APTS Design of Studies - Preliminary Material

September 2019

Dave Woods (D.Woods@southampton.ac.uk ; http://www.southampton.ac.uk/~davew)
Statistical Sciences Research Institute, University of Southampton

This APTS module concerns how to influence the data generating process (mostly via carefully designed experiments) in order to "improve" the quality of the inferences and conclusions that can be drawn. To that end, it makes sense to review some simple ways we can assess the quality of a fitted statistical model.

Further good preparation for this module would be to review the material from the APTS courses on Statistical Computing and Statistical Modelling (available from https://warwick.ac.uk/fac/sci/statistics/apts/students/resources/).

Throughout this preliminary material, we will assume we observe independent data $\boldsymbol{y} = (y_1, \ldots, y_n)^{\mathrm{T}}$, where the ith observation follows a distribution with density/mass function $\pi(y_i; \boldsymbol{\theta}, \boldsymbol{x}_i)$, where $\boldsymbol{\theta}$ is a q-vector of unknown parameters (requiring estimation) and $\boldsymbol{x}_i = (x_{1i}, \ldots, x_{ki})^{\mathrm{T}}$ is a k-vector of values of controllable variables.

For example, in a simple chemistry experiment, y_i might be the measurement of product yield from the ith reaction, with x_{1i} being the setting of temperature for this reaction and x_{2i} being the setting of pressure.

Maximum likelihood, Fisher information and asymptotic normality

The likelihood is simply the joint distribution of the data $\boldsymbol{y}$ considered as a function of the parameters $\boldsymbol{\theta}$. For independent data:

$$L(\boldsymbol{\theta}) = \prod_{i=1}^{n} \pi(y_i; \boldsymbol{\theta}, \boldsymbol{x}_i) .$$

The maximum likelihood estimator (MLE), $\hat{\boldsymbol{\theta}}$, maximises this quantity. Equivalently, $\hat{\boldsymbol{\theta}}$ maximises the **log-likelihood** $l(\boldsymbol{\theta}) = \log L(\boldsymbol{\theta})$; it is more usual to work with the log-likelihood rather than the likelihood directly.

Under a few (common) regularity conditions, the (expected) Fisher information matrix is given by the expectation of the negative Hessian of $l(\boldsymbol{\theta})$:

$$M(\boldsymbol{\theta}) = E_y \left[-\frac{\partial^2 l(\boldsymbol{\theta})}{\partial \boldsymbol{\theta} \partial \boldsymbol{\theta}^{\mathrm{T}}} \right] .$$

Informally, the Fisher information at $\hat{\boldsymbol{\theta}}$ measures the curvature of the log-likelihood around the estimate; the more "peaked" the likelihood function, the "better" we know the parameter. Slightly more formally, it can be shown (see, eg, Garthwaite, Jolliffe, and Jones 2002) that asymptotically the MLE follows the normal distribution

$$\hat{\boldsymbol{\theta}} \sim N(\boldsymbol{\theta}, M(\boldsymbol{\theta})^{-1}) .$$

That is, asymptotically, $\hat{\boldsymbol{\theta}}$ is **unbiased** with **variance-covariance** matrix

$$\mathrm{Var}(\hat{\boldsymbol{\theta}}) = M(\boldsymbol{\theta})^{-1} .$$

Hence (functions of) $M(\boldsymbol{\theta})$ summarise the precision of the MLE.

So far, in our notation we have ignored the fact that the (log-) likelihood and the Fisher information are also functions of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$. If we assume we can choose these values in an experiment, we call $\boldsymbol{\xi} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ a **design** and we can explicitly acknowledge the dependence of the information matrix on $\boldsymbol{\xi}$, $M(\boldsymbol{\theta}) = M(\boldsymbol{\xi}; \boldsymbol{\theta})$. Hence, it becomes natural to think about choosing a design to maximise (a function of) the Fisher information.

Simple example 1

Assume $y_1, \ldots, y_n$ are observations from a Binomial random variable, each with $m = 10$ trials. The likelihood and log-likelihood are given by

$$L(\theta) = \prod_{i=1}^{n} \binom{m}{y_i} \theta^{y_i} (1-\theta)^{m-y_i} ;$$

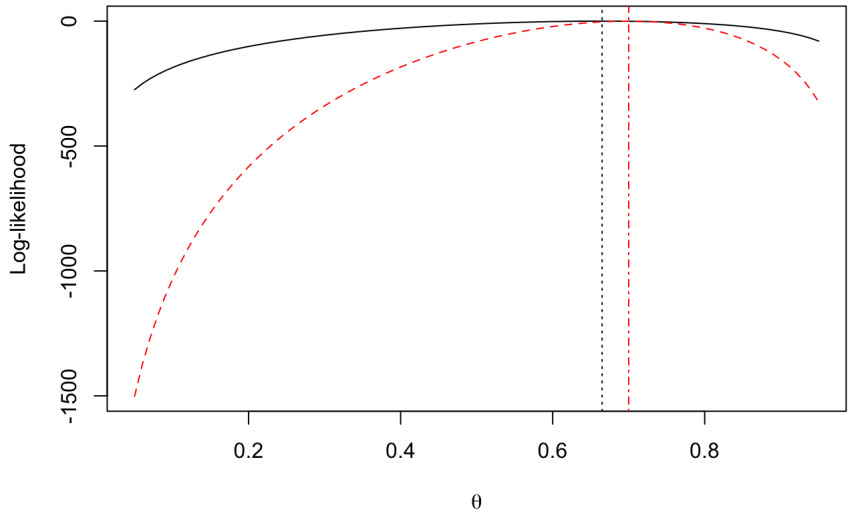
$$l(\theta) = \text{constant} + \sum_{i=1}^{n} y_i \log \theta + \left(nm - \sum_{i=1}^{n} y_i \right) \log(1-\theta) .$$

The MLE is given by $\hat{\theta} = \sum_{i=1}^{n} y_i/(nm)$ and the Fisher information by $M(\theta) = \frac{nm}{\theta(1-\theta)}$. Notice that the Fisher information is a function of the unknown parameter θ. This is common for models which are nonlinear in the unknown parameters, and creates an issue for design of experiments (see below).

Here, there are no controllable variables $\boldsymbol{x}$ (all the y_i are identically distributed) but there is still a "design" problem": how large should n be? Such **sample size** problems are common in, eg, clinical trials. As a demonstration, we can use the following R code, where we substitute the observed information (the negative Hessian of the log-likelihood at the MLE) for the expected information for two different values of n, and examine the curvature of the likelihood surface.

```
nloglik <- function(prob, y, m) -1 * sum(log(dbinom(y, m, prob)))
n <- 20
m <- 10
tp <- 0.7
y1 <- rbinom(n, m, tp)
mle1 <- nlm(nloglik, p = .5, y = y1, m = m, hessian = T)
y2 <- rbinom(5 * n, m, tp)
mle2 <- nlm(nloglik, p = .5, y = y2, m = m, hessian = T)
p <- seq(.05, .95, by = .01)
l1 <- sapply(p, nloglik, y = y1, m = m)
l2 <- sapply(p, nloglik, y = y2, m = m)
matplot(p, cbind(-1 * l1 + mle1$minimum, -1 * l2 + mle2$minimum), type = "l", ylab = "Log-likelihood", xlab = expressio
```

```
abline(v = sum(y1)/(n * m), lty = 3)
abline(v = sum(y2)/(5 * n * m), lty = 4, col = "red")
```




```
mle2$hessian / mle1$hessian
```

```
##          [,1]
## [1,] 5.304624
```

Two things to note in the above plot: 1. when we increase the sample size by a factor of five, the MLE moves closer to the true probability (the MLE is only asymptotically unbiased); and 2. the curvature of the likelihood surface (which is measured by the Fisher information) is much greater for the larger experiment. As measured by the inverse of the observed information, the larger experiment has (asymptotic) variance of the MLE about five times smaller, as we might expect.

Exercise 1 Now assume there is a single controllable variable $\boldsymbol{x}$ such that $\boldsymbol{P}(\boldsymbol{y}_i = 1) = \frac{1}{1+\exp(-\boldsymbol{x}_i\boldsymbol{\theta})}$. Either by adapting the code above, or by hand, find the expected or observed information for $\boldsymbol{\theta}$ and examine how it changes for different choices of $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ (eg produce plots like the one above for at least two different designs).

Linear models

Consider the linear model

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}\,,$$

with $\boldsymbol{X}$ a $\boldsymbol{n} \times \boldsymbol{p}$ model matrix, $\boldsymbol{\beta}$ a $\boldsymbol{p}$-vector of regression parameters and $\boldsymbol{\varepsilon} \sim \boldsymbol{N}(0, \boldsymbol{\sigma}^2\boldsymbol{I}_{\boldsymbol{n}})$, so $\boldsymbol{\theta} = (\boldsymbol{\beta}^{\mathrm{T}}, \boldsymbol{\sigma}^2)^{\mathrm{T}}$. The information matrix for $\boldsymbol{\beta}$ (treating $\boldsymbol{\sigma}^2$ as known) has a simple form:

$$\boldsymbol{M}(\boldsymbol{\xi};\, \boldsymbol{\beta}) = \frac{1}{\boldsymbol{\sigma}^2}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\,.$$

Exercise 2 Derive this equation.

Compare this information matrix to the Fisher information of the Binomial example - a key difference is that for the linear model, the information matrix does not depend on values of the unknown parameters $\boldsymbol{\beta}$ that require estimation. This is important for design; it means that we can designs experiments that maximise a function of $\boldsymbol{M}(\boldsymbol{\xi};\boldsymbol{\beta})$ without requiring a priori knowledge about the values of $\boldsymbol{\beta}$.

The least squares estimators for the linear model have the well-known form

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{y}$$

and, assuming the linear model is correct for the underlying data-generating process,

$$\hat{\boldsymbol{\beta}} \sim \boldsymbol{N}\left(\boldsymbol{\beta}, \boldsymbol{M}(\boldsymbol{\xi};\, \boldsymbol{\beta})^{-1}\right)\,.$$

For the linear model, this result is exact rather than asymptotic. However, what if the data were actually generated by a different linear model? For example,

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{X}_2\boldsymbol{\beta}_2 + \boldsymbol{\varepsilon}\,,$$

with $\boldsymbol{X}_2$ a second $\boldsymbol{n} \times \boldsymbol{p}_2$ model matrix and $\boldsymbol{\beta}_2$ a second $\boldsymbol{p}_2$-vector of parameters?

Exercise 3 Assuming that we still fit the original linear model $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, show that the variance of $\hat{\boldsymbol{\beta}}$ is unchanged but that the expected value becomes

$$\boldsymbol{E}(\hat{\boldsymbol{\beta}}) = \boldsymbol{\beta} + \boldsymbol{A}\boldsymbol{\beta}_2\,,$$

where $\boldsymbol{A} = \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{X}_2$.

The matrix $\boldsymbol{A}$ is commonly referred to as the **alias** matrix; its entries determine the biases of the MLEs. Notice that $\boldsymbol{A}$ also depends on the choice of design through the matrices $\boldsymbol{X}$ and $\boldsymbol{X}_2$. It therefore follows that if we fit a submodel to our data, the resulting bias of the parameter estimators can be influenced (and maybe reduced) through choice of design. Such situations are common - we may not have sufficient resource to design an experiment large enough to fit the full model, or we may be uncertain about which model terms should be included.

Simple example 2

The `EngrExpt` package in `R` contains a dataset `bright` for a designed experiment on the "de-inking" process of newspaper in recycling. We will use this as an example design for a linear model to examine the information and alias matrices. The data set contains columns for five controllable variables, see `?bright`. We will assume that the assumed model contains linear and two-way interactions (with corresponding columns in $\boldsymbol{X}$), and that the true model also contains three-way interactions (with columns in $\boldsymbol{X}_2$).

The features to note about the below output are the structure of the information matrix (diagonal) and the alias matrix (entries either -1 or 0). These features are common in **two-level regular fractional factorial designs** (see Wu and Hamada 2009).

```
library(EngrExpt)
Design <- 2 * as.data.frame(lapply(bright[, 1:5], as.numeric)) - 3
X <- model.matrix(~(.) ^ 2, data = Design)
X2 <- model.matrix(~(.) ^ 3, data = Design)[, -(1:16)]
t(X) %*% X
```

```
##                  (Intercept) type percent time hardness speed type:percent
## (Intercept)               16    0       0    0        0     0            0
## type                       0   16       0    0        0     0            0
## percent                    0    0      16    0        0     0            0
## time                       0    0       0   16        0     0            0
## hardness                   0    0       0    0       16     0            0
## speed                      0    0       0    0        0    16            0
## type:percent               0    0       0    0        0     0           16
## type:time                  0    0       0    0        0     0            0
## type:hardness              0    0       0    0        0     0            0
## type:speed                 0    0       0    0        0     0            0
## percent:time               0    0       0    0        0     0            0
## percent:hardness           0    0       0    0        0     0            0
## percent:speed              0    0       0    0        0     0            0
## time:hardness              0    0       0    0        0     0            0
## time:speed                 0    0       0    0        0     0            0
## hardness:speed             0    0       0    0        0     0            0
##                  type:time type:hardness type:speed percent:time
## (Intercept)              0             0          0            0
## type                     0             0          0            0
## percent                  0             0          0            0
## time                     0             0          0            0
## hardness                 0             0          0            0
## speed                    0             0          0            0
## type:percent             0             0          0            0
## type:time               16             0          0            0
## type:hardness            0            16          0            0
## type:speed               0             0         16            0
## percent:time             0             0          0           16
## percent:hardness         0             0          0            0
## percent:speed            0             0          0            0
## time:hardness            0             0          0            0
## time:speed               0             0          0            0
## hardness:speed           0             0          0            0
##                  percent:hardness percent:speed time:hardness time:speed
## (Intercept)                     0             0            0          0
## type                            0             0            0          0
## percent                         0             0            0          0
## time                            0             0            0          0
## hardness                        0             0            0          0
## speed                           0             0            0          0
## type:percent                    0             0            0          0
## type:time                       0             0            0          0
## type:hardness                   0             0            0          0
## type:speed                      0             0            0          0
## percent:time                    0             0            0          0
## percent:hardness               16             0            0          0
## percent:speed                   0            16            0          0
## time:hardness                   0             0           16          0
## time:speed                      0             0            0         16
## hardness:speed                  0             0            0          0
##                  hardness:speed
## (Intercept)                   0
## type                          0
## percent                       0
## time                          0
## hardness                      0
## speed                         0
## type:percent                  0
## type:time                     0
## type:hardness                 0
## type:speed                    0
## percent:time                  0
## percent:hardness              0
## percent:speed                 0
## time:hardness                 0
## time:speed                    0
## hardness:speed               16
```

```
solve(as.matrix(t(X) %*% X)) %*% t(X) %*% X2
```

```
##                  type:percent:time type:percent:hardness type:percent:speed
## (Intercept)                      0                      0                   0
## type                             0                      0                   0
## percent                          0                      0                   0
## time                             0                      0                   0
```

```
## hardness                          0                    0                    0
## speed                             0                    0                    0
## type:percent                      0                    0                    0
## type:time                         0                    0                    0
## type:hardness                     0                    0                    0
## type:speed                        0                    0                    0
## percent:time                      0                    0                    0
## percent:hardness                  0                    0                    0
## percent:speed                     0                    0                    0
## time:hardness                     0                    0                   -1
## time:speed                        0                   -1                    0
## hardness:speed                   -1                    0                    0
##                   type:time:hardness type:time:speed type:hardness:speed
## (Intercept)                        0               0                    0
## type                               0               0                    0
## percent                            0               0                    0
## time                               0               0                    0
## hardness                           0               0                    0
## speed                              0               0                    0
## type:percent                       0               0                    0
## type:time                          0               0                    0
## type:hardness                      0               0                    0
## type:speed                         0               0                    0
## percent:time                       0               0                   -1
## percent:hardness                   0              -1                    0
## percent:speed                     -1               0                    0
## time:hardness                      0               0                    0
## time:speed                         0               0                    0
## hardness:speed                     0               0                    0
##                   percent:time:hardness percent:time:speed
## (Intercept)                           0                  0
## type                                  0                  0
## percent                               0                  0
## time                                  0                  0
## hardness                              0                  0
## speed                                 0                  0
## type:percent                          0                  0
## type:time                             0                  0
## type:hardness                         0                 -1
## type:speed                           -1                  0
## percent:time                          0                  0
## percent:hardness                      0                  0
## percent:speed                         0                  0
## time:hardness                         0                  0
## time:speed                            0                  0
## hardness:speed                        0                  0
##                   percent:hardness:speed time:hardness:speed
## (Intercept)                            0                   0
## type                                   0                   0
## percent                                0                   0
## time                                   0                   0
## hardness                               0                   0
## speed                                  0                   0
## type:percent                           0                  -1
## type:time                             -1                   0
## type:hardness                          0                   0
## type:speed                             0                   0
## percent:time                           0                   0
## percent:hardness                       0                   0
## percent:speed                          0                   0
## time:hardness                          0                   0
## time:speed                             0                   0
## hardness:speed                         0                   0
```

Exercise 4 What does the alias matrix tell you about the bias of $\hat{\boldsymbol{\beta}}$?

Nonlinear models

If a statistical model is nonlinear in the unknown parameters, assessment of a design is more complicated. For the model

$$y_i = f(x_i; \beta) + \varepsilon_i\,, \qquad i = 1, \ldots, n\,,$$

with independent errors $\varepsilon_i \sim N(0, \sigma^2)$, the information matrix for $\boldsymbol{\beta}$ has the form

$$M(\xi; \beta) = \frac{1}{\sigma^2} F^{\mathrm{T}} F\,,$$

where $n \times p$ **sensitivity** matrix F has ijth entry $\partial f(x_i; \beta)/\partial \beta_j$ (see Atkinson, Donev, and Tobias 2007).

Exercise 5 Derive the form of this information matrix.

Exercise 6 Show that the information matrix for the linear model is a special case.

The main complication for assessing design performance for nonlinear models is the dependence of F on the value of $\boldsymbol{\beta}$ (eg see the binomial example above where the information depended on θ). Often, this dependence means designs are assessed **locally** for assumed values of $\boldsymbol{\beta}$. Other approaches attempt to robustify the design by assuming its worst-case performance across a set of values of $\boldsymbol{\beta}$ (maximin) or through **Bayesian methods**.

Bayesian methods

If a Bayesian paradigm is adopted, design performance can be assessed using the posterior variance-covariance matrix of the parameters of interest. Assume a linear regression model with conjugate prior distributions $\boldsymbol{\beta}|\sigma^2 \sim \boldsymbol{N}(\boldsymbol{\beta}_0, \sigma^2\boldsymbol{R}^{-1})$ and $\sigma^2 \sim$ Inverse-Gamma$(\boldsymbol{a}, \boldsymbol{b})$, with known prior mean $\boldsymbol{p}$-vector $\boldsymbol{\beta}$, known conditional $\boldsymbol{p} \times \boldsymbol{p}$ prior variance-covariance matrix $\sigma^2\boldsymbol{R}^{-1}$, and $\boldsymbol{a} > 0, \boldsymbol{b} > 0$. The conditional posterior variance-covariance matrix for $\boldsymbol{\beta}$ is given by:

$$\text{Var}(\boldsymbol{\beta}|\boldsymbol{y}, \sigma^2) = \sigma^2 \left(\boldsymbol{X}^{\text{T}}\boldsymbol{X} + \boldsymbol{R}\right)^{-1}.$$

It is therefore natural to use (the inverse of) this matrix to assess the quality of a design from a Bayesian perspective (Chaloner and Verdinelli 1995).

For nonlinear models, the situation tends to be more complicated. The posterior variance covariance matrix of the parameters is usually not available in closed form. The information matrix, evaluated at the posterior mode, can be used as an asymptotic approximation, but suffers once again from the problem of depending on quantities unknown prior to experimentation. In the course, we will review approaches that overcome these issues, including decision-theoretic approaches and associated computational methods (see Ryan et al. 2016, Overstall and Woods (2017))

References

Atkinson, A. C., A. N. Donev, and R. D. Tobias. 2007. **Optimum Experimental Design, with Sas**. 2nd ed. Oxford: Oxford University Press.

Chaloner, K., and I. Verdinelli. 1995. “Bayesian Experimental Design: A Review.” **Statistical Science** 10: 273–304.

Garthwaite, P. H., I . T. Jolliffe, and B. Jones. 2002. **Statistical Inference**. 2nd ed. Oxford: Oxford University Press.

Overstall, A. M., and D. C. Woods. 2017. “Bayesian Design of Experiments Using Approximate Coordinate Exchange.” **Technometrics** 59: 458–70.

Ryan, E. G., C. C. Drovandi, J. M. McGree, and A. N. Pettitt. 2016. “A Review of Modern Computational Algorithms for Bayesian Optimal Design.” **International Statistical Review** 84: 128–54.

Wu, C. F. J., and M. Hamada. 2009. **Experiments: Planning, Analysis, and Parameter Design Optimization**. 2nd ed. New York: Wiley.