An Estimation of The Unknown Theory Constants Using A Simulation Predictor

Jeong-Soo Park* and Keon-Woo Lim**

Abstract

A statistical method is described for estimation of the unknown constants in a theory using both of the computer simulation data and the real experimental data. The best linear unbiased predictor based on a spatial linear model is fitted from the computer simulation data alone. Then nonlinear least squares estimation method is applied to the real experimental data using the fitted prediction model as if it were the true simulation model. An application to the computational nuclear fusion devices is presented, where the nonlinear least squares estimates of four transport coefficients of the theoretical nuclear fusion model are obtained.

1. Introduction

Scientific researchers often use complex computer simulation programs for theoretical investigations because some physical experiments are too expensive or simply impossible. One feature of a computer simulation experiment, different from a physical experiment, is that the output is often deterministic - the response is observed without measurement error. This calls for distinct techniques useful in modeling deterministic systems.

Since simulation codes are often computationally very expensive to run, a careful selection of inputs and an efficient analysis of its outputs are necessary. In our application to nuclear fusion devices (called *tokamaks*), a single run of the computer simulation code (called BALDUR) requires about three to five minutes of CPU time on a supercomputer (CRAY-2). For many purposes, however, an emulator of the code based on a statistical prediction model is sufficient and can be run more cheaply than the original code. Thus, we model the response of computer simulation code as the realization of a stochastic process which has been succesively used in design and analysis of computer experiments[9]. This model is adapted from the "universal kriging" in the spatial statistics literature[7]. This approach provides a statistical basis for analysing deterministic data, for designing experiments for efficient prediction and for comparison of

* 전남대학교 통계학과

** 광주대학교 전자계산학과

※ This paper was supported by NON-DIRECTED RESEARCH FUND, Korea Research Foundation, 1993.

computer-encoded theory and experimental data in analysis of complex systems.

An objective of research in a large class of dynamic systems is to determine any unknown universal constants or coefficients (denoted by c^*) in a theory. The coefficients can be determined by "tuning" the computer model to the real data so that the tuned code gives a good match to data. In other words, this method involves simulating a theory model with ranges of coefficients and selecting elements from these ranges which best match the simulated data with the real data. The accurate estimation of such constants is very important, in our application, for designing the next generation of thermo-nuclear reactors. This connection of a computational and physical experiment requires the application of new statistical methods.

One similar problem is found in chemical kinetics where Miller and Frenklach (1983) used the response surface methodology. However, Sack, Schiller and Welch (1987) illustrated that their approach based on spatial prediction model is more flexible and efficient than the response surface methodology in handling computer simulation obsevations. Another interesting problem related to our method is found in the econometrics literature what is known as "calibration" [3], although only the economic time series models are considered. The unknown theory coefficients are estimated basically by the method of moments in which population moments are computed by the simulations of a model in economics theory.

This article deals with a nonlinear least squares estimation method (NLSE), where the spatial model is fitted to the computer simulation data alone, and then from the real data found nonlinear least squares estimates of c^* using the fitted spatial model as if it were the true simulation model. In section 2, we outline a model for computer simulation experiments. The NLSE formulation is given in section 3. In the application to nuclear fusion devices given in section 4, a cheaper emulator of BALDUR code has been constructed, and the universal constants were estimated from data of two tokamaks (named ASDEX and PDX). Finally, in section 5, we provide some concluding remarks.

2. A Statistical Model Approximating Computer Simulations

Following Sacks, Welch, Mitchell and Wynn (1989, abbreviatedly SWMW), we adopt a spatial regression model which treats the computer simulation response $Y(x)$ as a realization of a random function superimposed on a regression model[9],

$$Y(\chi) = \sum_{j=1}^{k} \beta_j f_j(\chi) + Z(\chi), \tag{2.1}$$

where f's are known functions and β's are unknown regression coefficients. Here the random process $Z(.)$ representing the systematic departure from the assumed linear model is assumed to be a Gaussian process with mean zero and covariance $COV(t,u) = \sigma_z^2 R(t,u)$ between $Z(t)$ and $Z(u)$, for $t = (t_1, ..., t_d), u = (u_1, ..., u_d)$, where σ_z^2 is the process variance (a scale factor) and R(t, u) is the correlation function. The rationale is that departures of the complex response from the simple regression model, though deterministic, may resemble a sample path of a (suitably chosen) stochastic process Z.

Computer observations are sometimes on the subject on measurement errors which may be due to approximation, round-off error, or Monte-Carlo routines in the simulation code. For a given set of design sites $\{s_1, ..., s_n\}$, let y be an $n \times 1$ vector of observations from a computer experiment given by

$$y(s_i) = Y(s_i) + \varepsilon_i, 1 \leqq i \leqq n, \tag{2.2}$$

where measurement errors ε_i are assumed to be uncorrelated, mean zero Normal random variables with constant variance $Var(\varepsilon_i) = \sigma_\varepsilon^2$, and assumed to be independent of $Z(x)$. In our application, the ε_i enters because there are Monte-Carlo integrations performed in BALDUR.

Some possible choices of covariance function are from the power exponential family which is given by

$$R(t, u) = \exp\left\{-\theta \sum_{i=1}^{d} |t_i - u_i|^2\right\}, \tag{2.3}$$

where the $\theta \geqq 0$. The non-negative parameter θ determines the covariance structure of Z: small θ reflects large correlations between nearby observations while large θ reflects small nearby correlations. It is thus related to the smoothness of the response. Of course, many other covariance functions are possible[7].

Once a covariance function and its parameters are specified, one can predict $Y(x)$ based on the model (2.1) using the observations $y(s)$. For the prediction formula, define the n $\times$n matrix V and n$\times$k matrix F by

$$V = [R(s_i, s_j)]_{1 \leqq i \leqq, \leqq j \leqq n} + \gamma_c^2, \tag{2.4}$$

where $\gamma_c^2 = \sigma_c^2/\sigma_z^2$ which is the ratio of noise versus signal variance, and

$$F = [f_l(s_i)]_{1 \leqq i \leqq n, 1 \leqq l \leqq k},$$

where $(s_1, ..., s_n)$ are the design sites. Here $\sigma_z^2 V$ is a covariance matrix between observations (or design sites), and F is so-called a design matrix. For any prediction site x, the n $\times$ 1 vector v_x and k $\times$ 1 vector f_x are defined by

$$v'_x = [R(s_1, x), ..., R(s_n, x)],$$

$$f'_x = [f_1(x), ..., f_k(x)],$$

respectively. Here v_x is a correlation vector between design sites and a prediction site x. Then the best linear unbiased predictor (BLUP) of $Y(x)$ given the observation vector y is (see SWMW or [7], pp. 44-58)

$$\hat{Y}(x) = [v'_x, f_x] \begin{pmatrix} V & F \\ F' & 0 \end{pmatrix}^{-1} \begin{pmatrix} y \\ 0 \end{pmatrix} = f_x \hat{\beta} + v'_x V^{-1}(y - F\hat{\beta}) \tag{2.5}$$

where $\hat{\beta} = (F'V^{-1}F)^{-1}F'V^{-1}y$ is the generalized least squares estimator of β. The mean squared error of prediction is

$$MSE(x) = MSE(\hat{Y}(x)) = E_\theta[\hat{Y}(x) - Y(x)]^2, \tag{2.6}$$

and the normalized mean squared error, $mse(x) = MSE(x)/\sigma_z^2$ is

$$mse(x) = R(x,x) - [v'_x f'_x] \begin{pmatrix} V & F \\ F' & 0 \end{pmatrix}^{-1} \begin{pmatrix} v_x \\ f_x \end{pmatrix}. \tag{2.7}$$

In the absence of measurement error ($\sigma_c^2 = 0$ in (2.2)), the prediction surface interpolates the observations because the predictor $\hat{Y}(s_i)$ at a design point s_i has $mse(s_i) = 0$," i. e.," $\hat{Y}(s_i) = Y(s_i)$ (see [7], pp. 44-58). This is one of the reasons why the model is used for deterministic data analysis of computer experiments. If $\sigma_c^2 > 0$, then $\hat{Y}(x)$ smoothes the observed data; smaller θ and larger γ_c^2 give smoother predictions. For a given "weight measure" μ on a support set χ, the (normalized) integrated mean squared error (IMSE) is

$$IMSE = \int_\chi mse(x) d\mu(x). \tag{2.8}$$

Note that neither mse nor IMSE depend on the data y nor on the unknown parameters β and σ_z^2. This makes it possible to design an experiment before taking the data, i.e., to select the observation sites which optimize some criteria such as IMSE and maximum mse (see SWMW or Sacks, Schiller & Welch, 1987, on this direction)[8], [9].

After computer simulation data have been collected at the design sites, maximum likelihood estimators (MLE) of the model parameters are computed to build a prediction model. Since we assume $y(x)$ has a multivariate Normal distribution with mean Fβ and covariance matrix σ_z^2V, the likelihood function of y is

$$L(y; \theta, \beta, \sigma_z^2, \gamma_c^2, x) = \frac{(2\pi\sigma_z^2)^{-n/2}}{\sqrt{|V|}} \exp\left(\frac{(y - F\beta)' V^{-1}(y - F\beta)}{2\sigma_z^2} \right). \tag{2.9}$$

When θ and γ_c^2 are specified, the MLE of σ_z^2 is given by

$$\hat{\sigma}_z^2 = \frac{1}{n}(Y - F\hat{\beta})' V^{-1}(Y - F\hat{\beta}), \tag{2.10}$$

where $\hat{\beta}$ is the generalized least squares estimator of β as in

(2.5). Then -2 times the log likelihood function (except for constants) with $\hat{\beta}$ and $\hat{\sigma}_z^2$ plugged in is

$$\lambda = n\log\hat{\sigma}_z^2 + \log|V|. \tag{2.11}$$

Since the likelihood equations do not lead to a closed form solution, a numerical optimization procedure is used. We used a quasi-Newton optimizer with multiple initial values because of multi-modality of the likelihood surface.

Some combinations of β's will determine the prediction model, but the following simple model is used in our application:

$$Y(x) = \beta_0 + \beta_1\chi_1 + \cdots + \beta_d x_d + Z(x). \tag{2.12}$$

Of course, many other models are possible. To select the best prediction model, the "forward" or "backward" stepwise selection procedures may be possible (see Welch et. al. (1992) on this direction)[12].

After fitting a prediction model, the best additive approximation technique can be used to estimate the main effects of input variables which were defined in SWMW (1989). We applied the ACE[1] on $\hat{Y}(t_i)$ and t_i, $i = 1,2,\cdots,I$, where t_i's are random vectors having uniform distribution over the prediction region. ACE is a useful exploratory modeling tool to help determine which of the response $\hat{Y}(t_i)$ and independent variables $x_1, \cdots, x_d$ are in need of nonlinear transformation, and what type of transformation is needed. Such plots with variations, resulted from the ACE, can be used as a guide for summarizing and guessing the fitted model (see Figure 1 in section 4). The variations with respect to a variable X_k are obtained by $\hat{Y}(t_i)$ minus the sum of the transformations corresponding to the independent variables except the variable X_k.

3. Nonlinear Least Squares Estimation Using A Simulation Predictor

In this section, we describe nonlinear least squares method for estimating unknown universal constants c^* in a computer code using real experimental data (or database) and computer data. In collecting computer data, some candidates of c^* are used as inputs of computer simulation code with other variables. We introduce the following notations for q universal constants:

d : dimension of input variables of computer code
c^* : the true (unknown) universal constants (q dimensional),

ξ : the independent variables in experimental database(d-q dimensional),
z : the experimental observations in database; $z = z(c^*, \xi)$,
c : the input variables of computer code corresponding to c^* (q dimensional),

w : the input variables of computer code corresponding to ξ (d-q dimensional),
n : number of observations for computer code,
m : number of observations in experimental database,
Y : the true computer code as a function of c and w.

Here $s' = (c, w)$ represents a design site selected for a computer experiment. Thus the computer response y is a function of s, i.e.,
$y = Y(c, w) + \epsilon$ as in (2.2).

Since we assume the true simulation code is close to the real experimental data with some variation, the following model is used:

$$z = Y(c^*, \xi) + e, \tag{3.1}$$

where e is assumed to be independent and identically distributed with mean 0 and variance σ_e^2, and it is also assumed to be independent of Y and ϵ in (2.2).

A major difficulty of the computer experiment in our application is that one run of BALDUR takes approximately 3 to 5 CPU minutes on a CRAY-2 supercomputer. If we use a nonlinear regression technique, then c^* may be estimated by minimizing the residual sum of squares

$$RSS(c^*) = \sum_{i=1}^{m}[z_i - Y(c^*, \xi_i)]^2, \tag{3.2}$$

where z_i is an observed response from real experiments and $Y(c^*,/\xi_i)$ is the corresponding theoretical value (an output of BALDUR) at the experimental point ξ_i. Since there are 74 observations in the experimental data set, one evaluation of $RSS(c)$ takes about 74 × 4 minutes (5 hours) on a CRAY-2. It is too time-consuming to run the code as many times as needed for an iterative nonlinear optimizer to find c^*.

Nonlinear least squares method first fits the model (2.1) by MLE using computer data alone, then by treating the fitted prediction model as if it were the true model, find the nonlinear least squares estimators ($\hat{c}^*$) so that the simulation predictor gives the best match to the experimental data.
That is, find $\hat{c}^*$ such that minimizes

$$\hat{RSS}(c^*) = \sum_{i=1}^{m}[z(c^*, \xi_i) - \hat{Y}(c^*, \xi_i)]^2, \tag{3.3}$$

where $\hat{Y}(c^*, \xi_i)$ is the best linear unbiased prediction of the true computer response Y at (c^*, ξ_i). Since it is difficult to have a closed form of the first derivative of (3.3) with respect to c^*, a numerical optimization routine is necessary to find $\hat{c}^*$. Note that $\hat{Y}$ is a (computationally) cheaper emulator of the expensive computor simulation code. This makes the problem computationally feasible.

The advantages of this method are that it is reasonably easy and cheap to implement, and that the computer and experimental data are uncoupled. The prediction residuals $z(c^*, \xi_i) - \hat{Y}(c^*, \xi_i)$ can be used to check the validities of the prediction model, $\hat{c}^*$ and the least squares estimation method. Note that the selection of a prediction model for $\hat{Y}(c^*, \xi)$ is important because the values of $\hat{c}^*$ may vary according to the selected prediction model. In our application the model (2.12) is used.

4. An Application to Nuclear Fusion Model

In this section we describe how the NLS method of the previous section is applied to nuclear fusion model. More details including the data sets can be found in Park (1991).

4.1 Problem Formulation.

An objective of research in nuclear fusion reactors (tokamaks) is to understand the transport mechanism governing the process, and in particular, to obtain an appropriate model for the global energy confinement time and to determine the parameters (transport coefficients or rate constants) in a transport model.
Since some of the constants can not be mathematically determined[10], a systematic statistical method is required. This method may need to use both experimental data and computer data obtained by running a tokamak simulation code (Called BALDUR, [11])based on a theoretical model, because it is difficult to extract information on the constants from experimental data alone.

One of the simple measures of energy efficiency in tokamak is the global energy confinement time τ_E. The theoretically-based confinement model may be written as[4] :

$$\tau E = f(c^*, \alpha, R, P, I, N, B), \tag{4.1}$$

where f is a known function (calculated by a complex code), a and R are the minor and major radii of the tokamak, respectively, P is the input total power, I is the plasma current, N is the electron density, B is the toroidal magnetic field and $c^* = (c_1^*, \cdots, c_4^*)$ are the adjustable constants determining energy transfer by turbulent modes known as drift waves, rippling, resistive ballooning and critical value of η_i, respectively. Of course, there are several other variables which are not the major ones.

The experimental data were taken from the database collected by S. Kaye for two tokamaks: ASDEX (32) in Germany, PDX (42) in Princeton, which have fixed values of R and a, with number of observations in parentheses.

Following the previous statistical analysis[4] on the experimental data, we take $\log_{10}$ transformations to P, I, N, B and τ_E. Therefore the variables considered in this study are c_1, c_2, c_3, c_4, $\log P, \log I, \log N, \log B$ and $\log \tau_E$. The first four c's are input variables of the BALDUR code corresponding to the true coefficients $c_1^*, c_2^*, c_3^*$and c_4^*. Note that the

experimental data consists of only the second four variables and $\log_{\tau_E}$ whereas the computer data consists of eight independent variables and $\log_{\tau_E}$ obtained from BALDUR.

4.2 Design and Analysis of Tokamak Simulation Experiments.

In selecting input sites of the simulation code, we used data-adaptive sequential optimal designs as the following manner: for initially chosen parameters (such as θ and γ_c^2) of the model (2.1), we first find the optimal design which minimizes the IMSE given in (2.8), and use this design to obatain computer observations, then find MLE's of the model parameters, and use them to choose the next stage optimal design under the condition that the previous design is given (see[6] for details of the designs). Following this procedure we obtained 66 and 64 observations from ASDEX and PDX tokamak simulators, respectively.

Table 1 shows the parameter estimates obtained from the computer data, where β_2 and β_4 is small for both tokamaks. This indicates that the effects of c_2 and c_4 are small. Note that the intercept value (β_0) for $\log_{\tau_E}$ of PDX is less than that of ASDEX. This indicates that PDX is cooler than ASDEX for similar input parameters. Also note that σ_c^2 ($= \sigma_z^2 \times \gamma_c^2$) for ASDEX is bigger than that of PDX, which means that computer observations of ASDEX have larger "measurement" errors than those of PDX.

The main effects plot with variations (drawn based on ACE) for ASDEX in Figure 1 also illustrate that variable c_2, c_4 and $\log B$ have a little effect on $\log_{\tau_E}$. The variable $\log P$ turns out the most strong factor effecting negatively on $\log_{\tau_E}$. The variables $\log I$ and $\log N$ make positive effects whereas c_1 and c_3 make negative effects. The plot for PDX is very similar to Figure 1 except that variable $\log B$ makes positive effect. This conclusion is consistent to the expectation from a theory and to the finding of Kaye and Goldston[4] who fitted the experimental data.

Symbol	Descriptions	ASDEX value	PDX value
n	computer data sample size	66	64
$\beta 0$	Intercept for Y	-1.41	-1.81
$\beta 1$	Intercept for C_1	-0.28	0.30
$\beta 2$	Intercept for C_2	0.07	-0.07
$\beta 3$	Intercept for C_3	-0.17	-0.11
$\beta 4$	Intercept for C_4	0.08	0.00
$\beta 5$	Intercept for log P	-0.71	-0.33
$\beta 6$	Intercept for log I	0.25	0.17
$\beta 7$	Intercept for log N	0.15	-0.03
$\beta 8$	Intercept for log B	0.09	0.28
θ	Correlation parameter	2.24	10.09
σ^2_z	Variance of Y	0.45	0.46
σ^2_c	measurement error variance	0.023	0.002

4.3 Estimation of The Unknown Theory Constants (c^*).

The following (4.2) presents the results of c^* estimation - the ones of most interest. In computing the *RSS* of (3.3), each machine was treated independently (mainly because they have different geometry and different design region). Thus we used the objective function in optimization to find c^* as a sum of the corresponding function in (3.3) for each tokamak. We used a quasi-newton optimization routine to find $\hat{c}^*$. Several starting values were tried to avoid the local minima.

Our NLS estimates of c^* using a fitted model are:

$$\hat{c}_1^* = 0.140, \hat{c}_2^* = 2.282,$$
$$\hat{c}_3^* = 2.397, \hat{c}_4^* = 0.546. \qquad (4.2)$$

From the main effects study on the computer data, we at least know that c_2^* and c_4^* are almost unestimable because c_2^* and c_4^* have a little effect on $\log_{\tau_E}$.
So the estimates of c_2^* and c_4^* is not reliable at this time. Figure 2 shows residual plot (residual vs. predicted values).

5. Concluding Remarks

We have considered nonlinear least squares estimation of

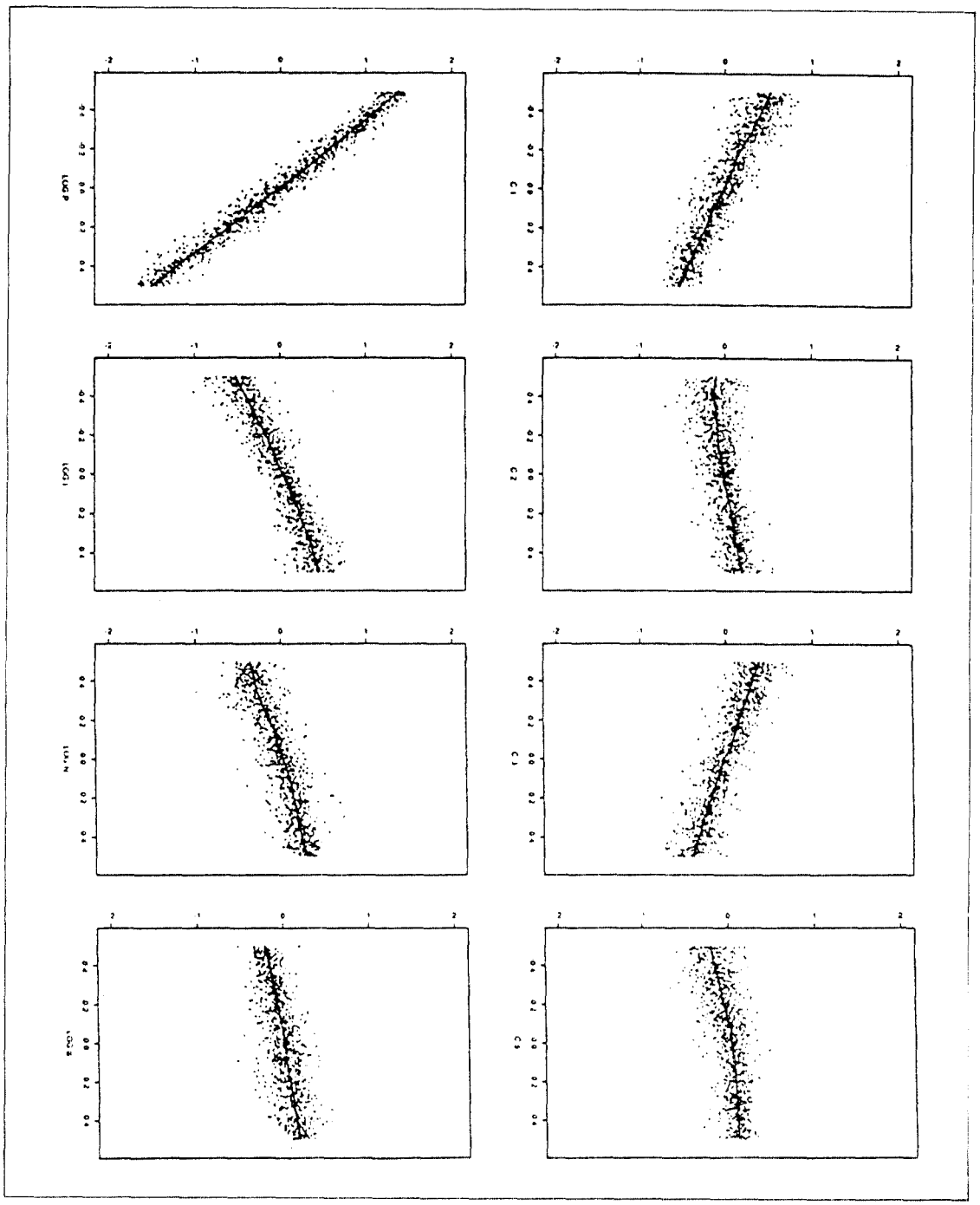


Figure 1. Main effect plots with variations for ASDEX tokamak simulator drawn by the ACE method.

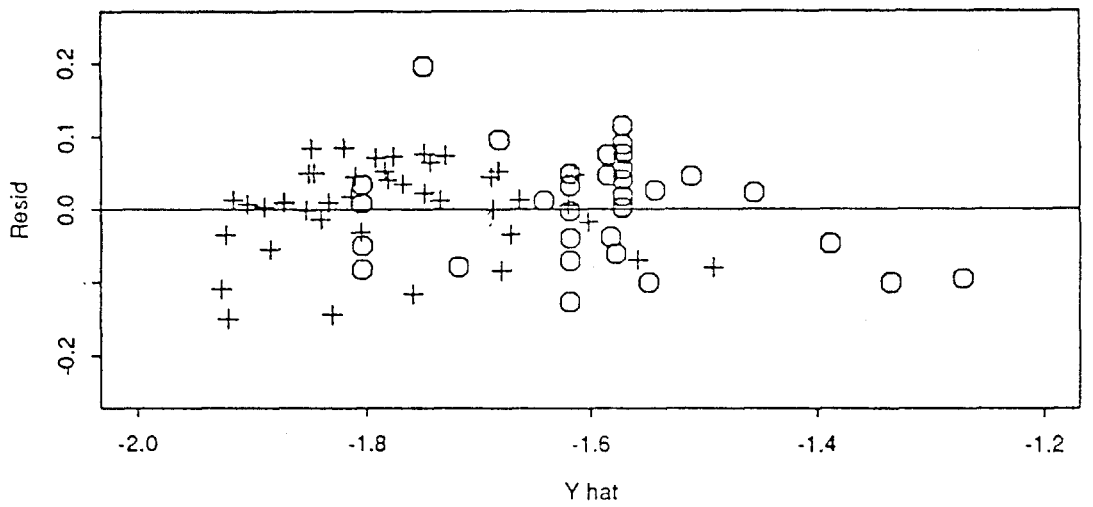


Figure 2. Residual vs. Ŷ plot for tokamak experimetal data. Octagon - ASDEX, cross - PDX.

the unknown constants using the Kriging prediction models which were fitted by the computer data. So this method is different from the ordinary nonlinear regression where a known regression function is used instead of a predictor. Thus we need some justifications that the estimation method work well in estimating the true constants. The simulation study using some toy-models reported in Park[6] suggest that it does not work badly. As we are aware of no directly relevant asymptotic (or finite sample) theory to justify our approach (the estimation and confidence intervals), it is an open problem to show the statistical properties of $\hat{c}^*$ such as the consistency and the asymptotic normality. This work may provide the standard error formula for the estimated constants.

We do not claim that the methodology presented here is the final answer, but it seems to be a good start. Based on the assumed normality of the responses of computer experiments, Park[6] considered maximum likelihood estimations of the unknown constants using both data. The results of this research will be shown in another article (Cox, Park, and Singer, 1992). If there is prior knowledge about the constant vector c^*, the Bayesian estimation approach may be reasonable. It is anticipated that still other methods will be proposed in the future.

The methodology we have proposed here may prove useful in other applications in other disciplines where the unknown constants in a theory must be estimated using complex computer simulation codes and real experimental data.

References

[1] Breiman, L., and Friedman, J. H. (1985). Estimating Optimal Transformations for Multiple Regression and Correlations. *Journal of The American Statistical Association*, 80, 580-619.

[2] Cox, D. D., Park, J. S., and Singer, C. E. (1992). A Statistical Method for Tuning a Computer Code to a Data Base. *Technical Report No.45*, Department of Statistics, University of Illinois at Urbana-Champaign, IL.

[3] Gregory, A. W., and Smith, G. W. (1990). Calibration As Estimation. *Econometric Reviews*, 9, 57-89.

[4] Kaye, S. M., and Goldston, G. C. (1985). Global Energy Confinement Scaling for Neutral-Beam-Heated Tokamak. *Nuclear Fusion*, 25 , 65-69.

[5] Miller, D., and Frenklach, M. (1983). Sensitivity Analysis and Parameter Estimation in Dynamic Modeling of Chemical Kinetics. *International Journal of Chemical Kinetics*, 15, 677-696.

[6] Park, J. S. (1991). *Tuning Complex Computer Codes to Data and Optimal Designs*. Unpublished Ph.D. Thesis, Department of Statistics, University of Illinois at Urbana-Champaign, IL.

[7] Ripley, B. (1981). *Spatial Statistics*. John Wiley, New York.

[8] Sacks, J., Schiller, S. B. and Welch, W. J. (1987). Designs for Computer Experiments. *Technometrics*, 31 , 41-47.

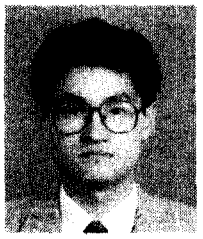
[9] Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989). Design and Analysis of Computer Experiments (with discussion). *Statistical Science*, 4 , 409-435.

[10] Singer, C. E. (1988). Theoretical Particle and Energy Flux Formulas for Tokamaks. *Comments on Plasma Physics and Controlled Fusion*, 11 , 165.

[11] Singer, C. E., Post, D. E., Mikkelsen, D. R., Redi, M. H., et. al. (1988). BALDUR: A One-Dimensional Plasma Transport Code. *Computer Physics Communications*, 49 , 275-398.

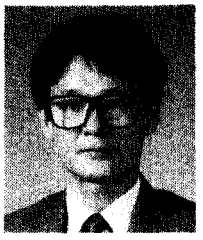
[12] Welch, W., Buck, R., Sacks, J., Wynn, H., Mitchell, T., and Morris, M. (1992). Screening, Predicting, and Computer Experiments. *Technometrics*, 34, 15-25.

저자소개



박정수

1981년 2월 전남대학교 수학교육과
1983년 2월 서울대학교 계산통계학과(석사)
1991년 10월 미국 일리노이대학교 통계학과(박사)
1992년 9월~현재 전남대학교 통계학과 전임강사



임건우

1986년 2월 광주대학교 전자계산학과
1988년 8월 미국웨스턴 일리노이 대학교 전자계산학과(석사)
1990년 3월~현재 광주대학교 전자계산학과 조교수