Economic Valuation of Environmental Change

Module 5.3: Design of experiments

Book chapters: PR Ch. 19, CBB CH. 5

LaTex commands

In [23]:

```
# Import packages we'll need for this module
#################################################
from numpy import *  # numpy is used a lot in Python, and some load it "as np" - but then
# every time we use a numpy command, so I prefer not to use a prefix in this case
from numpy.linalg import inv  #not really necessary since we already imported the entire n
# but makes life easier taking inverses, else we would have to type "linalg.inv" all the t
from numpy.linalg import det #same for determinant
import matplotlib.pyplot as plt
from scipy.stats import invgamma #for draws from inverse gamma
from scipy.stats import norm #for evaluating normal priors for betas
from scipy.optimize import minimize #needed for MLE routine within the GS
from sklearn.neighbors import KernelDensity as KD #for smoth plotting of the (empirical) d
import pandas as pd #for creating data frames and output tables
import math #for pi
from scipy.special import gammaln #for evaluating the multivariate t-density - same s Mat
```

Introduction

The efficient design of experiments (in economic valuation and other areas) is a vast and complex field that has spawned its own strand of the literature in recent decades. The following just gives a very basic introduction with focus on empirical implementation. For more in-depth reading I recommend the following sources, in addition to the book chapters listed above:

- Henscher, D., Rose, J., Greene, W., 2015. Applied choice analysis: A primer. Cambridge University Press. 2nd edition.
- Johnson, F., Kanninen, B., Bingham, M., Ozdemir, S., 2007. Experimental design for stated choice studies, in: Kanninen, B. (Ed.), Valuing environmental amenities using stated choice studies. Springer, pp. 159 - 202.
- Rose, J., Bliemer, C., 2009. Constructing effcient stated choice experimental designs. Transport Reviews 29, 587-617.

Choice of attributes and levels

Typical CE's include 3-6 attributes (in additon to price or "bid"), with each attribute characterizes by 2-4 levels. For example, in the red tide survey we used the following attributes and levels:

- width of coverage (6 miles, 12 miles)
- accuracy for the first 12 hours of the forecast (50%, 75%, 100%)
- accuracy for the second 12 hours of the forecast (50%, 75%, 100%)
- bid ($ 5, $ 15, $ 25, $ 35)

Attributes should be simple to explain and measure, and ideally chosen such that any mix of attribute levels is reasonably realistic. The latter condition hardly ever holds for all combinations of levels, but should at least hold for many combinations.

For example, in the red tide case, it would be difficulty to present a choice profile with higher accuracy later in the forecast than earlier. In other applications attribute levels my reflect zero-sum shares of an area, such as "percent of wetland open to trail-based recreation," vs. "percent open to bird-viewing recreation with platforms, but no trails" vs. "percent closed to public." These naturally can't exceed 100 when added up, and often need to add to 100 to make sense.

Similarly, it would be a stretch to present to a respondent a choice profile (= option) that stipulates a high chance of fish survival or population growth along with very poor water or habitat conditions.

Levels should ideally span the natural or planned / envisioned range of a given attribute, with perhaps one or two interpolation points. Bid levels should be informed by similar "fees" in existing markets, if possible, and / or by focus group discussions. Note that in contrast to basic CV, the lowest bid may not necessarily attract all votes - it all depends on the remaining attribute mix. Analogous reasoning holds for the highest bid. Remaining bids (typically 2-4) can then be distributed uniformly between the endpoints.

Note that the **status quo (SQ) profile**, which will always be the same in all choice sets and for all respondents, does not directly figure into design considerations, other than perhaps to provide guidance on lower or upper bounds for attribute levels.

The full factorial and first trimming

The full set of all possible combinations of attributes and levels is called the **full factorial** (FF). In other words, the full factorial comprises the set of all (theoretically) possible **choice profiles**. This should always be the starting point for any CE design. For smaller designs such as the one we used for the RT survey this can be done "by hand" in Excel. For larger designs with hundreds or even thousands of initial profiles, you can use a factorial design program that produces combinations of generic attribute levels (e.g. 1,2,3,4...), which can then be replaced with actual levels. Python has a pre-packaged module for this called "PyDOE2."

Here is an example that fits the RT design with 2 x 3 x 3 x 4 = 72 profiles:

In [ ]:

```
pip install --upgrade pyDOE2
```

In [12]:

```
import pyDOE2 as doe #for creating data frames and output tables
```

In [22]:

```
# create the full factorial
startset = doe.fullfact([2, 3, 3, 4]) #72 by 4

#optionally, sort by attributes
startset = startset[startset[:, 3].argsort()]  # sort by bid
startset = startset[startset[:, 2].argsort(kind='mergesort')]  # sort by accuracy, 2nd pe
startset = startset[startset[:, 1].argsort(kind='mergesort')]
startset = startset[startset[:, 0].argsort(kind='mergesort')]

#optionally, convert to data frame
startsetdf = pd.DataFrame(startset, columns = ['band','acc1','acc2','bid'])
display(startsetdf)
```

	band	acc1	acc2	bid
0	0.0	0.0	0.0	0.0

	band	acc1	acc2	bid
1	0.0	0.0	0.0	1.0
2	0.0	0.0	0.0	2.0
3	0.0	0.0	0.0	3.0
4	0.0	0.0	1.0	0.0
...	...	...	...	...
67	1.0	2.0	1.0	3.0
68	1.0	2.0	2.0	0.0
69	1.0	2.0	2.0	1.0
70	1.0	2.0	2.0	2.0
71	1.0	2.0	2.0	3.0

72 rows × 4 columns

You can now replace the generic numbers with actual attribute settings.

Here is the FF of 2 x 3 x 3 x 4 = 72 profiles for the red tide survey:

- RT starting set of profiles

At this stage it is usually both possible and necessary to **eliminate nonsensical profiles** that either violate mathematical principles (e.g. must add to 100%) or common sense. As mentioned above, in the RT case we need to drop profiles with higher forecast accuracy in the second 12-hour period compared to the first 12-hour period.

Failure to eliminate nonsensical profiles will inevitably trigger protest responses and undesirable respondent behavior during data collection (e.g. stop paying attention to any choice profile and voting for the SQ on all occasions).

For the RT case, this reduced the set of **permissible profiles** to 48:

- RT permissible set of profiles

Determine the minimum and maximum set of profiles

The minimum set

The **minimum set of profiles** is the number of profiles needed to **econometrically identify all desired effects**, as stipulated by your theoretical / structural model. As mentioned in previous modules, we should aim to estimate as many "nonlinear" (= level-specific) effects as possible, plus their interactions.

Let L_1 through L_A be the number of levels for A attributes. Then the number of degrees of freedom (= min. number of profiles) required can be computed as:

$$\begin{aligned} \text{main effects: } & \sum_{i=1}^{A}\left(L_i-1\right)\\ \text{2-way interactions: } & \sum_{i=1}^{A}\sum_{j\neq i}\left(L_i-1\right)\left(L_j-1\right)\\ \text{3-way interactions: } & \sum_{i=1}^{A}\sum_{j\neq i}\sum_{k\neq i,j}\left(L_i-1\right)\left(L_j-1\right)\left(L_k-1\right)\\ & \text{etc.}\end{aligned} \tag{1}$$

plus 1 for the opt-out (SQ) option, and another DoF for the model error.

In most applications, identification of as many 2-way interactions as possible or that make sense in practice will be sufficient.

In a random utility context we are (typically) holding the marginal utility of income constant for all respondents, so there are no meaningful interactions between attributes and the bid vector. In fact, we are only estimating a single main effect for "bid " (i.e. we treat bid as a linear variable).

Thus, for the RT project, we have, in principle:

- $1+2+2+1=6$ main effects (bid = linear)
- $1*2$ interactions of "band" with "acc1"
- $1*2$ interactions of "band" with "acc2"
- $2*2$ interactions of "acc1" with "acc2"

This would imply 12 degrees of freedom to identify coefficients, plus 2 more (SQ, error term) for a total of 14. For actual implementation, we decided a single "linear" interaction between "acc1" and "acc2" would be sufficient. This reduced the total number of DoF's to 11.

The analyst now needs to verify that the total number of permissibe choice sets equals or exceeds the number of DoFs required to estimate the model. This is no problem in our case, with 48 permissible policy profiles and only 11 DoF's required to estimate the model.

The maximum set

But can we implement all 48? This brings us to the question of the **maximum set of profiles** that can be engaged in the design. This number is related to the targeted sample size, specifically the targeted number of respondents N, the envisioned number of choice sets per respondent J, and the desired minimum number of observations per profile, n_j.

This also has bearings on the number of **blocks** B of choice sets within the survey at large. Each block comprises the same J choice sets, but they differ across blocks. Each block is given to a sub-sample of respondents.

For the RT project, our budget allowed for the collection of 500 completed surveys, thus $N=500$. We also wanted to limit the number of choice sets per respondent to $J=4$. This yields a total of 2000 choice sets that can be accommodated by the survey. For the full set of 48 permissible profiles, this further implies 12 blocks (48/4), each with a sub-sample size of (approximately) 41 observations.

While this would generally be satisfactory, in our case we opted for fewer profiles and more observations per profile since we wanted a sufficient number of observations for the **first profile shown to each respondent** to test for and potentially circumvent sequencing effects.

Accommodating all 48 profiles would imply only about 10 observations per profile when we ignore all but the first choice set. Ultimately, we settled for a total of 20 unique profiles, 5 blocks @ 4 choice sets, and thus 100 observations per block. This also implies that each unique profile is seen first by 25 respondents, thus permitting the estimation of a "first-choice" only model down the line if desired.

In a nutshell, all is well if the minimum number of profiles to identify the model falls well below the maximum number that can be administerd at reasonable sample sizes, as in the current case. If the two numbers get close, or the reverse holds, the permissible set of profiles needs to be reduced to what is referred to as a **fractional factorial** design. The literature referenced above gives some techniques and examples of how this can be accomplished, while keeping the design as efficient as possible (= able to identify as many main effects and interactions as possible).

Grouping profiles into choice sets

The next task is to efficiently choose the final set of profiles (here: 20 out of 48), and group them into pairs to form actual choice sets (here we need 20 unique choice sets - the same number as the final number of unique profiles).

This is usually done with specialized software, and following some **efficieny criterion**. The general procedure is as follows:

1) Randomly pick the desired number of unique profiles (here 20) from the total permissible set (here 48) and form an equal number of pairs (here: 20) 2) Apply some efficiency criterion involving the resulting data matrix and / or optimization constructs (such as the variance-covariance matrix of model parameters) and compute the efficiency score. 3) Change the pairing of profiles in some random and / or systematic fashion (possibly bringing in different profiles from the permissible set) 4) Re-compute the efficiency score 5) Repeat 2)-4) until some convergence condition is met, or for a pre-specified number of iterations

A popular efficiency score is the **"D-efficiency"** which minimizes the determinant of the inverse of the information matrix corresponding to a simple conditional logit model, setting all coefficients to zero, and scaling by the number of parameters in the model. This is especially attractive if the ultimate model for data analysis is envisioned to be the conditional logit, but works quite well for other, more complex RUMs.

Popular software packages to form choice sets based on statistical efficiency criteria include SAS and Ngene. I found STATA's "*dcreate*" module quite suitable for this task, as long as D-efficiency is a meaningful optimization criterion (which it is for most applications).

Note that the analyst will usually impose certain **constraints** on the optimization algorithm to avoid undesirable pairings from a "common sense" or "theoretical" perspective. Typically, this includes checks for **dominance**, e.g. where one option is more desirable based on at least one attribute, equally desirable based on all others, AND offered at a lower bid. However, such constrained optimization may have problems converging, such that the analyst may prefer a less constrained or unconstrained optimality search, and impose constraints ex-post "manually," as described below.

Grouping choice sets into blocks

Once the optimal set of choice sets has been determined, the sets need to be grouped into blocks. This is usually also accomplished using statistical software.

For example, STATA's *dcreate* package repeatedly mixes choice sets into blocks until the block "treatment" is as close as possible to orthogonal to the attributes as possible.

Here is the final result produced by *decreate* for the RT survey:

- RT dcreate output

Final adjustments

In the RT case, we managed to force *dcreate* to avoid dominance beyond 4 problematic cases (out of 20). Anything more stringent led to convergence problems.

We thus had to "fix" these cases manually ex-post. For example, in line 10 of the dcreate output, option 1 shows higher attribute settings and a lower price compared to option 2. We simply changed prices for those cases by hand to preserve maximum balance on price (= all price levels appear with about equal frequency in the design matrix).

Here is the final design version that was ultimately used in the survey:

- RT Final Design

As mentioned before, we then instructed the survey agency to make sure each choice set within a given block was seen first by a subset of (approx.) 25 respondents. From a survey implementation perspective, this implied $5x4 = 20$ survey versions:

- RT Block Rotations

Fundamental design considerations

The specialized literature often lists four desirable features of an experimental design:

1) level balance (each attribute level appears about equally often) 2) orthogonality (attribute vectors have minimal correlation) 3) minimal overlap (avoid same attribute levels for both options) 4) utility balance (about same utility for each option)

Level balance is usually achieved in following the design steps above - but can be enhanced in the "final adjsutment phase" (as we did with bids in the RT case). **Orthogonality** used to be the primary design focus before statistical efficiency became a (somewhat) competing criterion. Naturally, orthogonality across attribute variables enhance econometric efficiency, but often comes at the cost of conflicting with the other three desirable factors and / or identification goals, and may not be possible if a fractional factorial design has to be used. Similarly, a certain degree of **overlap**, while not ideal from an orthogonality perspective, facilitates choices for respondents (if attribute levels are the same across options, one can focus solely on the remaining attributes) and may reduce "choice fatigue." **Utility-balance**, in turn, is essentially the opposite of dominance - making choices tougher and forcing respondents to focus and pay attention. This is the underlying concept driving efficiency designs based on modeling / statistical considerations.

In a nutshell, while computational algorithms are available to facilitate the design task, the analyst's input and judgment are needed at virtually every step of the process. In my opinion avoiding nonsensical profiles and dominant option pairs are the single most important considerations for successful implementation.