley & Sons, New York.
- Sinnwell J, Schaid D (2007). *ibdreg: Regression Methods for IBD Linkage With Covariates*. R package version 0.1.1, URL <http://CRAN.R-project.org/package=ibdreg>.
- Turlach B, Weingessel A (2007). *quadprog: Functions to solve Quadratic Programming Problems*. R package version 1.4-11, URL <http://CRAN.R-project.org/package=quadprog>.
- Walsh C, Mac Nally R (2008). *hier.part: Hierarchical Partitioning*. R package version 1.0-3, URL <http://CRAN.R-project.org/package=hier.part>.

### Affiliation:

Ulrike Grömping  
 Department II – Mathematics, Physics, Chemistry  
 BHT Berlin – University of Applied Sciences  
 D-13353 Berlin  
 E-mail: [groemping@bht-berlin.de](mailto:groemping@bht-berlin.de)  
 URL: <http://prof.tfh-berlin.de/groemping/>