D-Optimal Experimental Design

Adam Rahman

September 4, 2017

Consider the problem of estimating a vector $\mathbf{x}$ from measurements $\mathbf{y}$ given by the relationship

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0,1).$$

The variance-covariance matrix of such an estimator is proportional to $(\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}$. A reasonable goal during the design phase of an experiment would therefore be to minimize $(\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}$ in some way.

There are many different ways in which $(\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}$ might be made minimal. For example, minimization of the trace of $(\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}$ (A-Optimality), minimization of the maximum eigenvalue of $(\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}$ (E-Optimality), minimization of the determinant of $(\mathbf{A}^\mathsf{T}\mathbf{A})^{-1}$ (D-Optimilaity), and maximization of the trace of $\mathbf{A}^\mathsf{T}\mathbf{A}$ (T-Optimality) all have their merits.

Perhaps the most commonly used of these optimality criteria is D-Optimality, which is equivalent to maximizing the determinant of $\mathbf{A}^\mathsf{T}\mathbf{A}$. Typically, the rows of $\mathbf{A} = [\mathbf{a}_1, ..., \mathbf{a}_q]^T$ are chosen from M possible test vectors $\mathbf{u}_i \in \mathcal{R}^p$, $i = 1, ...M$, which are known in advance. That is,

$$\mathbf{a}_i \in \{\mathbf{u}_1, ..., \mathbf{u}_M\}, \quad i = 1, ..., q$$

Given that the matrix $\mathbf{A}$ is made up of these test vectors $\mathbf{u}_i$, the matrix $\mathbf{A}^\mathsf{T}\mathbf{A}$ can be written as

$$\mathbf{A}^\mathsf{T}\mathbf{A} = q\sum_{i=1}^{M} \lambda_i \mathbf{u}_i {\mathbf{u}_i}^\mathsf{T} \tag{1}$$

where λ_i is the fraction of rows in $\mathbf{A}$ that are equal to the vector $\mathbf{u}_i$ [2]. Then, the the D-optimal experimental design problem can be written as a minimum determinant problem [2]

$$\begin{array}{ll} \underset{\boldsymbol{\lambda}}{\text{minimize}} & \log\det\,(\sum_{i=1}^{M} \lambda_i \mathbf{u}_i {\mathbf{u}_i}^\mathsf{T})^{-1} \\ \text{subject to} & \\ & \begin{array}{rcl} \lambda_i & \geq & 0, \quad i = 1, ..., m \\ \sum_{i=1}^{M} \lambda_i & = & 1 \end{array} \end{array}$$

Due to the inequality constraint, this primal formulation cannot be interpreted as a primal SQLP. By defining $\mathbf{Z} = \mathbf{u}\ diag(\boldsymbol{\lambda})\ \mathbf{u}^\mathsf{T}$, the dual problem is ([1])

$$\begin{array}{ll} \underset{\mathbf{Z},\ \mathbf{z}^l,\ \boldsymbol{\lambda}}{\text{maximize}} & \log\det\,(\mathbf{Z}) \\ \text{subject to} & \\ & \begin{array}{rcll} -\sum_{i=1}^{p} \lambda_i(\mathbf{u}_i {\mathbf{u}_i}^\mathsf{T}) + \mathbf{Z} & = & 0, & \mathbf{Z} \in \mathcal{S}^n \\ -\boldsymbol{\lambda} + \mathbf{z}^l & = & 0, & \mathbf{z}^l \in \mathcal{R}^p_+ \\ \mathbf{1}^T\boldsymbol{\lambda} & = & 1, & \boldsymbol{\lambda} \in \mathcal{R}^p \end{array} \end{array}$$

We proceed with writing the D-Optimal design problem as an SQLP by first considering the objective function. The objective function depends only on the determinant of the matrix variable $\mathbf{Z}$, which is the log-barrier. This indicates that the variable v^s in the dual equation is equal to 1 in this formulation, while v^q and v^l are both zero. Since $\boldsymbol{\lambda}$ does not appear in the objective function, the vector $\mathbf{b}$ is equal to $\mathbf{0}$.

The constraint matrices $\mathbf{A}$ are easy to define in the case of the dual formulation, as they multiply multiply $\boldsymbol{\lambda}$. In the first constraint, each λ_i is multiplied by the matrix formed by $-\mathbf{u}_i\mathbf{u}_i^\top$, so define $\mathbf{A}_i$ to be

$$\mathbf{A}_i = -\mathbf{u}_i\mathbf{u}_i^\top, \quad i = 1, ..., p.$$

Then, the constraint matrix is $\mathbf{A}^s = [svec(\mathbf{A}_1), ..., svec(\mathbf{A}_p)]$. In the second constraint containing the linear variable $\mathbf{z}^l$, the constraint matrix is $\mathbf{A}^l = -\mathbf{I}_p$, and in the third constraint containing only the unconstrained variable $\boldsymbol{\lambda}$, the constraint matrix is $\mathbf{A}^u = \mathbf{1}^\top$. Since there is no quadratic variable, $A^q = \mathbf{0}$.

Finally, define the right hand side of each constraint

$$\begin{aligned} \mathbf{C}^s &= \mathbf{0}_{n\times n} \\ \mathbf{C}^l &= \mathbf{0}_{p\times 1} \\ \mathbf{C}^u &= 1 \end{aligned}$$

which fully specifies the D-Optimal design problem as an SQLP.

To convert this to a form usable by **sdpt3r**, we initialize our input variables by noting we have three blocks - $\mathbf{X}$, $\mathbf{z}^l$, and $\boldsymbol{\lambda}$

```
R> blk <- matrix(list(),nrow=3,ncol=2)
R> At <- matrix(list(),nrow=3,ncol=1)
R> C <- matrix(list(),nrow=3,ncol=1)
```

As before, we declare the three blocks in `blk`. The first block is semidefinite containing the matrix $\mathbf{Z}$, the second a linear block containing $\mathbf{z}^l$, and the third an unrestricted block containing $\boldsymbol{\lambda}$

```
R> blk[[1,1]] <- "s"
R> blk[[1,2]] <- n

R> blk[[2,1]] <- "l"
R> blk[[2,2]] <- p

R> blk[[3,1]] <- "u"
R> blk[[3,2]] <- 1
```

Next, by noting the variable $\boldsymbol{\lambda}$ does not appear in the objective function, we specify `b` as a vector of zeros

```
R> b <- matrix(0,nrow=p,ncol=1)
```

Next, looking at the right-hand side of the constraints, we define the matrices `C`

```
R> C[[1,1]] <- matrix(0,nrow=n,ncol=n)
R> C[[2,1]] <- matrix(0,nrow=p,ncol=1)
R> C[[3,1]] <- 1
```

Finally, we construct `At` for each variable

```
R> A <- matrix(list(),nrow=p,ncol=1)
$>
R> for(k in 1:p){
R>   A[[k]] <- -uk %*% t(uk)
R> }
```

```
R>
R> At[[1,1]] <- svec(blk[1,], A)
R> At[[2,1]] <- diag(-1,nrow=p,ncol=p)
R> At[[3,1]] <- matrix(1,nrow=1,ncol=p)
```

The final hurdle necessary to address in this problem is the existence of the log-barrier. Recall that it is assumed that v^s, v^q, and v^l in the dual problem are all zero in OPTIONS. In this case, we can see that is not true, as we have a log term containing $\mathbf{Z}$ in the objective function, meaning v^s is equal to one. To pass this to sqlp, we define the OPTIONS$parbarrier variable as

```
R> OPTIONS$parbarrier <- matrix(list(),nrow=3,ncol=1)
R> OPTIONS$parbarrier[[1]] <- 1       #for vs
R> OPTIONS$parbarrier[[2]] <- 0       #for vq
R> OPTIONS$parbarrier[[3]] <- 0       #for vl
```

The D-Optimal experimental design problem can now be solved using sqlp

```
R> sqlp(blk, At, C, b, OPTIONS)
```

To demonstrate the output generated from a D-optimal experimental design problem, we consider a simple 3×25 matrix containing the known test vectors $\mathbf{u}_1, ..., \mathbf{u}_{25}$ (the data is available in the sqlp package). To generate the required input for sqlp, we use the function doptimal, which takes as input an $n \times p$ matrix $\mathbf{U}$ containing the known test vectors, and returns the input necessary for sqlp. The output we are interested in is y, corresponding to $\boldsymbol{\lambda}$ in our formulation, the percentage of each $\mathbf{u}_i$ necessary to achieve maximum information in the experiment.

```
R> data(DoptDesign)
R> DoptDesign

    [,1]  [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8]  [,9] [,10] [,11] [,12] [,13] [,14] [,15] [,16]
V1 0.531 0.769 0.646 0.865 0.369 0.869 0.171 0.788 0.174 0.022 0.883 0.357 0.926 0.260 0.183 0.264
V2 0.232 0.661 0.632 0.095 0.314 0.674 0.911 0.274 0.454 0.346 0.695 0.685 0.476 0.398 0.564 0.994
V3 0.510 0.815 0.754 0.386 0.939 0.621 0.146 0.237 0.772 0.749 0.543 0.101 0.025 0.589 0.014 0.868

   [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
V1 0.122 0.629 0.058 0.791 0.433 0.169 0.637 0.323 0.146
V2 0.310 0.482 0.663 0.218 0.018 0.097 0.254 0.407 0.153
V3 0.410 0.469 0.888 0.745 0.441 0.718 0.926 0.791 0.117

R> out <- doptimal(DoptDesign)
R> blk <- out$blk
R> At <- out$At
R> C <- out$C
R> b <- out$b
R> OPTIONS <- out$OPTIONS

R> out <- sqlp(blk,At,C,b,OPTIONS)
R> out$y

       [,1]
 [1,] 0.000
 [2,] 0.000
```

```
 [3,] 0.000
 [4,] 0.000
 [5,] 0.000
 [6,] 0.000
 [7,] 0.154
 [8,] 0.000
 [9,] 0.000
[10,] 0.000
[11,] 0.000
[12,] 0.000
[13,] 0.319
[14,] 0.000
[15,] 0.000
[16,] 0.240
[17,] 0.000
[18,] 0.000
[19,] 0.000
[20,] 0.000
[21,] 0.000
[22,] 0.000
[23,] 0.287
[24,] 0.000
[25,] 0.000
```

The information matrix $\mathbf{A}^\top\mathbf{A}$ is a linear combination of the test vectors $\mathbf{u}_i$, weighted by the optimal vector $\mathtt{y}$ above.

References

[1] Kim-Chuan Toh, Michael J Todd, and Reha H Tütüncü. Sdpt3 - a matlab software package for semidefinite programming, version 1.3. *Optimization methods and software*, 11(1-4):545–581, 1999.

[2] Lieven Vandenberghe, Stephen Boyd, and Shao-Po Wu. Determinant maximization with linear matrix inequality constraints. *SIAM journal on matrix analysis and applications*, 19(2):499–533, 1998.