Computer model calibration as a method of design

Carl Ehrett

April 9, 2018

1 Introduction

1.1 Computer experiments

Suppose that one wishes to improve one's understanding of, say, the movement of people in a crowd escaping from a building in a crisis situation. This is an example of an area in which field data are extremely difficult to acquire. Merely assembling a crowd of research subjects in one place is costly and difficult. Asking them to flee a building may result in behaviors which are unlike those in real crisis situations – but which may nonetheless present unacceptable physical risk to the subjects. Inducing them to flee through the generation of a (real or apparent) crisis is similarly infeasible. Observational data are likewise scarce here, since panic-inducing crises are by their nature difficult to predict and chaotic in ways that hinder the reliable collection of data.

In the face of these difficulties, computer models offer an alternative to the choice between attempting field data collection and giving up on the hope of progress. Using existing theory concerning human psychology and movement, it is possible to construct a computer model simulating the behavior of people evacuating from a large building. For example, the SIMULEX model described by Thompson & Marchant (1995) allows one to observe simulated evacuation behaviors in any specified building layout, using any desired physical distribution of individuals, whose individual relevant characteristics (walking speed, initial bodily orientation, etc) may be controlled by the researcher. Thus, computer models provide a means to collect data which might otherwise be largely inaccessible.

The study of computer models from a statistical perspective calls for specialized tools and techniques. Gaussian processes (GPs) are a popular tool for modeling the output of computer code. There are three reasons for this popularity: (1) The use of a GP does not require detailed foreknowledge of the approximate parametric form of the computer model. Researchers often lack such foreknowledge in the case of complex computer models. (2) GPs easily interpolate the observed data. This is an advantage when the observations come from deterministic computer code that is free of observation error. (3) The variance of a GP provides a natural form of uncertainty quantification. A Bayesian approach to the study of computer models is undertaken by Currin *et al.* (1991); the authors approach GPs as prior distributions on the unknown form of the computer model. A frequentist applications of GPs to computer models is provided by Sacks *et al.* (1989), who use GPs not only for estimating uncertainty but also as the basis for their approach to experimental design in the area of deterministic computer models. Santner *et al.* (2003) offer a comprehensive discussion of to the use of GPs for prediction, design, and sensitivity analysis with respect to computer experiments from both frequentist and Bayesian perspectives. It is against the background of these works that the past two decades of research into computer model calibration takes place.

1.2 Computer model calibration

Suppose that we wish to use the SIMULEX model to compare two different proposed building codes to be enforced in, say, St. Louis, Missouri. We may use average walking speed and average interpersonal distance as input parameters for this model, both to settle the initial physical distribution of people throughout the building and to influence their behavior during evacuation. It is well-established that average walking speed (Bornstein & Bornstein, 1976) and interpersonal distance (Sorokowska *et al.*, 2017) vary across locales. These values may be unknown for the case of St. Louis. Thus we may wish to find the true values for average walking speed and interpersonal distance in St. Louis; we may wish, in other words, to *calibrate* these parameters in the model.

Broadly, in model calibration, we may consider a model to be of the form $\eta(x, \theta)$, where (x, θ) comprise all inputs to the model. Control inputs — inputs under the control of the researcher (in the evacuation example, this would include the building layout) comprise x, whereas θ is the set of calibration inputs — parameters the values of which are not under researcher control, but rather are unknown values which must be estimated for successful simulation. Thus where f describes the true system, we consider the model to be

$$f(x, \theta) = f(x) = \eta(x, \theta) + \delta(x) \tag{1}$$

where δ describes the model discrepancy – i.e., the bias of the model as an estimate of the real system. Notice that we may write $f(x) = f(x, \theta)$ since θ does not vary in reality. To undertake model calibration, we must have access to at least some observations of the real system; it is to these real observations that we calibrate the computer model.

Much interest in the past two decades has centered on Bayesian methods for model calibration. The appeal of a Bayesian approach to model calibration lies in the fact that the calibration parameters are a source of uncertainty for the model. This uncertainty should be quantified so that its effect on the model can be made explicit. We can thus use Bayesian methods to arrive at a posterior distribution on the calibration parameters which balances our prior knowledge about the calibration parameters with what can be learned from the available data, and which also allows for accurate uncertainty quantification on the model outputs.

The work of Kennedy & O'Hagan (2001) has been influential in this area. Kennedy and O'Hagan offer a Bayesian approach to computer model calibration that allows for the uncertainty of the calibration parameters in the predictions of the resulting calibrated model. This area is furthered by Higdon *et al.* (2004), who develop an approach that undertakes model calibration with quantification of the related uncertainty, as well as explicitly incorporating uncertainty regarding the computer model output, the bias of the computer model, and uncertainty due to observation variance (of field data). That approach is further refined and exemplified by Williams *et al.* (2006). Loeppky *et al.* (2006) offer an MLE-based alternative to the Bayesian approach promulgated by Kennedy and O'Hagan, intending thereby to improve the identifiability of the calibration parameters in the face of model discrepancy. Bayarri *et al.* (2007b) extends the approach of Kennedy and O'Hagan, providing a framework that is intended to allow for simultaneous validation and calibration of a computer model (using the same training data). Though they work within the Bayesian framework set by Kennedy and O'Hagan, they mitigate the integrity of the Bayesian analysis through what they call "modularization". Modularization refers to separating sources of information, so that the model has distinct components or "modules", rather than allowing all information to combine into a single analysis under the umbrella of Bayes' theorem. Liu *et al.* (2009) focus directly on the use of modularization, exploring its advantages and disadvantages; amongst other potential motivations, they show that modularization can improve the identifiability of calibration parameters. Bayarri *et al.* (2007a) furthers the project of Bayarri *et al.* (2007b) by applying the latter's methodology to functional data. To do so, the methodology for dealing with scalar data is hierarchically applied to the coefficients of a wavelet representation of the functional data. Similarly, Paulo *et al.* (2012) focuses on applying the lessons of Bayarri *et al.* (2007b) to computer models with multivariate output.

Above, I have described model calibration as a matter of estimating the true values of unknown parameters. Indeed, that is the sort of model calibration that will form the focus on the present work. However, in a broad sense of the term, model calibration takes a second form as well. For convenience, we may distinguish this second form by referring to it as *tuning*, rather than as calibration. In model tuning, a computer model includes inputs which must be calibrated, but which have no particular interpretation; i.e., tuning parameters do not represent the truth regarding some value of the system being modeled. The work of Brynjarsdóttir & O'Hagan (2014) emphasizes the distinction between calibration and tuning. They show that whereas a well-modeled discrepancy function ($\delta(\cdot)$ in Equation 1) may not be necessary for tuning, since the tuning process can accomplish much of the work of a discrepancy function, an accurate discrepancy function is crucial for calibration, for the same reason. Letting the calibration process take on the duties of the discrepancy function reduces the identifiability of the calibration parameters.

Another extension of the framework of Kennedy & O'Hagan (2001) and Bayarri *et al.* (2007b) is the *state-aware* calibration of Brown & Atamturktur (2016). When one knows or suspects that the appropriate value of a calibration parameter θ is dependent upon the other inputs x, state-aware calibration allows for the calibration procedure to estimate θ as $\theta(x)$. This may obviate the need for a discrepancy function, either because the true value of the calibration parameter does indeed vary with the other inputs, or (in a tuning problem) because varying the tuning parameter can accomplish the work of a discrepancy function in aligning the computer model response with reality.

1.2.1 Gaussian processes

In principle, model calibration[1] need not rely on a GP emulator, or any other sort of emulator; one could (e.g.) complete a full Bayesian analysis via an MCMC chain that involves running the relevant computer model at each iteration of the chain. However, computer models are frequently too computationally expensive to allow for such profligacy. Instead, a computationally tractable emulator is constructed using a sample of observations from the computer model. As described in Section 1.1, GPs are popular prior distributions on computer model output, because (1) their use does not require detailed foreknowledge of the model function's parametric form, (2) GPs easily interpolate the (deterministic) computer model output, and (3) GPs facilitate uncertainty quantification through the variance of the posterior GP. In this section, I provide brief background on Gaussian processes and their use in regression broadly and in computer model calibration specifically.

Background

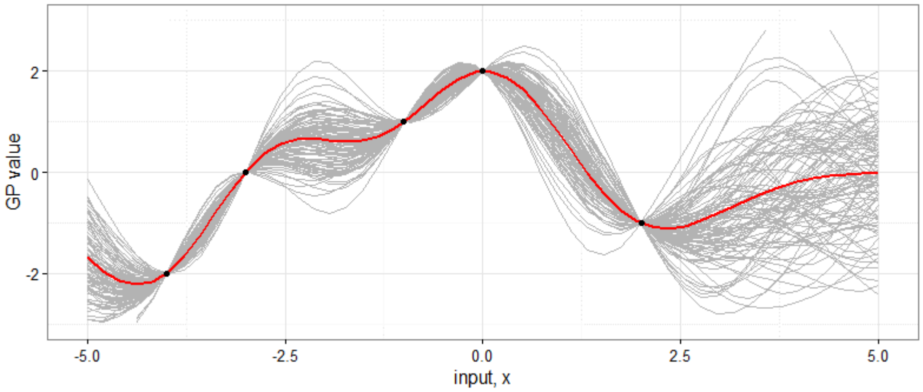

Figure 1: Example of a Gaussian process trained to interpolate five data points (black dots).

Gaussian process regression

Gaussian processes in computer model calibration

1.2.2 Markov chain Monte Carlo methods

Background

Metropolis-Hastings algorithm

Elimination of boundary constraints

1.2.3 Normalization of inputs and standardization of outputs

Blah

[1] In this work, "calibration" will usually be used broadly to include model tuning; where it is used in such a way as to exclude tuning, this will be made clear in context.

1.2.4 Computational difficulties

Blah

Likelihoods Blah

Ill-conditioned covariance matrices

2 Calibration for design

3 Application

3.1 Project background

3.2 Emulation of finite element simulator

Blah

3.2.1 Wind turbine blade simulator

Blah

3.2.2 Mathematical basis for the emulator

Blah

3.2.3 Experimental design

Blah

3.2.4 Covariance parameters

Blah

Finding covariance parameters via MCMC Blah

Grid optimization Blah

Gradient method Blah

4 MCMC using the emulator

Blah

4.1 MCMC methods

Blah

4.2 The model

Blah

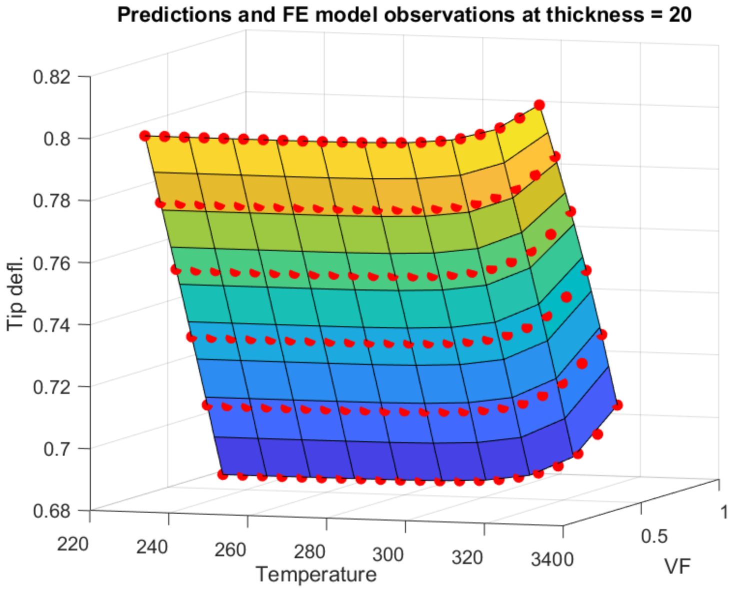

Figure 2: A slice of the GP emulator (restricted to the output for tip deflection) at thickness =20mm. Red dots are observations from the simulator.

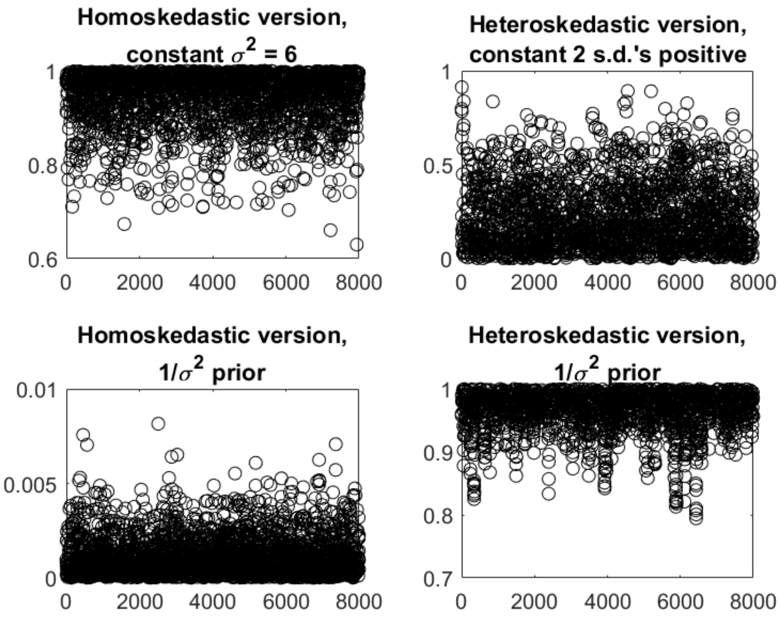

Figure 3: MCMC results at various observation variance settings.

	Heteroskedastic, constant	Homoskedastic, constant	Heteroskedastic, prior	Homoskedastic, prior
Deflection	0.749	0.729	0.659	0.709
Rotation	0.0904	0.0865	0.0773	0.0843
Cost	276.16	236.11	350.80	233.95

Table 1: Comparison of model outputs, where the desired data outputs are assumed to be either homoskedastic or heteroskedastic, with either a specified constant variance or a $1/\sigma^2$ prior.

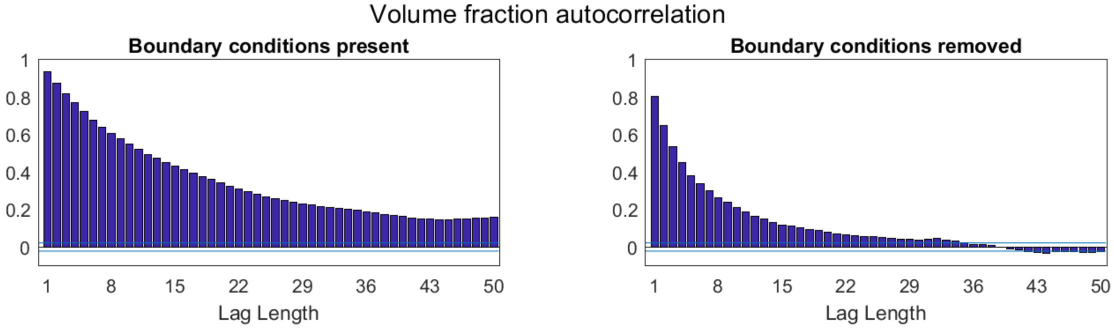

Figure 4: Auto-correlation for draws both with and without the elimination of boundary conditions.

4.2.1 Desired observation variance

4.2.2 Full model and likelihood

Blah

4.2.3 Convergence difficulties

Blah

4.2.4 Implementation of the Metropolis-Hastings algorithm

Blah

4.3 Which data to desire?

Blah

4.3.1 Motivations behind the choice of desired data

Blah

4.3.2 Differing results

Desired data d	σ^2_{defl}	σ^2_{rot}	σ^2_{cost}	$\mu_{v\|d}$	$\mu_{h\|d}$	$\sigma^2_{v\|d}$	$\sigma^2_{h\|d}$
$(0,0,0)$	375.45	277.69	2.62	0.215	$4.01 \cdot 10^{-2}$	$4.41 \cdot 10^{-2}$	$1.92 \cdot 10^{-3}$
$(0.65, 0.077, 96)$	16.74	15.25	$4.62 \cdot 10^{-7}$	$1.09 \cdot 10^{-3}$	$3.36 \cdot 10^{-4}$	$1.02 \cdot 10^{-5}$	$9.97 \cdot 10^{-6}$

Table 2: Comparison of results for two different (low) values of d. Values listed are, respectively, the posterior means for the observation variance of each model output, posterior means for volume fraction (v) and thickness (h), and posterior variance of volume fraction and thickness.

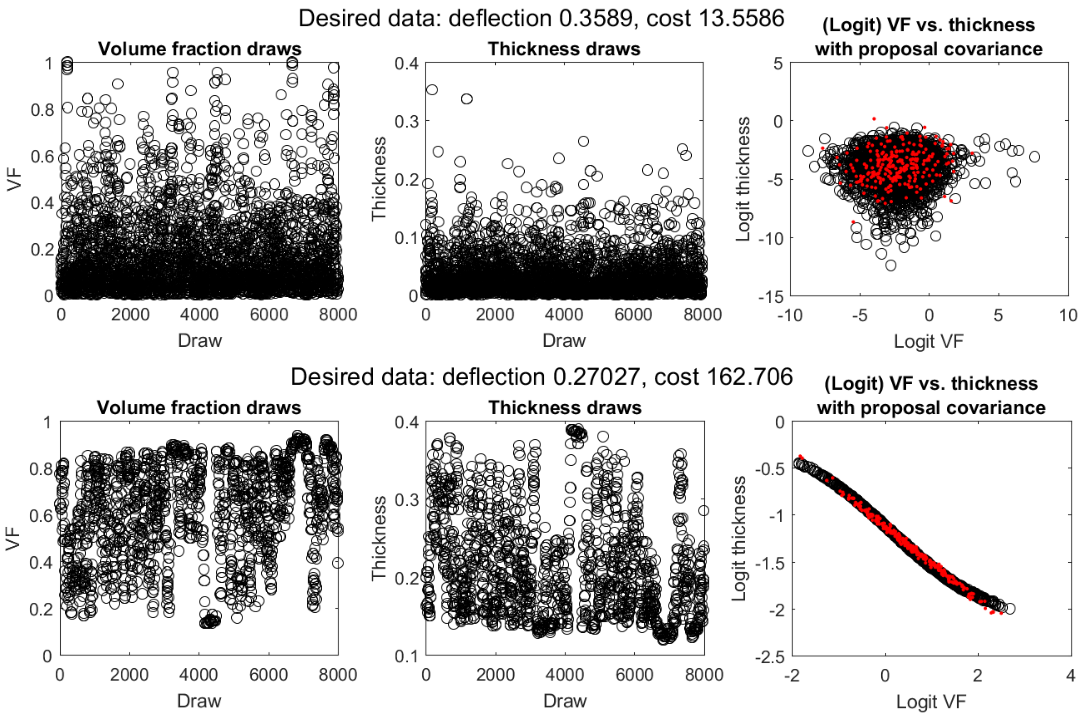

Figure 5: MCMC results for low deflection and cost (top row) and low deflection with easily achievable cost (bottom row).

4.4 Exponentially distributed desired data

Blah

4.4.1 Motivation

Blah

4.4.2 Implementation and results

Blah

4.5 Identifiability issues

Blah

5 Future work

Blah

5.1 Alternative means of handling cost

Blah

5.1.1 Removing cost from the model

Blah

5.1.2 Alternative priors for controlling cost

Blah

5.2 Building a desired data response surface

Blah

5.3 Implementing Hamiltonian Monte Carlo

Blah

5.3.1 Hamiltonian Monte Carlo

Blah

5.3.2 Benefits

Blah

5.4 Model discrepancy

Blah

6 Conclusion

Blah

References

Bayarri, M. J., Berger, J. O., Cafeo, J., Garcia-Donato, G., Liu, F., Palomo, J., Parthasarathy, R. J., Paulo, R., Sacks, J., & Walsh, D. 2007a. Computer Model Validation with Functional Output. *The Annals of Statistics*, **35**, 1874–1906.

Bayarri, Maria J, Berger, James O, Paulo, Rui, Sacks, Jerry, Cafeo, John A, Cavendish, James, Lin, Chin-Hsu, & Tu, Jian. 2007b. A Framework for Validation of Computer Models. *Technometrics*, **49**(2), 138–154.

Bornstein, Marc H., & Bornstein, Helen G. 1976. The pace of life. *Nature*, **259**(5544), 557–559.

Brown, D. Andrew, & Atamturktur, Sez. 2016. Nonparametric Functional Calibration of Computer Models. feb.

Brynjarsdóttir, Jenni, & O'Hagan, Anthony. 2014. Learning about physical parameters: The importance of model discrepancy. *Inverse Problems*, **30**(11).

Currin, Carla, Mitchell, Toby, Morris, Max, & Ylvisaker, Don. 1991. Bayesian Prediction of Deterministic Functions, with Applications to the Design and Analysis of Computer Experiments. *Journal of the American Statistical Association*, **86**(416), 953–963.

Higdon, Dave, Kennedy, Marc, Cavendish, James C., Cafeo, John A., & Ryne, Robert D. 2004. Combining Field Data and Computer Simulations for Calibration and Prediction. *SIAM Journal on Scientific Computing*, **26**(2), 448–466.

Kennedy, Marc C., & O'Hagan, Anthony. 2001. Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **63**(3), 425–464.

Liu, F., Bayarri, M. J., & Berger, J. O. 2009. Modularization in Bayesian analysis, with emphasis on analysis of computer models. *Bayesian Analysis*, **4**(1), 119–150.

Loeppky, Jason L., Bingham, Derek, & Welch, William J. 2006. Computer Model Calibration or Tuning in Practice. *Technometrics*.

Paulo, Rui, García-Donato, Gonzalo, & Palomo, Jesús. 2012. Calibration of computer models with multivariate output. *Computational Statistics and Data Analysis*, **56**, 3959–3974.

Sacks, Jerome, Welch, William J., Mitchell, Toby J., & Wynn, Henry P. 1989. Design and Analysis of Computer Experiments. *Statistical Science*, **4**(4), 409–423.

Santner, Thomas J., Williams, Brian J., & Notz, William I. 2003. *The Design and Analysis of Computer Experiments*. New York: Springer.

Sorokowska, Agnieszka, Sorokowski, Piotr, Hilpert, Peter, Cantarero, Katarzyna, Frackowiak, Tomasz, Ahmadi, Khodabakhsh, Alghraibeh, Ahmad M., Aryeetey, Richmond, Bertoni, Anna, Bettache, Karim, Blumen, Sheyla, Błażejewska, Marta, Bortolini, Tiago, Butovskaya, Marina, Castro, Felipe Nalon, Cetinkaya, Hakan, Cunha, Diana, David, Daniel, David, Oana A., Dileym, Fahd A., Domínguez Espinosa, Alejandra del Carmen, Donato, Silvia, Dronova, Daria, Dural, Seda, Fialová, Jitka, Fisher, Maryanne, Gulbetekin, Evrim, Hamamcolu Akkaya, Aslhan, Hromatko, Ivana, Iafrate, Raffaella, Iesyp, Mariana, James, Bawo, Jaranovic, Jelena, Jiang, Feng, Kimamo, Charles Obadiah, Kjelvik, Grete, Koç, Frat, Laar, Amos, de Araújo Lopes, Fívia, Macbeth, Guillermo, Marcano, Nicole M., Martinez, Rocio, Mesko, Norbert, Molodovskaya, Natalya, Moradi, Khadijeh, Motahari, Zahrasadat, Mühlhauser, Alexandra, Natividade, Jean Carlos, Ntayi, Joseph, Oberzaucher, Elisabeth, Ojedokun, Oluyinka, Omar-Fauzee, Mohd Sofian Bin, Onyishi, Ike E., Paluszak, Anna, Portugal, Alda, Razumiejczyk, Eugenia, Realo, Anu, Relvas, Ana Paula, Rivas, Maria, Rizwan, Muhammad, Salkičević, Svjetlana, Sarmány-Schuller, Ivan, Schmehl, Susanne, Senyk, Oksana, Sinding, Charlotte, Stamkou, Eftychia, Stoyanova, Stanislava, Šukolová, Denisa, Sutresna, Nina, Tadinac, Meri, Teras, Andero, Tinoco Ponciano, Edna Lúcia, Tripathi, Ritu, Tripathi, Nachiketa, Tripathi, Mamta, Uhryn, Olja, Yamamoto, Maria Emília, Yoo, Gyesook, & Pierce, John D. 2017. Preferred Interpersonal Distances: A Global Comparison. *Journal of Cross-Cultural Psychology*, **48**(4), 577–592.

Thompson, Peter A, & Marchant, Eric W. 1995. A Computer Model for the Evacuation of Large Building Populations. *Fire Safety Journal*, **24**, 131–148.

Williams, Brian, Higdon, Dave, Gattiker, Jim, Moore, Leslie, McKay, Michael, & Keller-McNulty, Sallie. 2006. Combining experimental data and computer simulations, with an application to flyer plate experiments. *Bayesian Analysis*, **1**(4), 765–792.