UNIVERSIDAD CARLOS III DE MADRID working papers

Working Paper 14
Business Economic Series 05
November 2012
(revised version of June 2012)

Departamento de Economía de la Empresa
Universidad Carlos III de Madrid
Calle Madrid, 126
28903 Getafe (Spain)
Fax (34-91) 6249607

RECONSIDERING OPTIMAL EXPERIMENTAL DESIGN FOR CONJOINT ANALYSIS[*]

Mercedes Esteban-Bravo[1], Agata Leszkiewicz[2], and José M. Vidal-Sanz [3]

Abstract

The quality of Conjoint Analysis estimations heavily depends on the alternatives presented in the experiment. An efficient selection of the experiment design matrix allows more information to be elicited about consumer preferences from a small number of questions, thus reducing experimental cost and respondent's fatigue. The statistical literature considers optimal design algorithms (Kiefer, 1959), and typically selects the same combination of stimuli more than once. However in the context of conjoint analysis, replications do not make sense for individual respondents. In this paper we present a general approach to compute optimal designs for conjoint experiments in a variety of scenarios and methodologies: continuous, discrete and mixed attributes types, customer panels with random effects, and quantile regression models. We do not compute good designs, but the best ones according to the size (determinant or trace) of the information matrix of the associated estimators without repeating profiles as in Kiefer's methodology. We handle efficient optimization algorithms to achieve our goal, avoiding the use of widespread ad-hoc intuitive rules.

Keywords: Conjoint Analysis, Optimal experimental designs, Optimization.

[*] Research funded by two research projects, S-0505/TIC-0230 by the Comunidad de Madrid and ECO2011-30198 by MICINN agency of Spanish Government.

[1] Mercedes Esteban-Bravo, Department of Business Administration, University Carlos III de Madrid, C/ Madrid 126, 28903 Getafe (Madrid), Spain; tel: +34 91 624 8921; fax: +34 91 624 8921; e-mail: mesteban@emp.uc3m.es

[2] Agata Leszkiewicz, Department of Business Administration, University Carlos III de Madrid, C/ Madrid 126, 28903 Getafe (Madrid), Spain; tel: +34 91 624 8921; fax: +34 91 624 8921; e-mail: agata.leszkiewicz@uc3m.es

[3] Jose M. Vidal-Sanz, Department of Business Administration, University Carlos III de Madrid, C/ Madrid 126, 28903 Getafe (Madrid), Spain; tel: +34 91 624 8642; fax: +34 91 624 9607; e-mail: jvidal@emp.uc3m.es

1 Introduction

Since the seminar paper of Green and Rao (1971), Conjoint Analysis (CA) has become a widespread marketing research tool for marketing scholars and practitioners (see e.g. Cattin and Wittink 1982; Wittink and Cattin 1989). CA encompasses a variety of techniques designed to analyze consumer preferences over multi-attributed products, estimating preference trade-offs between attributes from experimental data. Respondents are presented with a series of stimuli (product/service descriptions, illustrations, samples, prototypes etc.), and are asked to rank or rate them (metric or "classic" CA), or choose one from the shown set of profiles (choice-based CA). We focus on the classic CA, which considers a regression model

$$y_t = f\left(x_t\right)' \beta + \epsilon_t, \qquad t = 1, ..., T,$$

with compensatory, linear-in-parameters utility function $U(x_t) = f\left(x_t\right)' \beta$, where the response y_t is a utility ranking or a rating (measured either on a 0 to 100 attitude scale, a purchase probability scale, a strongly disagree to strongly agree scale, or some similar scale). Product profile x_t is a $k \times 1$ vector of deterministic regressors in a compact set χ in an Euclidean space representing attributes (discrete dummy and/or continuous variables), and sometimes other contextual block variables. f is a known continuous mapping from χ to $\mathbb{R}^p$ whose coordinates are linearly independent and may include an intercept, discrete interactions (products of dummies), or product of continuous regressors (to define multivariate polynomials similarly to surface response models). Function f could also have a known local maxima (self-explicated ideal point). We also allow $p < k$, if $f\left(x_t\right)$ is a projection of x_t on a linear space of smaller dimension. The vector β is a $p \times 1$ vector of unknown parameters. The errors ϵ_t are regarded as mutually independent random shocks, satisfying $E\left[\epsilon_t\right] = 0$, $E\left[\epsilon_t^2\right] = \sigma^2$. The coefficients β are estimated from an experimental setting using the OLS method. For a literature review and description of the methods and common CA applications, see Gustafsson, Herrmann, and Huber (2007). For a discussion of some problem areas in current CA methods, see Bradlow (2005) and Netzer et al. (2008).

In a matrix notation the model is $y = X\beta + \epsilon$, where y and ϵ are $T \times 1$ vectors, $X = [f(x_1)', \ldots, f(x_t)']'$ is a $T \times p$ design matrix, with row t containing $f(x_t)'$, and $x = [x_1', \ldots, x_T']'$. The classical theory considers $T \geq p$ independent observations, and it is assumed that $E(\epsilon) = 0$, $E(\epsilon\epsilon') = \sigma^2 I_T$, $rank(X) = p$. Then the OLS estimator $\hat{\beta} = (X'X)^{-1}X'y$ is unbiased, with non-singular variance

$$Var\left(\hat{\beta}\right) = \sigma^2 \left(X'X\right)^{-1}.$$

Often, there are linear identities across the attributes which cause the $(X'X)$ matrices to be singular. For example, for continuous regressors this occurs when we consider compositional data (proportions of several ingredients) and an intercept (the sum of the proportions is identical to the intercept variable), and also when we have discrete dummies with an intercept. In these cases, the model is usually reformulated (e.g., omitting a regressor). We will assume that the necessary operations to eliminate collinearities have been already implemented in the considered formulation.

If the ratings were normally distributed we could perform inference analysis with a small T. But generally, this is not the case. If the deterministic matrix $Q_T = (X'X/T)$ converges to a positive definite matrix Q, then under regular conditions $\sqrt{T}\left(\hat{\beta} - \beta\right)$ converges in distribution to $N\left(0, \sigma^2 Q^{-1}\right)$. When ratings are not normally distributed, which is a common situation, the asymptotic approximation is the only way to justify inferences for medium-size to large T. The smaller the matrix Q_T^{-1}, (respectively Q^{-1}) the more (asymptotically) efficient is the OLS estimator. Classical experiments (Cochran and Cox 1957; Cox 1958) usually assume normality and discrete attributes, and for a small T statisticians try to make $X'X$ diagonal (i.e. X is an orthogonal matrix), albeit orthogonal designs are neither always possible (e.g., in models with squared regressors, which is common in polynomial specifications) nor optimal. Allowing for a small correlation between estimators we might obtain estimators with smaller variances.

An experiment X^* is (approximately) optimal if Q_T^{-1} (respectively Q^{-1}) is the smallest possible covariance matrix according to some appropriate criteria measuring the size of this matrix. Suboptimal designs require a larger T to estimate the parameters with the same precision as X^*, increasing the market research cost and rating contamination caused by respondent's fatigue. Notice that for models linear in parameters, optimal designs are not adaptive. In other words, even if data is collected and processed sequentially, we do not use what we learn to change the experimental setting. The reason is that neither the matrix Q_T^{-1} nor Q^{-1} are affected by collected information about the previous ratings, y_{t-1}.

Several algorithms exist for selecting an optimal experimental design, either exact or approximate, which choose some convenient points from a set of candidates. Typically they are gradient-like algorithms, and there is an increasing interest in developing faster algorithms to compute optimal experimental designs. The literature has a more serious drawback. As we discuss later, these methods tend to choose designs with repeated product profiles. In most experimental settings this is not a problem, but it is unacceptable to replicate stimuli in CA. In this paper we propose a general approach to compute exact optimal designs for CA experiments, and we eliminate the problem of profile repetitions. The structure of the article is as follows. Section 2 discusses the state of the art tools for the design of optimal experiments, and their limitations, particularly for CA. In Section 3 we present a new approach to the design of experiments, and motivate the use of appropriate constraints, which prohibit profile repetitions for the same respondent, thus ensuring its suitability for CA. In Section 4 we discuss an integer version of the problem, and mixed cases. Section 5 presents an extension for panels of consumers. We conclude with a discussion of managerial implications and limitations. We also present a few short extensions, such as using partial profiles for complex products with many attributes, and extensions to CA for rank data under invariance to monotonous transformations.

2 Literature review on optimal experimental design

The design of conjoint experiments is a fundamental problem in marketing research. The available methods are designed to provide (nearly) optimal efficiency (see e.g. Kuhfeld, Tobias, and Garratt 1994). In this section we review the tools available for the design of optimal experiments, and the drawbacks for their application to CA experiments.

The experiments considered in CA are based on the classical statistical literature about optimal experimental designs. Broadly speaking there are two big approaches: approximate optimal designs proposed by Kiefer (1959), and exact optimal designs. The Kiefer's approach, seeks designs where the asymptotic covariance $\sigma^2 Q^{-1}$ of the estimators is as small as possible, minimizing some function $\phi\left(\sigma^2 Q^{-1}\right)$ measuring the size of the matrix. By contrast, the second approach is focused on the actual covariance matrix with finite sample T, minimizing a measure $\phi\left(\sigma^2 Q_T^{-1}\right)$. In general, approximate optimal designs are not appropriate for CA, as often the method leads to repetition of the product profiles. Therefore, we will not discuss this approach in detail. Nevertheless, it is useful to understand approximated designs in order to obtain full perspective of the problem. In Appendix A we provide an overview, and some results will be mentioned later in the paper.

The design of conjoint experiments has traditionally focused on exact optimal designs. These designs minimize some function of $\sigma^2 Q_T^{-1}$ measuring the size of this matrix, solving the problem

$$\min_{X\in\chi} \phi\left(\left(X'X\right)^{-1}\right),$$

with $\phi(\cdot) = \text{tr }(\cdot)$ in case of A-optimality, and $\phi(\cdot) = |\cdot|$ for D-optimality. In the first case the sum of variances of the estimators is minimized; in the second, researchers also pursue uncorrelated estimators in the vector $\widehat{\beta}$.

Exact optimal designs have several advantages in CA. First, they minimize the actual covariance of the estimators instead of an approximation. Besides, for optimal exact designs we can consider not only such constraints as $x \in \chi^T$, but we can also include transversal constraints, linking characteristics of product profiles (levels of categorical variables, or simply values of continuous variables). For example we can consider the prices for every attribute, or level, and include them in a budget constraint over the whole experiment $\sum_{t=1}^T c'x_t \leq m$, where c is a $k \times 1$ vector of attribute prices, and m is the total budget. Without loss of generality we can impose that the stimulus belongs to the space $\chi' = \left\{x \in \chi^T : g\left(x\right) \leq 0\right\}$. Once an exact optimal design Q_T has been computed, any design used in practice should be compared to this benchmark.

Several procedures have been considered in the literature. Dykstra (1971) suggested the iterative inclusion of additional profiles, using the recursive expressions for partitioned matrix $|Q_{T+1}| = |Q_T|\left(1 + f\left(x_{T+1}\right)' Q_T^{-1} f\left(x_{T+1}\right)\right)$. The algorithm sequentially selects one observation to improve the determinant, therefore at each iteration the profile x_{T+1} is chosen to maximize $f\left(x_{T+1}\right)' Q_T^{-1} f\left(x_{T+1}\right)$. If χ is finite (with factorial designs), this is done by swapping alternative profiles and evaluating the change in the determinant. Johnson and Nachtsheim (1983) consider some other alternatives. The exchange algorithm can be also applied for trace minimization, using the Woodbury matrix inversion identity $tr\ \left(Q_{T+1}^{-1}\right) = tr\ \left(Q_T^{-1}\right) - tr\ \left(\frac{Q_T^{-1} x_{T+1} x'_{T+1} Q_T^{-1}}{1+x'_{T+1} Q_T x_{T+1}}\right)$. Besides, some procedures initially developed for Kiefer's approximated optimal designs can be applied also in this context, such as the Fedorov method (Fedorov 1972).

Table 1: Exchange algorithms for computing exact designs

Algorithm	Description
Simple exchange algorithm (Mitchell and Miller Jr 1970; Wynn 1972)	Starts with an random n-point design. At each iteration one observation is added which maximizes the determinant, and then another observation deleted to maximize the efficiency gain.
DETMAX (Mitchell 1974)	Starts with an random n-point design. At each iteration the algorithm makes "excursions" from a n-point design: it is permitted to add/delete more than 1 observation until the determinant is improved.
Fedorov (1972)	Starts with an n-point nonsingular design. At each iteration the algorithm simultaneously adds one observation and deletes another so that the increase in determinant is maximal.
Modified Fedorov (Cook and Nachtsheim 1980)	Starts with an n-point nonsingular design. At each iteration the algorithm evaluates all pairs of design and candidate points, and selects the best candidate to switch with each design observation. Makes every swap that increases efficiency.
Coordinate exchange (Meyer and Nachtsheim 1995)	Does not use the candidate set. At each iteration, the initial design is improved by exchanging each point coordinate (attribute level) with every other possible coordinate. Exchanges which increase efficiency are maintained.

Table 1 presents a comparative summary of the commonly applied exchange algorithms. A detailed comparison and evaluation of their computational performance can be found in Cook and Nachtsheim (1980). In general, none of these methods exploits satisfactorily the available numerical optimization tools. But there is a more relevant drawback. After the optimal design is computed, we typically observe that a few rows (product profiles) are repeated several times, which is a major problem for its application in CA. This is not a surprising result, since the arguments of Lemma 1 in Appendix A also apply to the set $\mathbf{Q}_T = \{Q = X'X : X = [f(x_1)', \ldots, f(x_T)']'\}$. Therefore, with exact optimal designs we end up with repeated vertex questions with certain frequencies, not very differently from Kiefer's approximate designs.

3 A direct method for optimal exact designs

In this section we propose an efficient approach for computing exact optimal designs without repeated stimuli, providing the basis for usability of this approach in CA. We begin with an analysis of properties of optimal design problems, and we discuss the approach to solve this optimization problem efficiently with Newton-based methods. Next, we demonstrate how to create designs without duplicated treatments, which often appear in optimal designs. We also present some initial numerical results.

3.1 Using Newton-based algorithms

We begin with a simple optimization problem

$$\begin{aligned} \min_{x} \quad & \phi\left(\left(X'X\right)^{-1}\right) \\ \text{s.t.} \quad & \\ & X = [f(x_1)' \ldots f(x_T)']' \\ & x \in \chi^T, \end{aligned} \tag{1}$$

where ϕ is a measure of the size of a matrix, trace or determinant. It is a convex problem, since the objective function ϕ is convex, and we assume that the feasible set of experimental attributes is a nonempty, compact and convex set. The solution, x^*, is the exact optimal design matrix. Note that the optimal design x^* is not unique, as any permutation of the rows in x^* (reordering the questions or product profiles) renders the same matrix $Q_T = X'X$. Any of these solutions is equivalent. Different types of constraints can be considered to flexibly handle a variety of marketing scenarios and managerial problems: linear and nonlinear equality, or inequality constraints. Lower and upper bounds on x represent the set of feasible attributes, χ^T.

There are several Newton-based algorithms for general constrained convex mathematical programming with good theoretical properties. To solve Problem (1) with a Newton's method, we first calculated the first- and second-order derivatives. Unless stated otherwise, numerical examples considered in the paper assume that $f(x_t) = x_t$, and the design matrix $X = [x_1', \ldots, x_T']'$. Objective functions, gradients and Hessians for minimization of A- and D-optimality criteria for this benchmark case are presented in Table 2. In case of discrete attributes we also include intercept and consider transformation of variables to eliminate dummy collinearities. The proof for a more general expression can be found in the Appendix B, and can be easily adapted for other specifications of f.

Table 2: First and second order derivatives of the benchmark problems

	D-optimality	A-optimality
Objective	$\min \lvert \left(X'X\right)^{-1} \rvert$	$\min \operatorname{tr}\ \left(X'X\right)^{-1}$
Gradient	$-2 \left\lvert \left(X'X\right)^{-1} \right\rvert\ \text{vec}\ X \left(X'X\right)^{-1}$	$-2\ \text{vec}\ X \left(X'X\right)^{-2}$
Hessian[a]	$4 \left\lvert \left(X'X\right)^{-1} \right\rvert \left(\left(X'X\right)^{-1} \otimes X \left(X'X\right)^{-1} X'\right) +$ $2 \left\lvert \left(X'X\right)^{-1} \right\rvert K \left(X(X'X)^{-1} \otimes (X'X)^{-1}X'\right) +$ $2 \left\lvert \left(X'X\right)^{-1} \right\rvert K \left((X'X)^{-1}X' \otimes X(X'X)^{-1}\right) -$ $2 \left\lvert \left(X'X\right)^{-1} \right\rvert \left(\left(X'X\right)^{-1} \otimes I\right)$	$4 \left(\left(X'X\right)^{-1} \otimes X \left(X'X\right)^{-2} X'\right) +$ $4 \left(\left(X'X\right)^{-2} \otimes X \left(X'X\right)^{-1} X'\right) -$ $2 \left(\left(X'X\right)^{-2} \otimes I\right)$

[a] K is the commutation matrix, which transforms vec X into vec X'.

We have solved several numerical examples and observed that exact optimal designs indeed have repeated profiles, as expected from applying Lemma 1 to the set $\mathbf{Q}_T = \{Q = X'X : X = [x_1' \ldots x_T']'\}$. Below we discuss how to overcome this problem.

3.2 Avoiding repeated questions

The issue of duplicated product profiles can be resolved by imposing simple quadratic constraints on Problem (1), which prohibits profile repetitions in the optimal design matrix. Define a $T \times T$ similarity matrix $S = XX'$, with element i, j given by $S_{i,j} = x_i x_j'$, where x_i, x_j are corresponding rows of X. Notice that the Euclidean distance $d_{i,j} = \sqrt{(x_i - x_j)(x_i - x_j)'}$ satisfies $d_{i,j}^2 = S_{i,i} + S_{j,j} - 2S_{i,j}$, meaning that the matrix $D = [d_{ij}^2]$ can be expressed as

$$D = \text{diag}\ (S)\ 1_t' + 1_t\ \text{diag}\ (S)' - 2S,$$

where diag (S) is a vector containing the elements in the main diagonal of $S = XX'$, and 1_t is a $t \times 1$ vector of ones. Both S and D are symmetric matrices, and the diagonal elements in D are zero. We consider a lower bound over the Euclidean distance between stimulus i and stimulus j, for all pairs of different questions shown to the same respondent

$$L(D) \geq \underline{d},$$

where the linear operator $L(\cdot) : \mathbb{R}^{T\times T} \to \mathbb{R}^{T(T-1)/2}$ selects the lower triangle elements of a square matrix (excluding the diagonal elements equal to 0, and the symmetric upper triangle terms), and stacks them in a column vector; $\underline{d}$ is a $T(T-1)/2$ vector of positive distance tolerances, and the inequality is applied pointwise. Notice that $L(D) = H \cdot vec(D)$, where $vec(\cdot) : \mathbb{R}^{T\times T} \to \mathbb{R}^{T^2}$ is the operator that stacks the columns of a matrix, and H is a $\mathbb{R}^{T(T-1)/2\times T^2}$ sparse matrix

$$H = \left(\begin{array}{cccccccccc} \mathbf{0}_{(T-1)\times 1} & \mathbf{I}_{T-1} & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\ \cdots & \cdots & \mathbf{0}_{(T-2)\times 2} & \mathbf{I}_{T-2} & \cdots & \cdots & \cdots & \cdots & \cdots & \\ \cdots & \cdots & \cdots & \cdots & \mathbf{0}_{(T-3)\times 3} & \mathbf{I}_{T-3} & \cdots & \cdots & \cdots & \mathbf{0}_{T\times T} \\ & & & & & & \ddots & & & \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & & \cdots & \cdots & \\ 0_{1\times 1} & \mathbf{0}_{1\times(T-1)} & \mathbf{0}_{1\times 2} & \mathbf{0}_{1\times(T-2)} & \mathbf{0}_{1\times 3} & \mathbf{0}_{1\times(T-3)} & \cdots & \mathbf{0}_{1\times(T-1)} & \mathbf{I}_1 & \\ & & & & & & & & & \end{array} \right)_{T(T-1)/2\times T^2}$$

where $\mathbf{I}_r$ is the $r \times r$ identity matrix, and blank spaces are adequately sized blocks of zeros (as shown in the last row). For example for $T = 3$,

$$L \begin{pmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{pmatrix} = \begin{pmatrix} \mathbf{0} & \mathbf{1} & \mathbf{0} & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{0} & \mathbf{0} & \mathbf{1} & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} d_{11} \\ d_{21} \\ d_{31} \\ d_{12} \\ d_{22} \\ d_{32} \\ d_{13} \\ d_{23} \\ d_{33} \end{pmatrix} = \begin{pmatrix} d_{21} \\ d_{31} \\ d_{32} \end{pmatrix}$$

The matrix H can be obtained from the identity matrix I_{T^2}, by eliminating rows that correspond to diagonal and upper-triangle elements (for example, with $T = 3$ these are the rows 1,4,5,6,7,8).

3.3 Numerical results for some benchmark problems

We performed a series of simulations to test the performance of the algorithm and to compare the behavior of both criteria, trace and determinant. We begin with continuous attributes, and the problem with discrete variables will be discussed in the following Section 4. Both trace and determinant problems consider the $T(T-1)/2$ inequality constraints discussed in the previous section, and lower and upper bounds on the continuous attributes. The algorithm was implemented using MATLAB 6.5 on Mobile Workstation, Intel Core$^{\text{TM}}$2 Duo 2.20 GHz, with machine precision 10e-16. Both problems have been solved using the MATLAB subroutine "fmincon" with the option "interior-point" algorithm, included in the Optimization toolbox. Since the 1980s *interior point methods* have become popular for solving nonlinear constrained problems (also large-scale). They are very efficient, both in terms of theoretical worst-case complexity and practical performance. The interior-point approach to constrained minimization is to solve a sequence of perturbed minimization problems by some parameter. As this parameter decreases to zero, the minimum of perturbed minimization problem should approach the minimum of original minimization problem (for details see e.g. Byrd, Hribar, and Nocedal 1999). To solve the perturbed problem, we consider a Newton framework using a line search. Solving the Karush-Kuhn-Tucker equations, we first compute the Newton search direction, $p_k = -H_k^{-1} g$, where H is the exact Hessian $\nabla^2\phi(x)$, and g is the gradient $\nabla\phi(x)$. To guarantee global convergence, we then compute a step size that determines the adjustment of the Newton direction, ensuring sufficient decrease and uniform progress towards a solution (Nocedal and Wright 2006).

Table 3 summarizes the parameter values for different simulation scenarios. We have solved Problem (1) for conjoint experiments of different sizes: small, medium, and large, with varying parameters for the number of stimuli (T) and product attributes (k). We have chosen sufficiently large T to ensure enough degrees of freedom for estimation of integer cases and interactions, which is the subject of the next Section. We have also checked that already for small values of $\underline{d}$, we overcome the problem of stimuli repetitions. The algorithm converges in seconds in all the cases. Below we analyze in detail the comparative behavior of trace and determinant, as well as the sensitivity of the algorithm to initial points.

Table 3: Parameter values for simulation of benchmark problems

Problem size	Small	Medium	Large
# profiles (T)	10	16	25
# attributes (k)	3	5	8
# model parameters (p)	3	5	8
Lower bound (lb)	1	1	1
Upper bound (ub)	10	10	10
Distance ($\underline{d}$)	5	5	5

Table 4 summarizes the algorithm results based on 100 runs with random initial points. The performance was evaluated in terms of the quality of the attained solution and computational cost. For the former we report the lowest and the highest objective function value attained in all runs (lowest and highest value of all local minima), and the quartiles. As with all other local algorithms, this approach may be trapped in a local minimum. To overcome this problem we should re-run the procedure few times. We also report medians for: (1) the rank of the optimal design matrix, and (2) the conditioning of the information matrix. To allow for comparability of A- and D-optimality measures we calculate $\phi_1(X_d) = \text{tr}\ \left(X_d^{*'}X_d^*\right)^{-1}$, where X_d^* is the solution to the determinant problem, and $\phi_2(X_a) = \left|\left(X_a^{*'}X_a^*\right)^{-1}\right|$, with X_a - solution to the trace problem. The evaluation of algorithm's computational cost is based on the median number of iterations, median number of function evaluations and median time needed for convergence.

For both trace and determinant criteria the convergence of algorithms takes a few seconds, and for the majority of scenarios the solution was found in less than a second. The determinant criterion converges faster than trace in all cases, however its performance is suspicious. We can observe that for medium-to-large scenarios the solution obtained with trace algorithm yields better determinant values than the solution to determinant problem (compare the left and right panel for "Determinant" block). This suggests that the determinant algorithm gets easily stuck in a local minimum.

As the dimension of X grows, the function $|\,(X'X)^{-1}\,|$ approaches rapidly to 0, so that the objective function values become smaller than the "machine epsilon" (computer upper bound on the relative error due to rounding in floating point arithmetic operations). It means that for any sufficiently large matrix X, the determinant of the inverse of the information matrix will be essentially zero (rounding off at the 16th decimal place). The algorithm does not iterate because any initial point leads to function value equal to 0, and is therefore identified as the solution. These results imply that direct optimization of D-optimality criterion often does not work well in practice. To moderate these problems, one could consider a logarithmic transformation of the objective function, or scaling it by multiplying by a sufficiently large scalar, but the trace criteria typically works better.

We have also analyzed the optimal designs qualitatively. In case of trace algorithm, indeed the values of elements of X were close to lower and upper bounds; in the determinant case which was stuck in a local minimum, the solutions were included in the sampling interval further from the boundaries. Therefore, for the trace, the optimal solution lies relatively close to the bounds of the problem, confirming the intuition that evaluating extreme stimuli yields most information. The optimal design matrix is of full rank in both cases, however the solution to trace problem has better conditioning than the solution to determinant problem.

Table 4: Simulation results for trace and determinant problems

Objective function	min tr $(X'X)^{-1}$			min $\lvert (X'X)^{-1} \rvert$		
Problem size	small	medium	large	small	medium	large
# of distinct solutions[a]	10	16	9	1	1	1
Algorithm converged[b]	94%	100%	100%	100%	100%	100%
Trace:[c]						
Minimum value	0.0090	0.0109	0.0128	0.0148	0.0337	0.0470
25% percentile	0.0094	0.0112	0.0130	0.0168	0.0488	0.0563
50% percentile	0.0095	0.0114	0.0131	0.0176	0.0552	0.0629
75% percentile	0.0096	0.0117	0.0133	0.0181	0.0644	0.0711
Maximum value	0.0101	0.0128	0.0136	0.0273	0.1034	0.0924
Determinant:[c]						
Minimum value	1.56e-08	1.79e-14	9.41e-24	4.64e-08	1.24e-12	4.05e-20
25% percentile	1.70e-08	1.99e-14	1.04e-23	6.00e-08	4.54e-12	1.11e-19
50% percentile	1.74e-08	2.24e-14	1.11e-23	6.64e-08	7.85e-12	1.99e-19
75% percentile	1.75e-08	2.39e-14	1.18e-23	7.17e-08	1.14e-11	2.94e-19
Maximum value	2.16e-08	3.00e-14	1.39e-23	1.36e-07	4.16e-11	8.38e-19
# iterations[d]	23	22	25	10	0	0
# function evaluations[d]	24	23	27	11	1	1
Time[d] (s)	0.12	0.37	2.69	0.06	0.02	0.08
Rank[d]	3	5	8	3	5	8
Condition[d]	4.61	8.13	14.24	9.01	65.83	122.38

[a] Objective function values are rounded to the 4th decimal place.
[b] Unsuccessful runs are due to a saddle point.
[c] Bordered blocks correspond to objective function values.
[d] Reported values are medians of all successful runs.

Summarizing, the optimization of trace criterion is a more reliable approach, because its performance is not significantly affected by the increase in the problem dimension: the number of iterations and function evaluations remains relatively stable across scenarios, and the solutions obtained with different initial points are close. The convergence is fast and even for the largest problems it does not take longer than 3 seconds. Given the high chances of converging to a suboptimal local minimum, the stability of trace criterion becomes a useful advantage, outperforming the determinant in practice. Therefore, we will focus on the trace in the remainder of the article.

4 The case of discrete and mixed attributes

In CA we often find discrete attributes. For example, whether a certain material is used, or a component has been selected from a given catalogue. Continuous variables, like prices, are also often represented by a small number of meaningful levels and treated as discrete variables. Typically, CA models have several categorical and perhaps also some continuous attributes. In experimental context, the discrete attribute is known as a factor, and the alternative values that it can take are known as levels of the factor. The standard formulation in a model with $i = 1, ..., L$ integer attributes, each of them having J_i levels is

$$y_t = \alpha + \sum_{i=1}^{L} \sum_{j=1}^{J_i} \gamma_{ij} \; d_{ijt} + \beta' Z_t + \epsilon_t, \tag{2}$$

where Z_t represents the set of p continuous regressors. The design matrix $x = [D_1, \ldots, D_L, Z]$ is partitioned in a way that every discrete attribute is represented by a matrix of indicator variables, $D_i = [d_{ijt}]$, taking values 0 or 1 to indicate the absence or presence of some level in the profile. Linear regression models with discrete dummies, like the one defined in equation (2), are affected by collinearities as $\sum_{j=1}^{J_i} d_{ijt} = 1, \ \forall \ i, t$. We consider the standard methods to eliminate multicollinearity from the model: (1) omission of one level in every factor, and (2) including dummy differences with respect to one factor. Depending on the selected method, the OLS estimators will be different as well as their covariance matrix, and we will obtain different optimal designs x^*.

(D1) The first approach involves substituting the regressor identity in the model. For example with $d_{iJ_it} = 1 - \sum_{j=1}^{J_i-1} d_{ijt}$ we can express

$$\begin{aligned} y_t &= \alpha + \sum_{i=1}^{L}\left(\sum_{j=1}^{J_i-1}\gamma_{ij}\ d_{ijt} + \sum_{i=1}^{L}\gamma_{iJ_i}\left(1-\sum_{j=1}^{J_i-1} d_{ijt}\right)\right) + \beta' Z_t + \epsilon_t \\ &= \left(\alpha + \sum_{i=1}^{L}\gamma_{iJ_i}\right) + \sum_{i=1}^{L}\sum_{j=1}^{J_i-1}\left(\gamma_{ij} - \gamma_{iJ_i}\right) d_{ijt} + \beta' Z_t + \epsilon_t, \end{aligned}$$

which is equivalent to **level omission**, and the interpretation of coefficients is relative to the parameter of a missing level. In a model with more factors, transformation by level omission can be conveniently written in a matrix form, $\tilde{f}(x) = xA$. The matrix A can be obtained from the identity matrix of size $\left(\sum_{i=1}^{L} J_i + k\right)$ by eliminating columns associated to the omitted levels. This method is sometimes called as *binary coding*.

(D2) In the second approach additional constraints are included, usually that $\sum_{j=1}^{J_i}\gamma_{ij} = 0$ (the dummy coefficients sum up zero). Substituting $\gamma_{iJ_i} = -\sum_{j=1}^{J_i-1}\gamma_{ij}$ in the model leads to

$$y_t = \alpha + \sum_{i=1}^{L}\sum_{j=1}^{J_i-1}\gamma_j\ \left(d_{jt} - d_{iJ_it}\right) + \beta' Z_t + \epsilon,$$

where new regressors are defined as **dummy differences**, $d'_{ijt} = (d_{ijt} - d_{iJ_it}) \in \{-1, 0, 1\}$.

Let $\tilde{f}(x) = x(I - B)A$ represent the transformation of dummy variables, which creates dummy differences with respect to the last level in each factor. In particular, I is an identity matrix of size $\left(\sum_{i=1}^{L} J_i + k\right)$, A is defined above, and B is a square sparse matrix with value 1 at columns associated to the omitted levels and zero otherwise. This method is sometimes called as *effects coding*.

In our examples, the design matrix will be partitioned as $X = f(x) = [1, \tilde{f}(x)]$, where the first column corresponds to the intercept, and $\tilde{f}$ represents the dummy coding method (D1 and D2).

Note that when the number of factors, L, is very small (one or two), and there are no continuous attributes, the number of different stimulus profiles that can be included in the experiment is small, and the experimental design problem is not relevant. All possible combinations of factor levels can be included in the experiment. Moreover, since replications are not allowed in CA, the inference analysis should be based on small sample analysis (typically under normality assumptions). But when the number of factors is large we may have larger size T, because the number of alternative stimuli increases multiplicatively, as $\prod_{i=1}^{L} J_i$, whereas the number of parameters increases additively. In this case, the experimental design does become important, as well as for the mixed CA (with both discrete and continuous attributes).

The selected procedure (D1) or (D2) affects the interpretation of the model parameters, but it does not essentially affect the efficiency (given the OLS estimators drawn with one of these procedures, we can directly recover the exact OLS estimations from the other procedure and vice versa). Again, we obtain the optimal design by minimizing the trace or the determinant of $(X'X)^{-1}$. The determinant is a more popular criterion, but it has some limitations, which we discussed in the previous section. When there are no continuous attributes, the approach (D1) renders orthogonal designs, as the observation vectors for different dummies are naturally orthogonal. In the second approach, often the columns in X sum up to zero, which is known as a balanced design (it happens when all attributes are run the same number of times at each level). Nevertheless the trace/determinant of the optimal matrix $(X'X)^{-1}$ might be quite different in approach (D1) and (D2). In any case, the relative efficiency of optimal solutions from each coding approach should not be directly compared as each procedure estimates different parameters.

To handle discrete attributes, we consider a branch-and-bound algorithm searching a tree, whose nodes correspond to continuous nonlinearly constrained optimization problems. The solvers have been compiled in both a sparse and a dense version, and are commercially available with TOMLAB (http://tomopt.com/tomlab/) - a software package in MATLAB for practical solution of optimization problems. TOMLAB includes several solvers for the solution of all types of applied optimization problems. In particular we consider MINLP solver developed by Roger Fletcher and Sven Leyffer at the University of Dundee. MINLP implements a branch-and-bound algorithm and a sequential quadratic programming (SQP) trust region algorithm, using a recently developed filter technique to promote global convergence (Leyffer 2001).

Formally, the optimal design problem in the mixed-integer conjoint context is

$$\begin{aligned} \min_{x} \quad & \text{tr } \left(X'X\right)^{-1} \\ \text{s.t} \quad & X = f(x) = [1, \tilde{f}(x)] \\ & x = [D_1, \ldots, D_L, Z] \\ & \sum_{j=1}^{J_i} d_{ijt} = 1, \ \forall\ i, t \\ & L\left(diag\left(xx'\right)1' + 1diag\left(xx'\right)' - 2xx'\right) \geq \ \underline{d} \\ & lb \leq Z \leq ub \\ & d_{ijt} \in \{0,1\} \text{ are integer,} \end{aligned} \tag{3}$$

where we choose optimally the value of continuous attributes, as well as the factor level to be shown in each stimuli. We include the intercept, as well as transformation to eliminate perfect collinearity in the dummy variables, $\tilde{f}(x)$. The third constraint requires that within each factor exactly one level is shown in a product profile, and is a simple linear equality constraint. We also impose the similarity constraint for avoiding repetitions (as motivated in Section 3.2). Additionally lower and upper bounds on variables can be considered, which for dummy variables are naturally 0 and 1.

With transformation of dummy variables specified as a linear function of the design matrix x, we can directly apply the results of the Appendix B, to obtain subroutine inputs: the gradient and Hessian. Note that the respective transformation matrices A, B are constant, therefore the expressions from the Appendix are further simplified.

We performed some simulations for one integer and two mixed examples. Table 5 summarizes parameter values for different scenarios.

Table 5: Parameter values for simulation of integer problems

	Scenario 1	Scenario 2	Scenario 3
Type	integer	mixed	mixed
# profiles (T)	10	16	25
# continuous attributes (k)	0	2	5
# integer attributes (L)	3	3	3
# attribute levels (J_i)	[3,3,3]	[3,3,3]	[3,3,3]
# model parameters (p)	7	9	12
Lower bound[a] (lb)	-	1	1
Upper bound[a] (ub)	-	10	10
Distance ($\underline{d}$)	1	1	1

[a] Lower and upper bounds considered on the set of continuous attributes, Z.

Table 6 presents simulation results based on 100 algorithm runs with different initial points, following the simulation approach in Section 3.2. Recall that the covariances of the two methods for eliminating collinearity in the dummies cannot be directly compared in terms of relative efficiency, as they estimate different parameters. As far as the quality of the solution is concerned, in all scenarios the transformed design matrix $X = f(x)$ is of full rank, and for Scenario 1 the collinearity problem is eliminated. The optimal design matrix obtained with "dummy differences" algorithm has in general better conditioning, than one obtained when omitting one level. In terms of computational cost, the performance of both approaches is similar.

Table 6: Simulation results for mixed and integer scenarios

Objective function			min tr $(X'X)^{-1}$			
	Scenario 1		Scenario 2		Scenario 3	
Approach to collinearity[a]	D1	D2	D1	D2	D1	D2
# of distinct solutions[b]	4	4	87	77	77	59
Algorithm converged[c]	100%	100%	90%	89%	77%	62%
Trace[d]						
Minimum value	4.0625	1.3333	2.6203	0.8998	1.6600	0.5938
25% percentile	4.0625	1.3333	2.6885	0.9110	1.7105	0.6038
50% percentile	4.0625	1.3333	2.7161	0.9178	1.7282	0.6112
75% percentile	4.0625	1.3542	2.7716	0.9257	1.7699	2.4216
Maximum value	4.5625	1.3958	26.3661	4.8150	11.1515	6.0186
Determinant:						
Minimum value	0.0023	3.18e-06	2.51e-09	2.19e-12	1.64e-18	2.87e-21
25% percentile	0.0023	3.18e-06	3.29e-09	4.31e-12	3.08e-18	5.38e-21
50% percentile	0.0023	3.18e-06	3.72e-09	5.02e-12	3.92e-18	6.45e-21
75% percentile	0.0023	3.18e-06	4.51e-09	6.14e-12	6.99e-18	4.36e-19
Maximum value	0.0023	3.18e-06	1.02e-06	1.55e-09	3.32e-15	3.34e-18
# iterations[e]	1	1	1	1	1	1
# function evaluations[e]	53.5	38	82	87	90	91
Time[e] (s)	7.35	5.81	85.91	65.46	733.74	811.38
Rank[e]	7	7	9	9	12	12
Condition[e]	16	4	603.90	154.34	1540.17	376.32

[a] D1 - level omission; D2 - including dummy differences.
[b] Objective function values are rounded to the 4th decimal place.
[c] Unsuccessful runs are due to stack overflow and exceeding the working memory limit.
[d] Bordered blocks correspond to objective function values.
[e] Reported values are medians of all successful runs.

Some of our conclusions drawn from continuous conjoint problem are confirmed here. As predicted, including additional profiles, attributes and factor levels increases the optimization costs: more time and function evaluations are needed to converge to the optimum. The convergence for the pure integer scenario is a matter of seconds, and the objective function values obtained from different initial points are very close. The mixed-integer problem is more complex and computationally challenging. The median times for convergence range from 1 to 13 minutes, while the number of function evaluations remains quite stable in both scenarios.

4.1 Model with interactions: fractional factorial designs

Consider a CA models with several factors (discrete attributes), where each factor may take different levels. A full factorial model considers all possible interactions for each dummy in the model (factors and levels).

$$y_t = \gamma + \sum_{j_1=1}^{J_1} .... \sum_{j_L=1}^{J_L} \beta_{j_1,j_2,...,j_K} \times (d_{j_1 t} d_{j_2 t} \cdots d_{j_K t}) + \varepsilon_t.$$

The number of parameters increases multiplicatively with $J_1 \times ...J_L$. The model can also include continuous attributes. Then we can have also interaction between the dummies and the continuous attribute.

In any case, the effort required to estimate a full factorial model is cost-prohibitive and tedious for the respondent. In practice researchers generally use fractional-factorial designs, containing just interactions of a few factors (e.g. products of pairs, or threesomes of dummies), and evaluating fewer product profiles. For an introduction see Addelman (1962), Green (1974), or Kuhfeld et al. (1994).

To handle fractional factorial designs, necessary transformations already have been made to eliminate the multicollinearity from the design matrix x. Multicollinearity is handled omitting one level per factor, or replacing the dummy variables by deviations with respect to one of the levels. To estimate this model and perform inference analysis, we need to assume normality, and compute one rating observation for each combination (for asymptotic analysis we would need replications, computing several ratings per combination). First note that the interaction term between variable a and b can be written in terms of the design matrix x

$$W = \sum_{t=1}^{T} e_t \left(e_t' f(x) E_a \otimes e_t' f(x) E_b \right)$$

where $f(x) = [1, \tilde{f}(x)]$ is a transformed design matrix where we already eliminated the multicollinearity, and included the intercept. e_t is a unit vector with 1 in position t, so that $e_t' f(x)$ selects the t-th row of $f(x)$, and E_a, E_b are elementary matrices, so that $f(x)E_a$ and $f(x)E_b$ select columns of $f(x)$ corresponding to variables a, and b respectively. Calculation of the gradient can be found in Section 8.3, in the Appendix B. The analytical formula for the Hessian is much more involved, therefore the algorithm uses finite differences to calculate it.

We provide some examples to illustrate the behavior of the method, including Scenario 2 from the previous section, and omitting a factor level to eliminate multicollinearity (approach D1 from previous section). The examples consider a model with two-way interactions between: 2 continuous attributes (“Continuous” case), a continuous and a categorical variable (“Mixed” case), and 2 categorical variables (“Integer” case). Table 7 presents the simulation results based on 100 runs with random initial points.

The performance of the approach for the “Continuous” and “Mixed” case is very good. Including interactions does not result in the increase in computational cost, in comparison to the model without interactions. As far as the quality of the solution is concerned the optimal design matrix is of full rank, but the algorithm sometimes converges to a local minimum (there are a few outlying objective function values). However, ”the curse of dimensionality” affects the performance of the proposed approach as the standard Branch-and-Bound algorithm is considered. Other alternative algorithms can be considered to tackle this issue (Lawler and Wood 1966).

Table 7: Simulation results for a model with interactions

Objective function	min tr $(X'X)^{-1}$		
Type of interaction:	Continuous	Mixed	Integer
# of distinct solutions[a]	92	91	26
# algorithm success[b]	97%	96%	26%
Trace[c]:			
Minimum value	2.6754	2.6921	14.2166
25% percentile	2.7389	2.7542	14.2720
50% percentile	2.7818	2.8245	14.5569
75% percentile	2.8317	3.6244	78.8042
Maximum value	28.3776	1.7E+12	1.15e+14
Determinant:			
Minimum value	1.35e-09	1.35e-09	4.94e-09
25% percentile	2.34e-09	2.37e-09	6.52e-09
50% percentile	2.82e-09	3.03e-09	1.09e-08
75% percentile	3.55e-09	3.97e-09	1.27e-07
Maximum value	5.86e-07	1.93e-06	1.02e-06
# iterations[d]	1	1	18.5
# function evaluations[d]	37	39	328.5
Time[d] (s)	72.73	79.86	767.82
Rank[d]	9	9	9
Condition[d]	515.10	595.27	1269.92

[a] Objective function values are rounded to the 4th decimal place.

[b] Unsuccessful runs are due to stack overflow and exceeding the working memory limit.

[c] Bordered blocks correspond to objective function values.

[d] Reported values are medians of all successful runs.

4.2 A comparison with commonly used software

We have compared the performance of our approach with the software which is commonly used by practitioners in traditional conjoint experiments: Conjoint Value Analysis (CVA) by Sawtooth Software and %MktEx by SAS. Both programs allow only categorical attributes and rely on exchange algorithms (see Table 1) to optimize the determinant of the covariance matrix. Continuous attributes like prices are not explicitly permitted. Instead, they are usually discretized and represented by a few meaningful levels.

The setting is as follows. To ensure comparability of results with Sawtooth Software and SAS, we focus exclusively on experiments with discrete attributes and begin with determinant minimization. The design matrix X has an intercept, and categorical variables are orthogonally coded, which is a common practice in CA (for details see Kuhfeld 2010). The values of orthogonal codes of dummy variables with 2 and 3 levels are presented in Table 8.

We have created 4 hypothetical conjoint experiments, with varying number of product profiles (T), categorical product attributes (L), and attribute levels (J) (see the upper panel in Table 9). As far as the profile repetitions are concerned, the designs obtained with our approach will never have duplicated observations, for SAS we have activated the “no duplicates” options , and Sawtooth Software’s CVA does not take this issue into account. To achieve additional efficiency gains in the performance of our algorithms we used sparse versions of the constraints and their derivatives, taking into account patterns of non-zero elements in the corresponding matrices. For each of the scenarios, we have run 10 times our “determinant” algorithm, and chosen the design with the smallest objective function value.

Table 8: Orthogonal coding of dummy variables

2-level dummy			3-level dummy				
Original		Orthogonal	Original			Orthogonal	
1	0	1.0000	1	0	0	1.3660	-0.3660
0	1	-1.0000	0	1	0	-0.3660	1.3660
			0	0	1	-1.0000	-1.0000

If orthogonality is imposed in the design (meaning $X'X$ is diagonal), then the trace and determinant are closely related. Denote by v_i the sample variances of each regressor. For orthogonal regressors, the A-optimality criterion minimizes $\sum_{i=1}^{p}(1/v_i)$, and the D-optimality criterion minimizes $\prod_{i=1}^{p}(1/v_i)$. By the inequality of arithmetic and geometric means, we obtain that

$$tr\left(Q_T^{-1}\right) = \sum_{i=1}^{p}\frac{1}{v_i} \le p\left(\prod_{i=1}^{p}\frac{1}{v_i}\right)^{1/p} \le p\left|Q_T^{-1}\right|^{1/p}$$

holds with equality if and only if all variances are identical. In particular this happens for pure factorial designs (discrete attributes only) with binary coding of dummies. Notice that for binary regressors the variance $v_i = p_i\left(1-p_i\right)/T$ where p_i is the frequency of level 1. Then $1/v_i$ is minimized when $p_i = 0.5$, i.e. when the same number of 0s and 1s are included for all regressors, and therefore both criteria are equal. This happens at the minimum of both criteria, but not in other case. Nevertheless we have found that trace minimization renders better numerical results. When optimizing the determinant, again we have encountered problems discussed previously in Section 3.3. For larger conjoint experiments the determinant of the covariance matrix is smaller than machine epsilon, and therefore the roundoff objective function value is 0. This prevents the algorithm from iterating towards a better solution. Minimizing the trace we search implicitly for the same optimum, but the problem has a much better numerical behavior.

Lower panel of Table 9 summarizes optimization results and design characteristics obtained with SAS, Sawtooth Software and our both approaches: minimizing the determinant and trace. In all cases we report two efficiency measures: determinant and trace of the optimal design. Recall, that the optimal design in SAS, Sawtooth Software, and "determinant" version of our approach is computed by minimizing the determinant. Finally, we also present the results from our approach based on trace minimization. When available, we provide a few measures of algorithm's computational cost: time to converge, number of iterations, function evaluations, and for SAS number of operations needed to find the design[1].

For small conjoint experiments (COMP1 and COMP2) our "determinant" approach achieves the same design efficiency as SAS and Sawtooth Software at a lower computational cost. For larger conjoint experiments numerical optimization of the determinant is problematic, and the algorithm gets stuck in a local minimum, which is a problem also for Sawtooth Software (COMP4). On the other hand, when minimizing the trace, which is a more stable criterion, in all scenarios we achieve the same design efficiency as SAS: the traces and determinants of covariance matrices calculated with SAS and our approach are equal. Moreover, our "trace" algorithm performs faster than SAS in 3 out of 4 cases. Table 10 compares the results to COMP2 example obtained with SAS, Sawtooth Software, and our "determinant" and "trace" approach.

In this section we presented only a part of functionality of our approach. To ensure comparability with available CA software, we have limited the scope of comparative examples to experiments where treatments are only categorical variables. Our "trace" approach achieves the same efficiency as SAS, and is faster in most of the examples considered. Despite the problems with numerical optimization of determinant function, our "determinant" algorithm still matched %MktEx macro in terms of design efficiency in two of the scenarios, outperforming it in terms of computational cost. Furthermore, our approach is far more flexible and provides functionalities which are not available either in SAS or Sawtooth Software. We can optimize the trace or determinant, and handle continuous and/or discrete variables. Additionally, we can

[1] Number of operations is calculated as the sum of the following positions in SAS output: # algorithm searches, # design searches, # design refinements.

Table 9: Optimal design search in 4 comparative scenarios

		COMP1	COMP2	COMP3	COMP4
Parameters					
	# profiles (T)	8	10	16	18
	# attributes (L)	4	3	4	5
	# levels (J)	[2, 2, 2, 2]	[3, 3, 3]	[3, 3, 3, 3]	[3, 3, 3, 3, 3]
SAS					
	Determinant	3.05e-5	1.18e-7	1.91e-11	1.56e-14
	Trace[a]	0.6250	0.7292	0.5972	0.6111
	Time (s)	2.60	3.84	4.00	3.25
	# Operations	1	82	61	1
Sawtooth Software					
	Determinant	3.05e-5	1.18e-7	1.91e-11	1.75e-14
	Trace[a]	0.6250	0.7292	0.5972	0.6250
	Time (s)	1	1	4	15
Our approach minimizing determinant					
	Determinant	3.05e-5	1.18e-7	3.19e-11	1.19e-13
	Trace[a]	0.6250	0.7292	0.6729	0.9395
	Time (s)	0.12	0.61	0.31	0.50
	# Iterations	1	3	1	1
	# Function evaluations	4	40	4	4
Our approach minimizing trace					
	Trace	0.6250	0.7292	0.5972	0.6111
	Determinant[b]	3.05e-5	1.18e-7	1.91e-11	1.56e-14
	Time (s)	0.27	0.33	2.04	3.95
	# Iterations	1	1	1	1
	# Function evaluations	12	10	42	28

[a] Trace of the covariance matrix of the D-optimal design.

[b] Determinant of the covariance matrix of the A-optimal design.

solve problems by imposing quite general linear and nonlinear constraints, for example experimental budget restrictions. Our approach combines the flexibility and computational efficiency, which gives it an overall advantage compared to previous ones. Next we discuss extensions of the method to many other contexts, and we focus on the trace.

Table 10: Optimal designs from Comparison 2

	SAS			Sawtooth Software			“Det” approach			“Trace” approach		
Attribute	A1	A2	A3	A1	A2	A3	A1	A2	A3	A1	A2	A3
Profile												
1	1	3	1	1	2	1	2	2	1	1	2	1
2	3	3	2	1	3	2	1	2	2	3	1	1
3	2	2	1	3	3	3	1	1	2	2	1	1
4	3	2	3	1	1	3	3	3	1	1	3	3
5	2	1	2	2	2	3	3	2	3	1	1	2
6	1	2	3	2	1	2	1	3	3	2	3	2
7	3	1	1	3	2	2	2	3	2	3	1	3
8	1	1	3	2	3	1	2	1	1	2	2	3
9	2	3	3	3	1	1	3	2	1	3	2	2
10	1	2	2	1	3	3	2	1	3	3	3	1
						Level balance						
Level 1	4	3	3	4	3	3	3	3	4	3	4	4
Level 2	3	4	3	3	3	3	4	4	3	3	3	3
Level 3	3	3	4	3	4	4	3	3	3	4	3	3

5 Optimal designs: extension to customer panels

There are many practical benefits of using consumer panels in conjoint studies. With a relatively homogeneous sample of respondents the experiment requires a few profile evaluations per respondent, reducing fatigue and learning effects. Homogeneous respondents may have identical preferences, but each one of them might report their utility ratings with a random origin of coordinates. In other words, the response measure is an interval scale rather than a ratio scale in the taxonomy of Stevens (1951). CA researchers can handle this problem introducing heterogeneous intercepts. However, if we take a small number of measures for each individual, we cannot estimate the specific value of the intercept for each one. Alternatively, we can handle the problem using a random effects model. For $i = 1, ..., N$ respondents, and T_i questions per individual, we consider the model

$$y_{it} = \eta_i + f\left(x_{it}\right)' \beta + \epsilon_{it} \tag{4}$$

where η_i are exogenous random variables with mean 0 and variance σ_η^2. If we include this effect in a overall shock $u_{it} = \eta_i + \epsilon_{it}$, then the autocovariance matrix for each respondent is $E(u_i u_i') = \Omega = \left(\sigma_\eta^2 11' + \sigma_\epsilon^2 I_T\right)$ has a special structure with $\sigma_\eta^2 + \sigma_\epsilon^2$ as diagonal elements, and σ_η^2 otherwise. Finally, the panel is balanced if $T_i = T$ for each respondent (we assume this to simplify notation).

Consider the matrix notation $X = (f(x_{11})', \ldots, f(x_{1T})', \ldots, f(x_{N1})', \ldots, f(x_{NT})')' \in \mathbb{R}^{NT \times p}$, the vector of responses $Y = (y_{11}, \ldots, y_{1T}, \ldots, y_{N1}, \ldots, y_{NT}) \in \mathbb{R}^{NT \times 1}$ and $u \in \mathbb{R}^{NT \times 1}$ analogously to Y. Then we can estimate consistently by OLS using $\widehat{\beta} = \left(X'X\right)^{-1} X'Y$. But this estimation is quite inefficient, as $Var\left[u\right] = (I_N \otimes \Omega)$. The method is not even consistent if η_i is correlated with some regressor (e.g. a socio-demographic block factor). In this Section we apply our approach for construction of exact optimal designs in the context of conjoint panels. We consider two of the most popular ways to estimate panels (both consistent even when endogenous random effects are intrinsically eliminated.) With a panel of consumers question repetitions are a concern only for an individual respondent. We can avoid them by introducing a lower bound on distances between product profiles for every individual. However, we do not forbid question repetitions across individuals.

5.1 Within-Groups (WG) estimation

One commonly used way to get rid of individual-specific effects, η_i, is to subtract time averages from the original panel model (4), leading to the within-groups (WG) model

$$\ddot{y}_{it} = \ddot{f}(x_{it})'\beta + \ddot{\epsilon}_{it}$$

where $\ddot{y}_{it} = y_{it} - \bar{y}_i$, $\ddot{f}(x_{it}) = f(x_{itk}) - \overline{f(x_{ik})}$, and $\ddot{\epsilon}_{it} = \epsilon_{it} - \bar{\epsilon}_i$. Stacking the observations for all individuals, such that $Y = (y'_{1t}, y'_{2t}, \dots, y'_{Nt})'$, $X = (f(x_{1t})', f(x_{2t})', \dots, f(x_{Nt})')'$, and $\epsilon = (\epsilon'_{1t}, \epsilon'_{2t}, \dots, \epsilon'_{Nt})'$ equivalent compact form model is

$$MY = MX\beta + M\epsilon$$

with $M = I_{NT} - \left(I_N \otimes \frac{1}{T} 1_T 1'_T\right) = I_{NT} - P$. Both M and P are idempotent matrices, and premultiplication by the matrix M creates deviation from the mean. We obtain mean centered data and the individual effect η_i disappears (because $\bar{\eta}_i = \eta_i$). Then OLS estimator is $\hat{\beta}_{OLS} = (X'MX)^{-1} X'MY$, with the variance

$$Var\left(\hat{\beta}_{OLS}\right) = \sigma^2_\epsilon (X'MX)^{-1}.$$

Assuming vector preferences the design matrix is $X = f(x) = x$, and with a symmetric, constant matrix M we can directly apply the results of Proposition 3. Table 11 presents the analytical derivatives for the WG problem.

Table 11: Analytical derivatives for the WG problem.

Objective	$\min_X \ \text{tr} \ (X'MX)^{-1}$
Gradient	$-2 \ \text{vec} \ MX (X'MX)^{-2}$
Hessian	$4 \left[(X'MX)^{-1} \otimes MX (X'MX)^{-2} X'M\right] +$ $4 \left[(X'MX)^{-2} \otimes MX (X'MX)^{-1} X'M\right] -$ $2 \left[(X'MX)^{-2} \otimes M\right]$

5.2 Estimation based on differences

Another way to get rid of individual effect η is to take increments, so that

$$\Delta y_{it} = \Delta X'_{it}\beta + \Delta\epsilon_{it},$$

where $\Delta y_{it} = y_{it} - y_{i(t-1)}$, $\Delta X_{it} = \Delta f(x_{it}) = f(x_{it}) - f\left(x_{i(t-1)}\right)$, and $\Delta\epsilon_{it} = \epsilon_{it} - \epsilon_{i(t-1)}$. Define the matrices

$$\Delta_T = \begin{bmatrix} -1 & 1 & 0 & \dots & 0 \\ 0 & -1 & 1 & \dots & 0 \\ & & \dots & & \\ 0 & \dots & 0 & -1 & 1 \end{bmatrix}_{(T-1)\times T}, \quad H = \begin{pmatrix} 2 & -1 & \dots & 0 & 0 \\ -1 & 2 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \ddots & 2 & -1 \\ 0 & 0 & \dots & -1 & 2 \end{pmatrix}_{(T-1)\times(T-1)}.$$

Now, let's stack in a column the observations for N individuals to obtain a compact form of the transformed model

$$DY = DX\beta + D\epsilon,$$

with a differentiation matrix $D = (I_N \otimes \Delta_T)$. To estimate the resulting model efficiently we have to apply GLS since $\{\Delta\epsilon_{it}\}$ follows a non invertible MA(1), which implies that $E\left[\Delta\varepsilon_i \Delta\varepsilon'_i\right] = \sigma^2_\varepsilon H$. The GLS estimator with N customers and T questions for each one is

$$\hat{\beta} = \left(\sum_{i=1}^{N} \sum_{t=2}^{T} \Delta X_{it} H^{-1} \Delta X_{it}' \right)^{-1} \sum_{i=1}^{N} \sum_{t=2}^{T} \Delta X_{it} H^{-1} \Delta y_{it} = \left(X' D' \mathbf{H}^{-1} D X \right)^{-1} X' D' \mathbf{H}^{-1} D Y,$$

where $\mathbf{H} = (I_N \otimes H)$ is analogous to H but with dimension $N(T-1)$. Notice that $\hat{\beta}$ is an unbiased estimator with non singular variance

$$Var\left(\hat{\beta}\right) = \sigma_{\varepsilon}^2 \left(\sum_{i=1}^{N} \sum_{t=2}^{T} \Delta X_{it} H^{-1} \Delta X_{it}' \right)^{-1} = \sigma_{\varepsilon}^2 \left(X' D' \mathbf{H}^{-1} D X \right)^{-1},$$

An exact optimal design for this method should minimize $\phi \left[\left(X' D' \mathbf{H}^{-1} D X \right)^{-1} \right]$.

The analytical derivatives to minimize the trace in the GLS problem are explicitly given in Proposition 3, because $Z = D' H^{-1} D$ is a constant matrix. Table 12 presents the solution to the classic conjoint model with vector preferences.

Table 12: Analytical derivatives for the GLS estimator in a differenced model.

Objective	$\min_X \ \text{tr} \left(X' D' H^{-1} D X \right)^{-1}$
Gradient	$-2 \ \text{vec} \ D' H^{-1} D X \left(X' D' H^{-1} D X \right)^{-2}$
Hessian	$4 \left[\left(X' D' H^{-1} D X \right)^{-1} \otimes D' H^{-1} D X \left(X' D' H^{-1} D X \right)^{-2} X' D' H^{-1} D \right] +$ $4 \left[\left(X' D' H^{-1} D X \right)^{-2} \otimes D' H^{-1} D X \left(X' D' H^{-1} D X \right)^{-1} X' D' H^{-1} D \right] -$ $2 \left[\left(X' H^{-1} X \right)^{-2} \otimes D' H^{-1} D \right]$

5.3 Numerical results

We report some simulations for panels, analogous to the ones reported in the Section 3.2, where we considered a single consumer. Here we simulate a panel with $N = 10$ consumers, each of whom is shown 5 stimuli profiles (T). The remaining simulation parameters can be found in Table 3.

Table 13 presents the results of a simulation for the conjoint panel estimated with WG and a GLS-in-differences approach. We run the algorithm 100 times with random initial points. Comparing optimal designs for the individual and the panel case, the most interesting result is the significant improvement in the trace optimality measure, with a relatively small increase of optimization cost. The algorithm converges in seconds in all cases, and the conditioning of the full-rank design matrix is good. The performance of WG and GLS-in-differences approaches is similar. The within-groups optimal design is faster to compute, but the objective function is worse, as we expected. The problem for GLS in differenced models is slower, but the solution is more stable - the minima lie very close.

Table 13: Simulation results for conjoint panels

	WG model			GLS in differences model		
Objective function	min tr $(X'MX)^{-1}$			min tr $\left(X'D'H^{-1}DX\right)^{-1}$		
problem size	small	medium	large	small	medium	large
# of distinct solutions[a]	57	62	66	23	64	68
# algorithm success	100%	100%	100%	100%	100%	100%
Trace[b]:						
Minimum value	0.00309	0.00516	0.00829	0.00028	0.00076	0.00187
25% percentile	0.00311	0.00519	0.00833	0.00028	0.00079	0.00192
50% percentile	0.00312	0.00521	0.00835	0.00029	0.00080	0.00194
75% percentile	0.00314	0.00522	0.00838	0.00029	0.00082	0.00196
Maximum value	0.00320	0.00530	0.00847	0.00043	0.00087	0.00203
Determinant:						
Minimum value	1.09e-09	1.16e-15	1.29e-24	4.81e-13	4.28e-20	2.48e-30
25% percentile	1.10e-09	1.18e-15	1.32e-24	5.75e-13	5.04e-20	3.53e-30
50% percentile	1.11e-09	1.19e-15	1.33e-24	6.41e-13	5.49e-20	3.97e-30
75% percentile	1.12e-09	1.20e-15	1.35e-24	6.79e-13	6.13e-20	4.55e-30
Maximum value	1.16e-09	1.24e-15	1.44e-24	1.01e-12	1.01e-19	6.44e-30
# iterations[c]	19	18	20	49.5	71.5	99.5
# function evaluations[c]	20	19	21	51	75	103.5
Time[c] (s)	0.44	0.91	2.69	1.08	3.78	13.73
Rank[c]	3	5	8	3	5	8
Condition[c]	1.26	1.35	1.45	2.86	4.44	6.19

[a] The objective function values are rounded to the 6th decimal place.
[b] Bordered blocks correspond to objective function values.
[c] Reported values are medians of all successful runs.

6 Conclusions

The current methods to design optimal experimental designs are typically inappropriate in CA, because they reduce treatments to a few points with multiple repetitions of the same attribute profiles, and often some of the extreme vertices are dangerous to use or expensive. In these cases, the optimal design is not implemented, but it should be computed to be used as a reference to measure the efficiency of the implemented designs.

This paper propose a general approach to compute the optimal matrix X^* with Newton type methods, avoiding repeated product profiles. Implementation results confirm the suitability of our approach to CA. We discuss cases with continuous and categorical product attributes, models for a single respondent and a panel of respondents.

The proposed procedure has the following advantages: (1) it is flexible to construct discrete-continuous designs; (2) it is easily implemented to the case of partial profiles in high dimensions; (3) the approach can easily handle other alternative linear regression estimators such as Stein's Shrinkage, ridge regression or LASSO estimators; and (4) it can handle estimators which are invariant to monotonous transformations in the measurement scale, even with ranking data. Below we briefly review each of these issues.

6.1 The discretization of continuous attributes

In many CA models, we often find continuous attributes reformulated as discrete ones. For example, prices are sometimes included as a continuous variable, but often just a reduced set of 3-4 alternative prices are included in the model. What is the rationale for this type of specifications? In this section we introduce some remarks about this approach.

Discrete attributes are often introduced by the researchers as a way to approximate a non linear function effect of a continuous attribute. Assume that

$$y_t = \beta_0 + \beta_1 \; f_1 \left(z_{1t} \right) + ... + \beta_k \; f_k \left(z_{Kt} \right) + \epsilon_t,$$

where z_t are continuous attributes and some of the functions in $f\left(\cdot\right)$ are unknown. If $f_i\left(z_i\right)$ is unknown and we build a partition $\left\{A_{ij}\right\}_{j=1}^{k_i}$ of the range of variation of z_i, we can approximate $f_i\left(z_i\right)$ by a simple function,

$$f_i \left(z_i \right) \simeq \sum_{j=1}^{k_i} \alpha_{ij} \; d_{ij},$$

where $d_{ij} = I\left(z_i \in A_{ij}\right)$ is a dummy variable, and I is the indicator function (equal to 1 if the event occurs, and 0 otherwise). The CA model can be written as

$$y_t = \beta_0 + \sum_{i=1}^{K} \sum_{j=1}^{k_i} \beta_{ij} d_{ijt} + \varepsilon_t$$

where $\beta_{ij} = \beta_i \alpha_{ij}$. If we have a single attribute, and we omit the intercept to avoid multicollinearity, the OLS estimator $\widehat{\beta}_{ij}$ is just the mean of all the data y_t for which $z_{it} \in A_j$. This way to specify and estimate a nonlinear regression model is known as a *regressogram.* It is the regression equivalent to the histogram for a density function. It is the most basic nonparametric regression estimator, but to ensure consistency the partition must be thinner when the sample size increases at an appropriate rate. The general case with several attributes is a standard semiparametric model for linear in components specifications. Notice that we might apply the same logic to an unknown general utility function $f\left(z_{1t}, ..., z_{Kt}\right)$, then the nonparametric approximation would be given by the full factorial model.

Continuous attribute discretization are commonly used in CA but in a primitive sense: as a parametric model specification, which is sometimes problematic. Note that, if the number of levels is too low we have an over-smoothing, and if it is too high - an over-fitting problem. This is a well known problem in nonparametrics and semiparametrics. The impact of the number of levels over CA estimations was pointed out by Currim, Weinberg, and Wittink (1981) and Wittink, Krishnamurthi, and Nutter (1982). There are other approaches that can render better results when the true f is smooth, such as orthogonal polynomials, etc. From a classical model perspective, this was the traditional method in surface response models, and certainly can be reconsidered as a nonparametric estimator (with the advantage that, for these models, there are much better tools for selecting the optimal level of smoothing than in the regressogram partitions). As a general rule we do not recommend the use of interval discretization to handle non linearities.

If we consider a more flexible parametric model with orthogonal polynomials, we can generally apply the methodology presented in this paper. For example, consider a simple case with regressors in $[0,1]$ and a Chebyshev Polynomials Basis $\left\{\phi_j\left(z\right) = \cos(j \arccos z)\right\}$, if we specify $f_i\left(z_i\right) = \sum_{j=1}^{k_i} \alpha_{ij} \; \phi_j\left(z_i\right)$. Then

$$y_t = \beta_0 + \sum_{j=1}^{k_1} \beta_{1j} \; \phi_j \left(z_i \right) + ... + \sum_{j=1}^{k_K} \beta_{Kj} \; \phi_j \left(z_{Kt} \right) + \epsilon_t,$$

and the optimal experimental design can be optimized with the presented methodology.

6.2 Partial profiles in high dimensions

Standard CA models assume that all the stimuli attributes affecting utility ratings are included in the model. But the product complexity has increased over time, and consumer preference models often have to analyze categories described by a massive number of attributes and levels. It is unfeasible to study all of them. Some researchers use partial profiles, where each profile contains an experimentally designed subset of the attributes, as discussed by Bradlow (2005). Sometimes adaptive questionnaires are used to select a few important attributes.

However, the omission of other significant regressors generates biased estimations. If $y_t = f(x_t)' \beta + U_t + \epsilon_t$, where $U_t = \gamma' f(z_t)$ is the utility associated to the omitted attributes z_t. If we omit the attributes $f(Z)$, estimating the model $y_t = f(x_t)' \beta + \epsilon_t$ by OLS, some issues must be taken into account. First, notice that question repetitions in X can be accepted provided that omitted attributes are changing. But in general only the most important attributes are included in the model, determined through exploratory direct assessment of attribute importances by the respondents. Therefore there is no need for repeated questions and it is convenient to include distance constraints on presented stimuli discussed in this paper. The second and more crucial issue, is that OLS estimators of the model with omitted variables is in general biased,

$$E\left[\widehat{\beta}\right] = \beta + Q_T^{-1} X' U.$$

with $U = (U_1, ..., U_T)'$. However, if we generate product profiles $\{(x_t, z_t)\}$, including the constraint that $X' f(Z) = 0$, we can avoid the bias problem and the covariance matrix of $\widehat{\beta}$ will be determined by $(X'X)^{-1}$ under the standard assumptions.

6.3 Alternative linear regression estimators

The Gauss-Markov Theorem ensures that OLS are the best linear unbiased estimators [BLUE], conditionally on the design matrix X. However, not all the design matrices render equally efficient estimators. Given these properties, we have focused on optimal experimental designs for OLS estimators, but the same method can be adapted to other increasingly popular estimators, such as Bayesian estimators for Gaussian Linear Regression. The classical model assumes that $Y \sim N\left(X'\beta, \sigma^2 I\right)$ with conjugate prior $\beta|\sigma^2 \sim N(\mu, \Sigma)$ and $1/\sigma^2$ distributed as a Gamma. In this case, β has a posterior distribution normal with $E\left(\beta|Y, \sigma^2\right) = \left(\Sigma^{-1} + X'X\right)^{-1} \left(\Sigma^{-1}\mu + X'Y\right)$ and covariance matrix $Var\left(\beta|Y, \sigma^2\right) = \sigma^2 \left(\Sigma^{-1} + X'X\right)^{-1}$. The trace (or determinant) of $\left(\Sigma^{-1} + X'X\right)^{-1}$ can be minimized similarly to the trace (determinant) of $(X'X)^{-1}$ in OLS, subject to the required constraints preventing profile repetitions. Notice that in any case, the choice between OLS and the Classical Bayesian method is irrelevant with large T, as the distance between $E\left(\beta|Y, \sigma^2\right)$ and the OLS estimator converges faster than $\sqrt{T}$. Even if the researcher considers another prior distribution (computing numerically the posterior), the Bernstein-von Mises Theorem ensures that the Bayes distribution a posteriori behaves asymptotically like the Maximum Likelihood estimator, under appropriate regularity conditions. With normal likelihood this estimator is precisely OLS. Therefore, the choice between Bayes or OLS matters essentially for relatively small T, which is precisely where the prior assumption has more impact.

The method can be also adapted to handle Stein's Shrinkage estimators that can have a smaller Mean Square Error (MSE) that OLS, reducing the variance in exchange for a small bias sacrifice. For example, we can consider a Ridge regression estimator $\widehat{\beta} = (X'X + \gamma I)^{-1} X'Y$, which minimizes $\|Y - X'\beta\|_2^2 + \gamma \|\beta\|_2^2$ where the parameter γ is set as a minimizer of the MSE trace or determinant conditionally on the data. Essentially the method penalizes complex models, and has a Bayesian interpretation. Another example is the LASSO estimator, similar to Ridge but replacing $\|\beta\|_2^2$ by the norm $\|\beta\|_1$. In experimental context, these methods are can be applied to handle collinearity problems, and the γ is selected as a function of X. The algorithms considered in this paper can be readily adapted to these estimators, minimizing the trace or determinant of the appropriate covariance matrix.

6.4 Ranking data: looking for invariance to monotonous transformations

Conceptually, a monotonous transformation of a utility function does not change the associated preorder of preferences. A drawback of CA methods based on regression models is that conditional means are not invariant to monotonous transformations of the response variable. Assume that $E[y|x] = f(x)' \beta$, then given a monotonous transformation h, in general

$$E[h(y)|x] \neq h(E[y|x]) = h\left(f(x)' \beta\right).$$

If the CA analysis is based on ratings, this is a nuisance but to some extent we can swallow it. But if the analysis is based on preference ordering, albeit OLS is a valid method to estimate $E[y|x]$, it is quite hard to accept the lack of invariance in the estimated utility function.

In this section we consider the question: is there any method to build CA models, which is invariant to monotonous transformations of the response variable? The answer is positive: the conditional median or 0.5 conditional quantile. Recall that the α-quantile of the conditional distribution of y is,

$$F_{y|x}^{-1}(\alpha) = inf\left\{t : \Pr(y \leq t|x) \geq \alpha\right\}.$$

Assuming that the conditional median is linear in parameters:

$$F_{y|x}^{-1}(0.5) = f(x_t)'\gamma$$

(it happens under normality, but also for other distributions) we can consider the 0.5-quantile regression

$$y_t = f(x_t)'\gamma + \epsilon_t$$

where $\epsilon|x$ has a conditional density function $g(\cdot|x)$ with zero median. Notice that quantiles are invariant to monotonous transformations, so that

$$F_{h(y)|x}^{-1}(0.5) = h\left(F_{y|x}^{-1}(0.5)\right) = h\left(f(x)'\gamma\right).$$

Another advantage of quantile regression is that the quantiles are identifiable under censure. For example, in CA using a positive ratio scale of preferences we would censure all products with disutility (negative ratings), as we only observe $y^c = \max\{0, y\}$. The quantile regression here is $F_{y^c|x}^{-1}(0.5) = \max\left\{0, F_{y|x}^{-1}(0.5)\right\} = \max\left\{0, h\left(f(x)'\gamma\right)\right\}$. By contrast, conditional mean of censured variables are only identifiable with additional distributional assumptions (e.g., a Tobit model).

How can we estimate the conditional median? Under regularity conditions, the Least Absolute Deviation (LAD) estimator minimizing $\sum_{t=1}^{T}\left|y_t - f(x_t)'\gamma\right|$ is a consistent estimator of γ. The LAD estimator can be computed solving the linear programing problem,

$$\begin{aligned}
&\min_{\{\gamma,u_1,\ldots,u_T\}} \sum_{t=1}^{T} u_t \\
&s.t. \\
&\qquad u_t \leq y_t - f(x_t)'\gamma,\ t = 1,\ldots,T \\
&\qquad u_t \leq -\left(y_t - f(x_t)'\gamma\right),\ t = 1,\ldots,T
\end{aligned}$$

easy to solve even with popular computational spreadsheets such as Microsoft Office Excel. The constraints force $u_t = |\widehat{\epsilon}_t|$ in the optimum. We can compare the output with the standard regression. If $y_t = f(x_t)'\beta + \varepsilon_t$ with $E[\varepsilon|x] = 0$, and conditional distribution of $y|x$ is symmetric, i.e. is $g(\cdot|x)$ symmetric in 0 for all x, then $\gamma = \beta$ the parameters of the 0.5 quantile regression are identical to the parameters of the standard linear regression model, and OLS and LAD are two alternative estimators for the same parameters. OLS is more efficient, but less robust to outliers in y_t. But, if the conditional distribution of y is asymmetric the differences can be crucial. An asymmetric distribution can be easily obtained when setting the preference measure scale in the questionnaire. The asymptotic behavior of LAD estimators is studied by Bassett and Koenker (1978). Our experimental design method can be adapted to conditional quantile estimators. Notice that the asymptotic covariance matrix of LAD estimators can be consistently estimated by

$$V_T = \frac{1}{4}\left(\sum_{t=1}^{T} g(0|x_t) f(x_t) f(x_t)'\right)^{-1} (X'X) \left(\sum_{t=1}^{T} g(0|x_t) f(x_t) f(x_t)'\right)^{-1}.$$

We will consider the optimal design minimizing the trace of this matrix. In particular, if $g(0|x)$ is independent of x, we obtain

$$V_T = \frac{1}{4g(0)^2}\left(\sum_{t=1}^{T} f(x_t) f(x_t)'\right)^{-1} = \frac{1}{4g(0)^2} Q_T^{-1}.$$

Therefore, in order to minimize $\phi(V_T)$ we can use the same optimal designs minimizing $\phi\left(Q_T^{-1}\right)$, which we have proposed for Least-Squares estimators. The approach can be implemented in the context of LAD estimation. Additionally, if $g(0|x)$ is a known function of x, we can adapt our approach to minimize the trace of V_T.

7 Appendix A: Approximate optimal designs

In this section we review the tools available for the design of approximate optimal experiments, and the drawbacks for their application to CA experiments.

What can we say about the matrix Q? We first consider the case with a finite number of explanatory variables (or treatments), $\chi = \{x_1, ..., x_r\}$, meaning that the attributes are described by dummy variables. With $T > r$ some of the treatments are replicated, and let T_j denote the number of replications of treatment x_j. Then we can write $Q_T = \sum_{j=1}^{r} (T_j/T)\ f(x_j) f(x_j)'$, with $T = \sum_{j=1}^{r} T_j$, and the limit matrix must be of the form

$$Q = Q_\omega = \sum_{j=1}^{r} \omega_j\ f(x_j) f(x_j)',$$

where $\{\omega_j\}$ are limit relative frequencies that sum up one. Notice that the optimal ω_j are the continuous approach to treatments' relative frequencies T_j/T. An exact design for a given sample size T puts emphasis on setting T_j, and an approximate design on setting ω_j in the continuous limit (either generating T random profiles based on these probabilities so that and $Q = E_\omega \left[f(x) f(x)'\right]$, or setting an exact integer number T_j such that T_j/T is close to ω_j).

The theory of approximate designs was developed by Jack Carl Kiefer and his school (Kiefer 1959). They proposed to select optimally ω_j, minimizing some convex function measuring the size of Q_ω^{-1}. The most common procedures minimize:

1. generalized variance: $\left|Q^{-1}\right| = \prod_{r=1}^{k} \lambda_r\left(Q^{-1}\right) = \prod_{r=1}^{k} 1/\lambda_r(Q)$ where $\lambda_r(Q), r = 1, ..., k$ the eigenvalues of Q. Equivalently, the logarithm can be considered. D-optimality criterion minimizes the volume of the confidence ellipsoid of the model parameters. It is probably the most popular method;

2. average variance: $tr\left(Q^{-1}\right) = \sum_{r=1}^{k} \lambda_r\left(Q^{-1}\right) = \sum_{r=1}^{k} 1/\lambda_r(Q)$. A-optimality criterion (average-variance optimality) minimizes the mean of the variances of the estimates; and

3. worst possible prediction error: $d\left(Q^{-1}\right) = \max_{x\in\chi}\left\{xQ^{-1}x\right\}$. This is sometimes denoted G-optimality. The celebrated Kiefer-Wolfowitz equivalence theorem proved that G-optimal and D-optimal designs are exactly the same.

4. the largest eigenvalue: $\max_r \left\{\lambda_r\left(Q^{-1}\right) =\right\} = \max_r \{1/\lambda_r(Q)\}$, called E-optimality or eigenvalue optimality.

More generally, we can minimize any non-negative function $\phi\left(Q^{-1}\right)$, provided that it is (1) positively homogeneous: $\phi(\delta A) = \delta\phi(A)$ for $\delta > 0$ to ensure that the factor σ^2/T is common to all designs; (2) non-increasing: $\phi(A) \leq \phi(B)$ when $(A - B)$ is non negative definite; and (3) convex (to ensure that ϕ satisfies the condition that information cannot be increased through interpolation). This approach was developed by Kiefer (1959) inspired by the suggestion of Wald (1943) to compare designs using D-optimality, see also Kiefer and Wolfowitz (1960). Sometimes we are just interested in a subset or a combination of insightful coefficients, say $C\beta$ with a non singular matrix C. Then the optimal design minimizes the size of the corresponding covariance $\phi(CQC')$ (Hausman 1982; Toubia and Hauser 2007). In particular, the L-optimality criteria minimizes $tr\left(CQ^{-1}\right)$ for an appropriate matrix C.

Following the Kiefer approach, we can randomly generate the designs with optimal probabilities ω^*, by minimizing a convex function ϕ

$$\min_\omega \phi\left(Q_\omega^{-1}\right) = \min_\omega \phi\left(\left[\sum_{j=1}^{r} \omega_j\ f(x_j) f(x_j)'\right]^{-1}\right) \tag{5}$$

subject to the constraint that ω is in the $\mathbb{R}_+^r$ simplex. We can add other convex constraints, e.g. a bound on the expected experiment cost $T \cdot c'\omega \leq m$ where m is the available budget, and c is a $r \times 1$ vector, whose elements are costs associated with each treatment in χ, so that the expected cost of a single profile is $c'\omega$.

Instead of generating random designs with distribution ω^*, we can consider appropriate integer numbers T_j of repetitions, such that the optimal ω_j^* is approximated by T_j/T (Pukelsheim and Rieder 1992). Approximate designs are convenient from a theoretical and computational perspective, but in practice the results must be rounded off leading to the loss of design efficiency. Alternatively, we can try to optimize $\phi\left(Q_T\right)$ in $\{T_j\}$ directly. Exact designs are sometimes used but the integer optimization is generally more difficult.

For continuous treatments, we can generate treatments randomly from a probability distribution w, and consider a limit information matrix

$$Q\left(w\right)=\int f\left(x\right)f\left(x\right)'w\left(dx\right)\in\mathbb{R}^{p\times p}$$

where w is a probability distribution on χ. We need to select the optimal probability function. In practice this problem becomes similar to the case with finite number of treatments, focusing on a few extreme cases. This makes sense, as the extreme conditions in experiments usually render more information for inference decisions. The following result provides a theoretical basis for this and some more general statements.

Lemma 1 *If χ is a convex compact set and f preserves convexity, then any feasible Q can be expressed as $\sum_{x_j\in\chi^e}\omega_j f\left(x_j\right)f\left(x_j\right)'$ where ω_j are discrete probabilities and χ^e is the set of extreme points of $\left\{f\left(x\right)f\left(x\right)':x\in\chi\right\}$.*

Proof. The set $\mathbf{Q}=\left\{Q=Q\left(w\right):w\geq0,\int dw=1\right\}$ is isomorphic to a convex and compact set in $\mathbb{R}^{p(p+1)/2}$. The Carathéodory's Theorem[2] guarantees that any $Q\in\mathbf{Q}$ can be achieved by a design w with no more than $1+p\left(p+1\right)/2$ points in its support. But we can obtain a more explicit representation. The classical Krein–Milman Theorem ensures that if $\mathbf{Q}$ is a compact convex set of $\mathbb{R}^{p(p+1)/2}$, then any $Q\in\mathbf{Q}$ can be expressed as $\sum_{x_j\in\chi^e}\omega_j f\left(x_j\right)f\left(x_j\right)'$ where $\sum_j\omega_j=1$ with $\omega_j\geq0$, and χ^e is the set of extreme points of $\left\{f\left(x\right)f\left(x\right)':x\in\chi\right\}$. ■

Therefore, the search for optimal designs may be restricted to designs with a finite support. If $\left\{f\left(x\right)f\left(x\right)':x\in\chi\right\}$ is a convex polygon in $\mathbb{R}^{p(p+1)/2}$ the first step is to compute the vertices, the second consists of solving a problem similar to (5), considering a frequency of repetitions for each vertex. Obviously, mixed models with continuous and discrete variables can be handled alike. These results can be directly adapted to experiments with heteroskedasticity, where $E(\epsilon\epsilon')=diag\left(\sigma^2\left(x_t\right)\right)$, considering information matrices $Q\left(w\right)=\int\sigma^2\left(x\right)f\left(x\right)f\left(x\right)'w\left(dx\right)$, and $Q=\sum_{j=1}^{r}\omega_j\sigma^2\left(x_j\right)f\left(x_j\right)f\left(x_j\right)'$. But in practice this cannot be applied unless we know $\sigma^2\left(x\right)$, to that end we can build a preliminary experiment to estimate this function but this is rarely considered.

In order to compute the approximate optimal designs solving (5), Kiefer's school has considered several algorithms. One of the most popular is the classical algorithm proposed by Fedorov-Wynn for D-optimality (Fedorov 1972; Wynn 1970), for the review see St. John and Draper (1975) and the references in Atwood (1973, 1976). These methods are variants from the steepest descent method algorithm, and they can be adapted for other criteria $\phi\left(\cdot\right)$ (Whittle 1973). However, the steepest descent methods converge very slowly. Atwood (1976) considers faster Newton directions. But the performance of these methods is not always good, and the search for optimal designs often restricts to low-dimensional models. As López Fidalgo (2009) states: "One may think the people working on optimal design must be good in optimization. They are not bad, but they are not experts in the topic. At the same time, people in optimization are sometimes far from statistics and even more from experimental designs. Therefore, there is a need of more cooperation between them." Several contemporary *constrained optimization* numerical algorithms can be implemented for a faster computation of ω^*, including classical *sequential quadratic optimization* algorithms, or the more recent *interior point algorithms*, (see e.g. Vandenberghe, Boyd, and Wu 1998; Boyd and Vandenberghe 2004, Ch.7). Solving the dual problem is a good strategy that often renders faster results.

Unfortunately, this approach is not adequate for CA. If the researchers use Kiefer's approximate design for a single customer, ω_j (respectively T_j) can be interpreted as the probability (absolute frequency) of times stimulus j is repeated. This is an entirely undesirable situation: if implemented, the repeated questions should be interspersed and presented separately over time to ensure that the respondent forgets the previous

[2] The Carathéodory's theorem states that if $y\in\mathbb{R}^d$ lies in the convex hull of a set P, there is a subset $P'\subset P$ consisting of no more than $d+1$ points such that y lies in the convex hull of P'.

answers. Even then, the procedure could easily be cost-prohibitive and tedious for the respondent, leading to biased estimations[3]. Therefore, approximate optimal designs should not be implemented in CA in general.

8 Appendix B: Matrix derivatives

This section presents the main results about matrix derivatives. First we introduce some concepts about functions of matrices and their derivatives. Let Z denote a $n \times q$ real matrix. We can consider a $m \times p$ real matrix valued function $\Phi(Z)$ (notice that scalar valued and vector valued functions are a particular case). We define the Jacobian of $\Phi(Z)$ as the $mp \times nq$ matrix

$$D\Phi(Z) = \frac{\partial vec\left(\Phi(Z)\right)}{\partial\left(vec\left(Z\right)\right)'}.$$

Using this definition, the properties of classical gradients and Jacobians are preserved. The differential of $\Phi(Z)$ will be given by $d\ \Phi(Z) = \Phi(Z)\ \ d\ vec\ Z = \Phi(Z)\ \ vec\left(dZ\right)$. Hessians, can be defined analogously as follows,

$$H\Phi(Z) = D\left(D\Phi(Z)\right)' = \frac{\partial}{\partial\left(vec\left(Z\right)\right)'} vec\left(\frac{\partial vec\left(\Phi(Z)\right)}{\partial\left(vec\left(Z\right)\right)'}\right)'.$$

The classical case where Φ is vector or scalar valued, is a particular case under this notation. For a detailed introduction to matrix derivatives see Magnus and Neudecker (1999).

8.1 Main derivatives

Consider a $T \times k$ design matrix, $X = f(x)$, where $f(\cdot)$ is a twice differentiable function. Let

$$\begin{aligned} A &= \frac{\partial vec\left(f\left(x\right)\right)}{\partial\left(vec\left(x\right)\right)'} \\ B &= \frac{\partial}{\partial\left(vec\left(x\right)\right)'} vec\left(\frac{\partial vec\left(f\left(x\right)\right)}{\partial\left(vec\left(x\right)\right)'}\right)'. \end{aligned}$$

the Jacobian and Hessian of f, and let Z be a constant positive definite weight matrix. We assume that Z is symmetric to simplify the notation, otherwise the derivatives become more involved. For example for the classic experimental regression model, with vector utility preferences $f(x) = x$, $A = I$, $B = 0$, and $Z = I$. Finally, let's define a commutation matrix K, such that vec $X' = K$ vec X.

Proposition 2 *Consider the objective function*

$$\min\left|\left(X'ZX\right)^{-1}\right|.$$

The gradient and Hessian are respectively, in a vec form,

$$\begin{aligned} D\phi(X) &= -2\left|\left(X'ZX\right)^{-1}\right|\ \ vec\ AZX\left(X'ZX\right)^{-1}, \\ H\phi(X) &= 4\left|\left(X'ZX\right)^{-1}\right| K\left(AZX(X'ZX)^{-1} \otimes (X'ZX)^{-1}X'ZA'\right) + \\ & \quad 4\left|\left(X'ZX\right)^{-1}\right|\left(\left(X'ZX\right)^{-1} \otimes AZX\left(X'ZX\right)^{-1} X'ZA'\right) - \\ & \quad 2\left|\left(X'ZX\right)^{-1}\right|\left(\left(X'ZX\right)^{-1} \otimes AZA'\right) - \\ & \quad \left|\left(X'ZX\right)^{-1}\right|\left(ZX\left(X'ZX\right)^{-1} \otimes B + \left(X'ZX\right)^{-1} X'Z \otimes B'\right). \end{aligned}$$

[3] We can apply directly the Kiefer method for a homogeneous consumer sample, were we ask just one question to each different respondent. Then the optimal frequencies ω^* can be used for randomization, allocating different respondents to an specific question.

Proof. Note that the general first order derivative of $|X|$ is $d|X| = |X|\text{tr } X^{-1}dX$, and the general first order derivative of the inverse is $X^{-1} = -X^{-1}(dX)X^{-1}$. Now recall main properties of trace: is invariant under cyclic permutations, the traces of a matrix and its transpose are equal, and additivity. The differential of $\left|(X'ZX)^{-1}\right|$ is

$$\begin{aligned} d\left|(X'ZX)^{-1}\right| &= \left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)\, d\,(X'ZX)^{-1} = \\ &= -\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\, d\,(X'ZX) = \\ &= -2\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\, X'ZdX = \\ &= -2\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\, X'ZA'dx. \end{aligned}$$

Then the first order derivative is

$$D\phi(X) = -2\left|(X'ZX)^{-1}\right| AZX\,(X'ZX)^{-1}.$$

According to the first identification table (Magnus and Neudecker 1999, p. 176), the gradient in vec form is $-2\left|(X'ZX)^{-1}\right|$ vec $AZX\,(X'ZX)^{-1}$.

Recall, one of the trace properties: (tr U)(tr V) = tr $U \otimes V$, where U and V are square matrices. Then consider the Hessian

$$\begin{aligned} d^2\left|(X'X)^{-1}\right| &= d\left[-2\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\, X'ZA'dx\right] = \\ &= -2\; d\left|(X'ZX)^{-1}\right| \cdot \text{ tr } (X'ZX)^{-1}\, X'ZA'dx \\ &- 2\left|(X'ZX)^{-1}\right| \text{ tr } d\,(X'ZX)^{-1}\, X'ZA'dx \\ &- 2\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\,(dX)'ZA'dx \\ &- 2\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\, X'Z(dA)'dx = \\ &= 4\left|(X'ZX)^{-1}\right| \left[\text{ tr } (X'ZX)^{-1}\, X'ZA'dx\right] \left[\text{ tr } (X'ZX)^{-1}\, X'ZA'dx\right] \\ &+ 2\left|(X'ZX)^{-1}\right| \text{tr } (X'ZX)^{-1}\, d\,(X'ZX)\,(X'ZX)^{-1}\, X'ZA'dx \\ &- 2\left|(X'ZX)^{-1}\right| \text{tr } (X'ZX)^{-1}\,(dx)'AZA'dx \\ &- 2\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\, X'Zdx'Bdx = \\ &= 4\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\, X'ZA'dx\; 1 \otimes (X'ZX)^{-1}\, X'ZA'dx \\ &+ 4\left|(X'ZX)^{-1}\right| \text{tr } (X'ZX)^{-1}\,(dx)'\, AZX\,(X'ZX)^{-1}\, X'ZA'dx \\ &- 2\left|(X'ZX)^{-1}\right| \text{tr } (X'ZX)^{-1}\,(dx)'AZA'dx \\ &- 2\left|(X'ZX)^{-1}\right| \text{ tr } (X'ZX)^{-1}\, X'Zdx'Bdx. \end{aligned}$$

Using the Kronecker property $\alpha \otimes A = \alpha A$, the Hessian is:

$$\begin{aligned} H\phi(X) = \quad & 4\left|(X'ZX)^{-1}\right| K\left(AZX(X'ZX)^{-1} \otimes (X'ZX)^{-1}X'ZA'\right) + \\ & 4\left|(X'ZX)^{-1}\right| \left((X'ZX)^{-1} \otimes AZX\,(X'ZX)^{-1}\, X'ZA'\right) - \\ & 2\left|(X'ZX)^{-1}\right| \left((X'ZX)^{-1} \otimes AZA'\right) - \\ & \left|(X'ZX)^{-1}\right| \left(ZX\,(X'ZX)^{-1} \otimes B + (X'ZX)^{-1}\, X'Z \otimes B'\right). \end{aligned}$$

∎

Proposition 3 *Consider the following objective function*

$$\min tr\;\left(X'ZX\right)^{-1}.$$

The gradient and Hessian are respectively in vec form

$$\begin{aligned}
D\phi(X) &= -2 \ vec \ AZX\left(X'ZX\right)^{-2} \\
H\phi(X) &= 4\left(\left(X'ZX\right)^{-1} \otimes AZX\left(X'ZX\right)^{-2} X'ZA'\right) + \\
& \quad 4\left(\left(X'ZX\right)^{-2} \otimes AZX\left(X'ZX\right)^{-1} X'ZA'\right) - \\
& \quad 2\left(\left(X'ZX\right)^{-2} \otimes AZA'\right) - \\
& \quad \left(ZX\left(X'ZX\right)^{-2} \otimes B + \left(X'ZX\right)^{-2} X'Z \otimes B'\right).
\end{aligned}$$

Proof. Using the main properties of the trace, the differential of tr $\left(X'ZX\right)^{-1}$ is

$$d \text{ tr } \left(X'ZX\right)^{-1} = - \text{ tr } \left(X'ZX\right)^{-2} d\left(X'ZX\right) = -2 \text{ tr } \left(X'ZX\right)^{-2} X'ZA'dx.$$

Following the identification table the first order derivative is

$$D\phi(X) = -2AZX\left(X'ZX\right)^{-2}.$$

and the gradient is simply the vec form of $D\phi(X)$.

For the Hessian, consider the second-order differential

$$\begin{aligned}
d^2 \text{ tr } \left(X'ZX\right)^{-1} &= d\left(-2 \text{ tr } \left(X'ZX\right)^{-2} X'ZA'dx\right) = \\
&= -2 \text{ tr } d\left(X'ZX\right)^{-2} X'ZA'dx \\
&\quad - 2 \text{ tr } \left(X'ZX\right)^{-2} (dX)'ZA'dx \\
&\quad - 2 \text{ tr } \left(X'ZX\right)^{-2} X'Z(dA)'dx = \\
&= 2 \text{ tr } \left(X'ZX\right)^{-1} d\left(X'ZX\right)\left(X'ZX\right)^{-2} X'ZA'dx \\
&\quad + 2 \text{ tr } \left(X'ZX\right)^{-2} d\left(X'ZX\right)\left(X'ZX\right)^{-1} X'ZA'dx \\
&\quad - 2 \text{ tr } \left(X'ZX\right)^{-2} (dx)'AZA'dx \\
&\quad - 2 \text{ tr } \left(X'ZX\right)^{-2} X'Zdx'Bdx = \\
&= 4 \text{ tr } \left(X'ZX\right)^{-1} (dx)'AZX\left(X'ZX\right)^{-2} X'ZA'dx \\
&\quad + 4 \text{ tr } \left(X'ZX\right)^{-2} (dx)'AZX\left(X'ZX\right)^{-1} X'ZA'dx \\
&\quad - 2 \text{ tr } \left(X'ZX\right)^{-2} (dx)'AZA'dx \\
&\quad - 2 \text{ tr } \left(X'ZX\right)^{-2} X'Zdx'Bdx.
\end{aligned}$$

Then according to the second identification table the Hessian is

$$\begin{aligned}
H\phi(x) = \quad & 4\left(\left(X'ZX\right)^{-1} \otimes AZX\left(X'ZX\right)^{-2} X'ZA'\right) + \\
& 4\left(\left(X'ZX\right)^{-2} \otimes AZX\left(X'ZX\right)^{-1} X'ZA'\right) - \\
& 2\left(\left(X'ZX\right)^{-2} \otimes AZA'\right) - \\
& \left(ZX\left(X'ZX\right)^{-2} \otimes B + \left(X'ZX\right)^{-2} X'Z \otimes B'\right).
\end{aligned}$$

■

8.2 Distance constraints

Proposition 4 *Consider the following distance constraints applied pointwise*

$$\left(1_{T\times T} - I_T\right)\epsilon - diag\left(XX'\right)\mathbf{1}' - \mathbf{1}diag\left(XX'\right)' + 2XX' \leq 0$$

The gradient of the constraints in a matrix form is

$$dC = -2\left(I_{T^2} + K_{T^2}\right)\left[\left(\mathbf{1}_{T\times 1} \otimes I_T\right)A - \left(X \otimes I_T\right)\right]$$

$$where \ A_{i.} = \left(vec \ e_i e_i' X\right)'.$$

Proof. The constraint can be written as

$$(1_{T\times T} - I_T)\,\epsilon - F - F' + 2S \leq 0$$

where $S = XX'$, and $F = diag(S)\mathbf{1}'$. In the constraint F, F' and S depend on X.

First, let's calculate the derivative of F. Note the the special structure of F (identical columns):

$$F = \begin{bmatrix} s_{11} & \dots & s_{11} \\ s_{22} & \dots & s_{22} \\ \dots & \dots & \dots \\ s_{tt} & \dots & s_{tt} \end{bmatrix}_{T\times T} = \begin{bmatrix} e_1'XX'e_1 & \dots & e_1'XX'e_1 \\ e_2'XX'e_2 & \dots & e_2'XX'e_2 \\ \dots & \dots & \dots \\ e_t'XX'e_t & \dots & e_T'XX'e_T \end{bmatrix}_{T\times T} = \sum_{i=1}^{T}\sum_{j=1}^{T} e_i'XX'e_iE_{ij}$$

where e_i is a unit vector containing 1 in the i-th element, and zeros otherwise. E_{ij} is an elementary matrix, containing 1 in the (i,j)-th element and zeros otherwise.

According to the first identification table (Magnus and Neudecker, p. 176), taking derivatives of a $T \times T$ matrix function $F(X)$ with respect to a $T \times k$ matrix X requires vectorizing both matrices

$$\text{d vec } F = A \text{ d vec } X \quad \Rightarrow DF(X) = A_{T^2\times Tk}.$$

Every row of a differential matrix A contains partial derivatives of each element of the vectorized F, taken with respect to vectorized X. Conveniently, in our case all columns are identical, therefore

$$\text{vec } F = (\mathbf{1}_{T\times 1} \otimes I_T)\, F_{.1}$$

is a column vector obtained by stacking T times first column of F. Each element of $F_{.1}$ is a scalar function of X, such that $F_{i1} = e_i'XX'e_i$, and its derivative is

$$d\ \phi(X) = d\,(e_i'XX'e_i) = e_i'(dX)X'e_i + e_i'X(dX)'e_i = 2 \text{ tr } X'e_ie_i'dX$$
$$\Leftrightarrow D\phi(X) = 2\,(\text{vec } e_ie_i'X)'\,.$$

Using the result from first identification table

$$\phi(X):\ \text{d}\phi = \text{tr } A'\text{d}X = (\text{vec } A)' \text{ d vec } X \Rightarrow D\phi(X) = (\text{vec } A)'\,,$$

we obtain following derivative of F:

$$DF = (\mathbf{1}_{T\times 1} \otimes I_T)\, F_{.1} = 2\,(\mathbf{1}_{T\times 1} \otimes I_T)\, A$$
$$\text{where} \quad A = \begin{bmatrix} (\text{vec } e_1e_1'X)' \\ (\text{vec } e_2e_2'X)' \\ \dots \\ (\text{vec } e_Te_T'X)' \end{bmatrix}.$$

It is straightforward to obtain the derivative of the second element, F', using the properties of vec operator

$$\text{vec } F' = K_{T^2} \text{ vec } F,$$

where K is a square commutation matrix. Then

$$d \text{ vec } F' = K_{T^2}d \text{ vec } F \Rightarrow DF' = K_{T^2}DF.$$

The last element in the constraint is $S = XX'$. If $S(X) = XX'$, then

$$dS(X) = (dX)X' + X(dX)'$$

and

$$\begin{aligned} \text{d vec } S(X) &= (X \otimes I_T) \text{ d vec } X + (I_T \otimes X) \text{ d vec } X' \\ &= (X \otimes I_T) \text{ d vec } X + (I_T \otimes X)\, K_{Tk} \text{ d vec } X \\ &= (X \otimes I_T) \text{ d vec } X + K_{T^2}\,(X \otimes I_T) \text{ d vec } X \\ &= (I_{T^2} + K_{T^2})\,(X \otimes I_T) \text{ d vec } X. \end{aligned}$$

Therefore

$$DS(X) = (I_{T^2} + K_{T^2})\,(X \otimes I_T)\,.$$

Finally, combining all three results, the derivative of constraint on the distance matrix is

$$\begin{aligned} DC &= -DF - K_{T^2}DF + 2DS = \\ &= -\,(I_{T^2} + K_{T^2})\,DF + 2DS \\ &= -2\,(I_{T^2} + K_{T^2})\,[(\mathbf{1}_{T\times 1} \otimes I_T)\,A - (X \otimes I_T)]\,. \end{aligned}$$

■

8.3 Interactions

In Section 4 we introduce a mixed-integer conjoint model with interactions. We assume that necessary transformations to eliminate collinearity in x have been made, and we added an intercept. Therefore $X = f(x) = xA+B$, with A, B - constant sparse matrices. Note that interactions between variable a and variable b can be written in terms of Kronecker product of columns corresponding to variables a, b, considered separately for each observation (row by row):

$$W = \sum_{t=1}^{T} e_t \left(e_t' f(x) E_a \otimes e_t' f(x) E_b\right)$$

where e_t is a unit vector with 1, in the position t and zeros otherwise, so that $e_t' f(x)$ takes the t-th row of $f(x)$. Matrices E_a, E_b are sparse, and post-multiplication by them selects columns of $f(x)$ corresponding to variables a, and b respectively. By simple transformations we can stack the interaction block W row-wise to $f(x)$

$$f_2(x) = f(f(x)) = f(x)D_1 + WD_2 = xAD_1 + BD_1 + WD_2$$

D_1, D_2 being constant sparse matrices which add block of zeros to the back and front of the matrix, and W is defined as a function of $f(x)$ above. Let $X = f_2(x)$, then applying the result of Proposition 3

$$\begin{aligned} d \text{ tr } \left(X'X\right)^{-1} &= -2 \text{ tr } \left(X'X\right)^{-2} X' dX = -2 \text{ tr } \left(X'X\right)^{-2} X' d\left(xAD_1 + BD_1 + WD_2\right) = \\ &= -2 \text{ tr } AD_1 \left(X'X\right)^{-2} X' dx - 2 \text{ tr } D_2 \left(X'X\right)^{-2} X' dW. \end{aligned}$$

The first element does not require any further calculations, so let's concentrate on the second element:

$$\begin{aligned} \text{tr } D_2 \left(X'X\right)^{-2} X' dW = &\text{ tr } D_2 \left(X'X\right)^{-2} X' d\left(\sum_{t=1}^{T} e_t \left(e_t' f(x) E_a \otimes e_t' f(x) E_b\right)\right) \\ &= \sum_{t=1}^{T} \text{ tr } D_2 \left(X'X\right)^{-2} X' e_t \ d \ \left(e_t' f(x) E_a \otimes e_t' f(x) E_b\right). \end{aligned}$$

Note that $D_2 \left(X'X\right)^{-2} X' e_t$ is a column vector and the elements in Kroneker product are row vectors (for continuous variables they are scalars), therefore the Kronecker expression is also a row vector. This simplifies the algebra needed to compute the gradient.

$$\begin{aligned}
&\sum_{t=1}^{T} e_t' X \left(X'X\right)^{-2} D_2' \ d \ \left(e_t' f(x) E_a \otimes e_t' f(x) E_b\right)' && (6) \\
&= \sum_{t=1}^{T} e_t' X \left(X'X\right)^{-2} D_2' \ d \ \left(E_a' A' x' e_t \otimes E_b' A' x' e_t\right) && (7) \\
&= \sum_{t=1}^{T} e_t' X \left(X'X\right)^{-2} D_2' \ d \ \text{ vec } E_b' A' x' e_t e_t' x A E_a && (8) \\
&= \sum_{t=1}^{T} e_t' X \left(X'X\right)^{-2} D_2' \text{ vec } \left(E_b' A' (dx)' e_t e_t' x A E_a + E_b' A' x' e_t e_t' (dx) A E_a\right) && (9) \\
&= \sum_{t=1}^{T} e_t' X \left(X'X\right)^{-2} D_2' \left[\left(E_a' A' x' e_t e_t' \otimes E_b' A'\right) d \text{ vec } x' + \left(E_a' A' \otimes E_b' A' x' e_t e_t'\right) d \text{ vec } x\right] && (10) \\
&= \sum_{t=1}^{T} e_t' X \left(X'X\right)^{-2} D_2' \left[\left(E_a' A' x' e_t e_t' \otimes E_b' A'\right) K_{T\times nucol} + \left(E_a' A' \otimes E_b' A' x' e_t e_t'\right)\right] d \text{ vec } x && (11)
\end{aligned}$$

In (6) we use the trace property for column vectors a, b that tr $ab' = a'b$. In (7) we apply $f(x) = xA+B$, and Kronecker property $(A \otimes B)' = (A' \otimes B')$. In (8), we use the Kronecker property for column vectors: $vec\ ab' = b \otimes a$. In (9) we take the derivative of a product $d(x'Ax) = (dx)'Ax + x'A(dx)$. In (10) apply vec $ABC = (C' \otimes A)$ vec B. Finally in (11) we use commutation matrix to get vec $x' = K$ vec x.

Therefore the gradient for the problem which includes interactions is

$$\begin{aligned} D\phi(x) = &- 2(\text{vec } X\,(X'X)^{-2}\,D_1'A')' \\ &- 2\sum_{t=1}^{T} e_t'X\,(X'X)^{-2}\,D_2'\left[(E_a'A'x'e_te_t' \otimes E_b'A')\,K_{T\times nucol} + (E_a'A' \otimes E_b'A'x'e_te_t')\right]. \end{aligned}$$

The Hessian of the problem has been computed numerically.

References

Addelman, Sidney (1962), "Symmetrical and Asymmetrical Fractional Factorial Plans", *Technometrics*, 4 (1), 47–58.

Atwood, Corwin L. (1973), "Sequences Converging to D-Optimal Designs of Experiments", *The Annals of Statistics*, 1 (2), 342–352.

——— (1976), "Convergent Design Sequences, for Sufficiently Regular Optimality Criteria", *The Annals of Statistics*, 4 (6), 1124–1138.

Bassett, Gilbert, Jr. and Roger Koenker (1978), "Asymptotic Theory of Least Absolute Error Regression", *Journal of the American Statistical Association*, 73 (363), 618–622.

Boyd, Stephen and Lieven Vandenberghe (2004), *Convex Optimization*, Cambridge University Press.

Bradlow, Eric T. (2005), "Current Issues and a "Wish List" for Conjoint Analysis", *Applied Stochastic Models in Business and Industry*, 21 (4-5), 319–323.

Byrd, Richard H., Mary E. Hribar, and Jorge Nocedal (1999), "An Interior Point Algorithm for Large-Scale Nonlinear Programming", *SIAM Journal on Optimization*, 9 (4), 877–900.

Cattin, Philippe and Dick R. Wittink (1982), "Commercial Use of Conjoint Analysis: A Survey", *Journal of Marketing*, 46 (3), 44–53.

Cochran, William G. and Gertrude M. Cox (1957), *Experimental Designs*, 2nd ed., New York: John Wiley & Sons.

Cook, R. Dennis and Christopher J. Nachtsheim (1980), "A Comparison of Algorithms for Constructing Exact D-Optimal Designs", *Technometrics*, 22 (3), 315–324.

Cox, David R. (1958), *Planning of Experiments*, New York: John Wiley & Sons.

Currim, Imran S., Charles B. Weinberg, and Dick R. Wittink (1981), "Design of Subscription Programs for a Performing Arts Series", *Journal of Consumer Research*, 8 (1), 67–75.

Dykstra, Otto, Jr. (1971), "The Augmentation of Experimental Data to Maximize $|X'X|$ ", *Technometrics*, 13 (3), 682–688.

Fedorov, Valerii V. (1972), *Theory of Optimal Experiments*, New York: Academic Press.

Green, Paul E. (1974), "On the Design of Choice Experiments Involving Multifactor Alternatives", *Journal of Consumer Research*, 1 (2), 61–68.

Green, Paul E. and Vithala R. Rao (1971), "Conjoint Measurement for Quantifying Judgmental Data", *Journal of Marketing Research*, 8 (3), 355–363.

Gustafsson, Anders, Andreas Herrmann, and Frank Huber (2007), *Conjoint Measurement: Methods and Applications*, Berlin: Springer Verlag.

Hausman, Jerry A. (1982), "The Effects of Time in Economic Experiments", in *Advances in Econometrics*, W. Hildenbrand, ed., Cambridge University Press.

Johnson, Mark E. and Christopher J. Nachtsheim (1983), "Some Guidelines for Constructing Exact D-Optimal Designs on Convex Design Spaces", *Technometrics*, 25 (3), 271–277.

Kiefer, Jack (1959), "Optimum Experimental Designs", *Journal of the Royal Statistical Society. Series B (Methodological)*, 21 (2), 272–319.

Kiefer, Jack and Jacob Wolfowitz (1960), "The Equivalence of Two Extremum Problems", *Canadian Journal of Mathematics*, 12, 363–366.

Kuhfeld, Warren F. (2010), "Experimental Design: Efficiency, Coding, and Choice Designs", Tech. Rep. MR-2010C, SAS.

Kuhfeld, Warren F., Randall D. Tobias, and Mark Garratt (1994), "Efficient Experimental Design with Marketing Research Applications", *Journal of Marketing Research*, 31 (4), 545–557.

Lawler, E. L. and D. E. Wood (1966), "Branch-and-Bound Methods: A Survey", *Operations Research*, 14 (4), 699–719.

Leyffer, Sven (2001), "Integrating SQP and Branch-and-Bound for Mixed Integer Nonlinear Programming", *Computational Optimization and Applications*, 18, 295–309.

López Fidalgo, Jesús (2009), "A Critical Overview on Optimal Experimental Designs", *Boletín de Estadística e Investigación Operativa*, 25 (1), 14–21.

Magnus, Jan R. and Heinz Neudecker (1999), *Matrix Differential Calculus with Applications in Statistics and Econometrics*, Wiley.

Meyer, Ruth K. and Christopher J. Nachtsheim (1995), "The Coordinate-Exchange Algorithm for Constructing Exact Optimal Experimental Designs", *Technometrics*, 37 (1), 60–69.

Mitchell, Toby J. (1974), "An algorithm for the construction of "D-optimal" experimental designs", *Technometrics*, 16 (2), 203–210.

Mitchell, Toby J. and FL Miller Jr (1970), "Use of design repair to construct designs for special linear models", *Math. Div. Ann. Progr. Rept.(ORNL-4661)*, 130–131.

Netzer, Oded, Olivier Toubia, Eric T. Bradlow, Ely Dahan, Theodoros Evgeniou, Fred M. Feinberg, Eleanor M. Feit, Sam K. Hui, Joseph Johnson, John C. Liechty, James B. Orlin, and Vithala R. Rao (2008), "Beyond Conjoint Analysis: Advances in Preference Measurement", *Marketing Letters*, 19 (3), 337–354.

Nocedal, Jorge and Stephen Wright (2006), *Numerical Optimization*, New York: Springer.

Pukelsheim, Friedrich and Sabine Rieder (1992), "Efficient Rounding of Approximate Designs", *Biometrika*, 79 (4), 763–770.

St. John, R. C. and N. R. Draper (1975), "D-Optimality for Regression Designs: A Review", *Technometrics*, 17 (1), 15–23.

Stevens, Stanley S. (1951), *Handbook of Experimental Psychology*, John Wiley & Sons Inc.

Toubia, Olivier and John R. Hauser (2007), "Research Note – On Managerially Efficient Experimental Designs", *Marketing Science*, 26 (6), 851–858.

Vandenberghe, Lieven, Stephen Boyd, and Shao-Po Wu (1998), "Determinant Maximization with Linear Matrix Inequality Constraints", *SIAM Journal on Matrix Analysis and Applications*, 19 (2), 499–533.

Wald, Abraham (1943), "On the Efficient Design of Statistical Investigations", *The Annals of Mathematical Statistics*, 14 (2), 134–140.

Whittle, Peter (1973), "Some General Points in the Theory of Optimal Experimental Design", *Journal of the Royal Statistical Society. Series B (Methodological)*, 35 (1), 123–130.

Wittink, Dick R. and Philippe Cattin (1989), "Commercial Use of Conjoint Analysis: An Update", *Journal of Marketing*, 53 (3), 91–96.

Wittink, Dick R., Lakshman Krishnamurthi, and Julia B. Nutter (1982), "Comparing Derived Importance Weights Across Attributes", *Journal of Consumer Research*, 8 (4), 471–474.

Wynn, Henry P. (1970), "The Sequential Generation of D-Optimum Experimental Designs", *The Annals of Mathematical Statistics*, 41 (5), 1655–1664.

——— (1972), "Results in the theory and construction of D-optimum experimental designs", *Journal of the Royal Statistical Society. Series B (Methodological)*, 34 (2), 133–147.