STOR 767 Spring 2019 Hw1: Computational Part

Due on 01/23/2019 in Class

YOUR NAME

Instruction.

- Homework 1 includes **Theoretical Part** (50%) and **Computational Part** (50%).
- For homework submission and grading, edit this document and create a PDF file to print and submit in class. Codes and key results should be displayed.

Exercise 1. *(5 pt)* **Hadamard matrix** is a useful construction for two-level orthogonal design. It's defined recursively by

$$\mathbf{H}_1 = \begin{pmatrix}1\end{pmatrix} \in \mathbb{R}^{1\times 1}, \quad \mathbf{H}_{2^k} = \begin{pmatrix}\mathbf{H}_{2^{k-1}} & \mathbf{H}_{2^{k-1}} \\ \mathbf{H}_{2^{k-1}} & -\mathbf{H}_{2^{k-1}}\end{pmatrix} \in \mathbb{R}^{2^k\times 2^k}. \quad (k \in \mathbb{N})$$

Create $\mathbf{H}_{2^4}$ in **R**.

```
YOUR CODE HERE
```

Exercise 2. *(5 pt)* It has been shown that a LASSO estimate for the location of X is of the following thresholding form

$$\hat{\mu}_{\text{LASSO}} = \underset{\mu\in\mathbb{R}}{\operatorname{argmin}} \frac{1}{2}(X-\mu)^2 + \lambda|\mu| = \begin{cases} X+\lambda, & X \leqslant -\lambda \\ 0, & -\lambda < X \leqslant \lambda \\ X-\lambda, & X > \lambda \end{cases}$$

Now let $\lambda = 1$ and consider 100 *i.i.d.* sample $\{X_i\}_{i=1}^n$ drawn from $\mathcal{N}(0,1)$. Return the vector of the LASSO estimates for their individual locations in **R**.

```
x <- rnorm(100)
YOUR CODE HERE
```

Exercise 3. *(5 pt)* Table 1 presents a mixed 2-level and 3-level orthogonal design from (Wu and Hamada 2011). The first four rows in 2-level factors A, B and C, as a 2^{3-1} design, have been repeated for the next eight 4-row groups. Groups are embedded into a 3^{3-1} design in 3-level factors D, E and F. In particular, column C = column A × column B, column F = column D + column E (mod 3) by encoding $(-1, 0, 1)$ in $(1, 2, 0)$. Create such design matrix in **R** without reading from Table 1 directly.

Table 1 $2^{3-1} \times 3^{3-1}$ Orthogonal Array

	2-Level Factors			3-Level Factors		
Run	**A**	**B**	**C**	**D**	**E**	**F**
1	-1	-1	1	-1	-1	-1
2	1	-1	-1	-1	-1	-1
3	-1	1	-1	-1	-1	-1
4	1	1	1	-1	-1	-1
5	-1	-1	1	0	-1	0
6	1	-1	-1	0	-1	0
7	-1	1	-1	0	-1	0
8	1	1	1	0	-1	0
9	-1	-1	1	1	-1	1
10	1	-1	-1	1	-1	1
11	-1	1	-1	1	-1	1
12	1	1	1	1	-1	1
13	-1	-1	1	-1	0	0
14	1	-1	-1	-1	0	0
15	-1	1	-1	-1	0	0
16	1	1	1	-1	0	0
17	-1	-1	1	0	0	1
18	1	-1	-1	0	0	1
19	-1	1	-1	0	0	1
20	1	1	1	0	0	1
21	-1	-1	1	1	0	-1
22	1	-1	-1	1	0	-1
23	-1	1	-1	1	0	-1
24	1	1	1	1	0	-1
25	-1	-1	1	-1	1	1
26	1	-1	-1	-1	1	1
27	-1	1	-1	-1	1	1
28	1	1	1	-1	1	1
29	-1	-1	1	0	1	-1
30	1	-1	-1	0	1	-1
31	-1	1	-1	0	1	-1
32	1	1	1	0	1	-1
33	-1	-1	1	1	1	0
34	1	-1	-1	1	1	0
35	-1	1	-1	1	1	0
36	1	1	1	1	1	0

```
YOUR CODE HERE
```

Exercise 4. *(5 pt)* Thickness data from a paint experiment based on Table 1 design in (Wu and Hamada 2011) are collected as below. Compute the sum of squares for all factors (main effects) from scratch, *i.e.* without resorting to any ANOVA-type **R** functions. Compare them with outputs produced by `aov`.

```
y <- scan(text = "0.755 0.550 0.550 0.600 0.900 0.875 1.000 1.000 1.400 1.225 1.225 1.475
0.600 0.600 0.625 0.500 0.925 1.025 0.875 0.850 1.200 1.250 1.150 1.150 0.500 0.550 0.575
0.600 0.900 1.025 0.850 0.975 1.100 1.200 1.150 1.300")

YOUR CODE HERE
```

Exercise 5. *(10 pt)* Write a function `optim_gd(par, fn, gr, gr_lips, maxit = 10000, tol = 1e-5)` to find the minimizer of a smooth convex function using gradient descent.[1]

- `par`: initial values for the parameters to be optimized over.
- `fn`: objective function to be minimized f on domain $\mathcal{X}$.
- `gr`: gradient of objective function ∇f.
- `gr_lips`: Lipschitz gradient constant L_f, *i.e.*

$$\|\nabla f(\boldsymbol{x}) - \nabla f(\boldsymbol{y})\|_2 \leqslant L_f \|\boldsymbol{x} - \boldsymbol{y}\|_2. \quad (\forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X})$$

- `maxit`: maximal number of iterations.
- `tol`: convergence tolerance parameter $\epsilon > 0$.

Iterations are performed by

$$\boldsymbol{x}^{k+1} := \boldsymbol{x}^k - \frac{1}{L_f} \nabla f(\boldsymbol{x}^k)$$

with stopping criterion

$$\frac{\|\nabla f(\boldsymbol{x}^k)\|_2}{\max\{1, \|\nabla f(\boldsymbol{x}^0)\|_2\}} \leqslant \epsilon.$$

Return a list with `par =` minimizer, `value =` optimal objective value, and `counts =` number of iterations performed. Apply it to the bivariate function[2]

$$f(x_1, x_2) = \log\left(1 + e^{-x_1 + x_2}\right) + \log\left(1 + e^{x_1}\right) + \log\left(1 + e^{-x_1 - x_2}\right)$$

with $L_f = \frac{5}{4}$ and initial value $(0, 0)$. Compare it with the built-in optimization function `optim` with `method = "BFGS"`.

```
YOUR CODE HERE
```

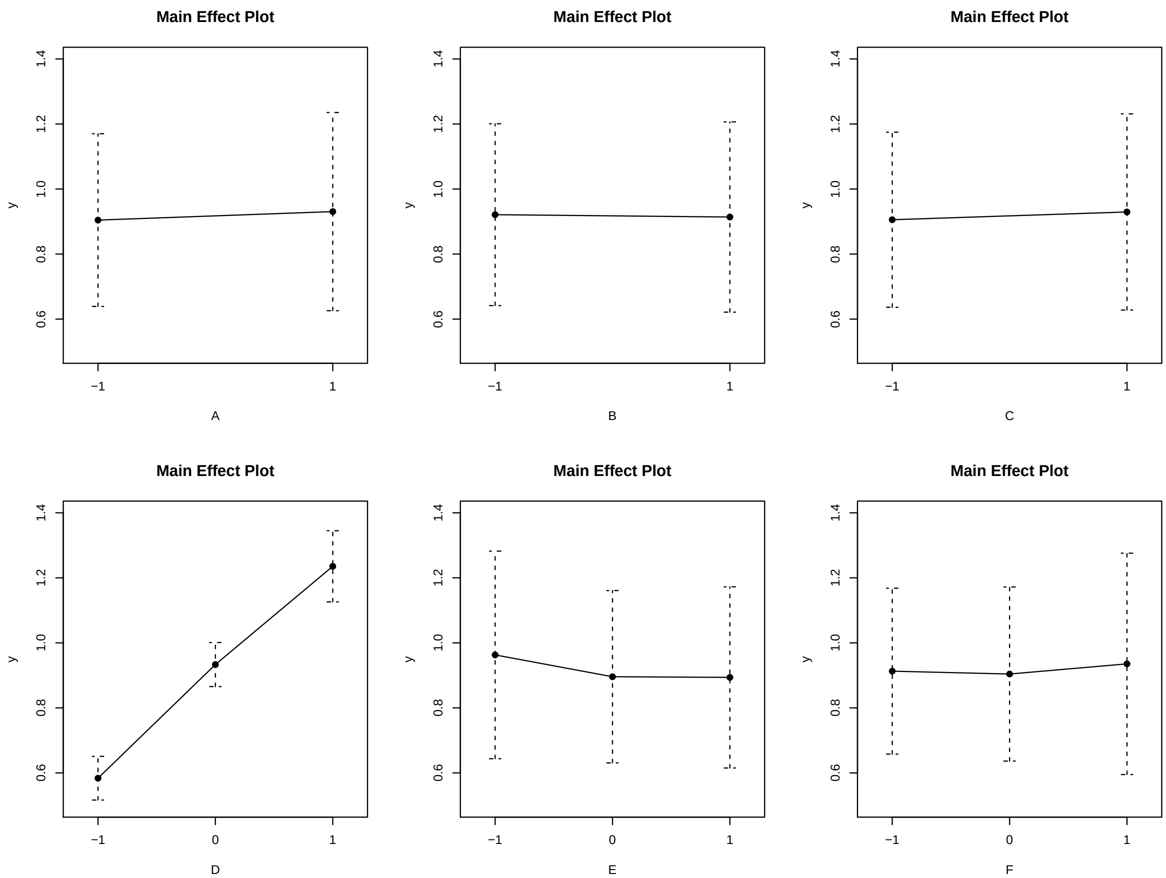
Exercise 6. *(10 pt)* Create the ANOVA table based on **Exercise 4** from scratch. Sum of squares, degrees of freedom, F-values, p-values and indicators of significance are to be reported. Comment on the relationship of all sum of squares and explain why.

```
YOUR CODE HERE
```

Exercise 7. *(10 pt)* Reproduce the code that generates the following plot based on **Exercise 4**.

[1] STOR 893 Fall 2018 lecture note http://quoctd.web.unc.edu/files/2018/10/lecture4-selected-cvx-methods.pdf.

[2] Negative log-likelihood of Logistic regression of the data $\{(-1, 1), (0, 0), (1, 1)\}$. Try performing `glm` to see whether the outputs coincide.




```
YOUR CODE HERE
```

Biblography

Wu, C.F. Jeff, and Michael S. Hamada. 2011. *Experiments: Planning, Analysis, and Optimization*. Vol. 552. John Wiley & Sons.