Automatic Complexity Control for System Identification

Ali Ghodsi and **Dale Schuurmans**
School of Computer Science
University of Waterloo
Waterloo, ON, Canada, N2L 3G1
{aghodsib,dale}@cs.uwaterloo.ca

Abstract

As a prerequisite for system identification based on c-mean clustering (FCM), it is necessary to assign the number of underlying partitions to be used for a given data set. However, for the FCM clustering algorithm it is not known how to assign the number of clusters optimally *a priori*, and the problem of selecting an appropriate number of clusters is usually treated heuristically. In this paper we derive a theoretical criterion for assigning the appropriate number of clusters. We use a generalization of Stein's unbiased risk estimator (SURE) to derive a generic criterion that defines the optimum number of clusters to use for a given problem. The efficacy of this criterion is illustrated in different experiments, including a benchmark problem involving the prediction of a chaotic time series.

1 Introduction

Fuzzy system identification has attracted a lot of interest in the past. With this technique, it is usually assumed that there is no prior knowledge about the system, or that the expert's knowledge is not sufficiently trustworthy. In this case, instead of using a fixed prior interpretation of the system, one often uses raw input-output data to augment one's prior knowledge or perhaps even generate new knowledge about the system. This approach was initially proposed by Takagi-Sugeno-Kang [10] under the name of *TSK fuzzy modeling*. Inspired by classical systems theory, TSK modeling is also referred to as *system identification* [9].

The problem of fuzzy system identification involves eliciting IF_THEN rules from raw input-output data. It usually proceeds in two steps: 1) clustering; and 2) specification of the input-output relations (IF_THEN rules). In this paper, we consider fuzzy clustering as an intuitive approach for generating objective rules in fuzzy modeling. We propose a method for improving the objectivity of this technique by deriving a suitable criterion for selecting the number of clusters for a fuzzy c-mean (FCM) [1][2] clustering process. The FCM clustering algorithm suffers from a major weakness that is usually treated heuristically[7]: it is not obvious how to assign the number of clusters *a priori*. To develop a theoretically well motivated criterion for choosing an appropriate number of clusters, we derive a generalization of Stein's unbiased risk estimator (SURE) [5] that can be used to define a generic criterion which defines the optimum number of clusters to use in a given problem.

In Section 2 of this paper we review fuzzy system identification. We then explain the under-fitting and over-fitting effects caused by using an inappropriate number of clusters for fuzzy system identification in Section 3. In Section 4 we derive a generalization of SURE, and in Section 5 show how it can be applied to fuzzy system identification based on FCM. Experimental results of the proposed criterion and its performance are presented in Section 6.

2 Fuzzy System Identification

Fuzzy system identification normally proceeds through two phases: clustering, and specification of input-output relations (IF_THEN rules). Clustering is a process whereby numerical data is placed into groups or clusters, such that data in a cluster tends to be similar, and data in different clusters tends to be dissimilar. Clustering is normally applied in situations where it is not known *a priori* how many groups are (or should be) present in a given data set. When clustering is applied to a set of input-output data, each cluster center can be considered as a fuzzy rule that describes the characteristic behavior of the system.

Consider a collection of data in an M-dimensional space, where the first N dimensions correspond to input variables, and the remaining $M-N$ dimensions correspond to output variables. Then clustering on this M-dimensional space divides data into fuzzy clusters that overlap with each other, and the membership of each data vector to each cluster can be defined by a membership grade in [0,1]. The data vector with membership grade equal to one is called the cluster center. Suppose that a set of s cluster centers $\{c_1^*, c_2^*, ...c_s^*\}$ has been generated by a clustering method. Each cluster center c_i^* can be decomposed to two vectors x_i^* and y_i^*, such that x_i^* represents the first N dimensions (the coordinates of the cluster centers in input space) and y_i^* represents the last $M-N$ dimensions (the coordinates of the cluster centers in output space).

The membership grade of each data vector is defined as follows:

$$\mu_i(x) = e^{-\alpha||x-x_i^*||^2}$$

where x is the input vector.

Each cluster center c_i corresponds to a fuzzy rule i, and the cluster identified above by the exponential membership function represents the antecedent of this rule. If A_i notifies the exponential membership function of cluster i, then rule i can be represented as:

IF X is $\mathsf{A_i}$ THEN $\mathsf{Y_i}$ is $\mathsf{B_i}$

where X is the vector of input variables, Y_i is the ith output variable and $\mathsf{B_i}$ is a singleton defined as a linear or quadratic combination of input variables.

When B is defined as a linear combination, the model is called the first order model, and when B is a quadratic combination, the model is called the second order model. For the first order model we consider in this study, B is: $\sum_{j=1}^{N} p_{ij}x_j + p_{i0}$, where p_{ij} is the coefficient of x_j in rule i.

Employing traditional fuzzy IF_THEN rules, the first order model would be expressed as follows:

r_1: IF X is $\mathsf{A_1}$ THEN $\mathsf{Y_1(X)} = \sum_{\mathsf{j=1}}^{\mathsf{N}} \mathsf{p_{1j}x_j} + \mathsf{p_{10}}$

$\vdots$

r_s: IF X is $\mathsf{A_s}$ THEN $\mathsf{Y_s(X)} = \sum_{\mathsf{j=1}}^{\mathsf{N}} \mathsf{p_{sj}x_j} + \mathsf{p_{s0}}$

For a given x_o, the output of the model y_0, is computed as:

$$y_0 = \frac{\sum_{i=1}^{s} \mu_i(x_0) Y_i(x_0)}{\sum_{i=1}^{s} \mu_i(x_0)}$$

This equation can be converted into a linear least-squares estimation problem by the following definition:

$$\beta_i = \frac{\mu_i(x_0)}{\sum_{j=1}^{s} \mu_j(x_0)}$$

So

$$y_0 = \sum_{i=1}^{s} \beta_i Y_i(x_0)$$

When there are n data vectors:

$$y_1 = \beta_{11}\Big(\sum_{j=1}^{N} p_{1j}x_j + p_{10}\Big) + \cdots + \beta_{1s}\Big(\sum_{j=1}^{N} p_{sj}x_j + p_{s0}\Big)$$

$$\vdots$$

$$y_n = \beta_{n1}\Big(\sum_{j=1}^{N} p_{1j}x_j + p_{10}\Big) + \cdots + \beta_{ns}\Big(\sum_{j=1}^{N} p_{sj}x_j + p_{s0}\Big)$$

This is a least square estimation problem and can be represented as:

$$Y = AP \qquad (1)$$

where Y is a matrix of output values, A is a constant matrix, and P is a matrix of parameters to be estimated. A necessary condition for $||Y-AP||^2$ to be minimized is that, P be: $P = (A^TA)^{-1}A^TY$

3 Model validation

Fuzzy system identification is a parameter estimation problem. One problem with model validation is selecting parameters that show good performance both on training and testing data. In principle, a model is selected to have parameters associated with the best observed performance.

Not surprisingly, a model selected on the basis of training data does not exhibit as good performance on the testing data. When squared error is used as the performance index, a zero-error model on the training data can always be achieved by using an adequate number of cluster centers. However, training error err and testing error Err do not demonstrate a linear relationship. In particular, a smaller training error does do not necessarily result in a smaller testing error. In practice, one often observes that, up to a certain point, the model error on testing data tends to decrease as the training error decreases. However, if one attempts to decrease the training error too far by increasing model complexity, the testing error often can take a dramatic increase.

The basic reason behind this phenomenon is that in the process of minimizing training error, after a certain point, the model begins to over-fit the training set. Over-fitting in this context means fitting the model to training data at the expense of losing generality. In the extreme form, a set of n training data points can be modeled exactly with n rules. Such a model follows the training data perfectly. However, these rules are not representative features of the true underlying data source, and this is why they fail to correctly model new data points.

In general, the training error rate err will be less than the testing error on the new data, Err. A model typically adapts to the training data, and hence the training error err will be an overly optimistic estimate of the generalization error Err. An obvious way to estimate generalization error is to estimate the degree of optimism OP inherent in a particular estimate, and then add a penalty term to the training error to compensate, i.e., such that $Err = err + OP$. The method described in the next section works in this way.

4 Estimating the optimism

Let:
$MSE(\widehat{f}) = E(\widehat{f} - f)^2$.
$\widehat{f}(X)$: *Prediction model*, estimated from a training sample by the TSK method.
$f(X)$: *Real model.*
err: *Training error*, which is the average loss over the training sample.
Err : *Generalization error*, which is the expected prediction error on test sample.

Recall that the training error, $err = \sum_{i=1}^{n}(\widehat{y} - y)^2$, is an estimate of the expectation of the squared error on the training data, $E(\widehat{y} - y)^2$, while the generalization error (test error) Err is an estimate of mean squared error, $MSE = (\widehat{f} - f)^2$, where $\widehat{f}(X)$ is the estimated model and $f(X)$ is the true model.

Now suppose $y_i = f(x_i) + \varepsilon_i$, where ε is additive Gaussian noise $N(0, \sigma^2)$. We need to estimate $\widehat{f}$ from training data $D = \{(x_i, y_i)\}_i^n$. Consider

$$E[(\widehat{y_0} - y_0)^2] = E[(\widehat{f} - f - \varepsilon)^2]$$

$$= E[(\widehat{f} - f)^2] + E[\varepsilon^2] - 2E[\varepsilon(\widehat{f} - f)] \quad (2)$$

$$= E[(\widehat{f} - f)^2] + \sigma^2 - 2E[\varepsilon(\widehat{f} - f)] \quad (3)$$

Here, the last term can be written as:

$$E[2\varepsilon(\widehat{f} - f)] = 2E[(y_0 - f)(\widehat{f} - f)] \equiv cov(y_0, \widehat{f})$$

We consider two different cases.

Case 1. Consider the case in which a new data point has been introduced to the estimated model, *i.e.*$(x_0, y_0) \notin D$. Since y_0 is a new point, $\widehat{f}$ and y_0 are independent. Therefore $cov(y_0, \widehat{f}) = 0$ and (3) in this case can be written as:

$$E[(\widehat{f} - f)^2] \quad = \quad \sigma^2 - E(\widehat{y_0} - y_0)^2 \quad (4)$$

This is the justification behind the technique of cross validation. In cross validation, to avoid overfitting or underfitting, a validation data set is used which is independent from the estimated model. The optimal model parameters should be selected to have the best performance index associated with this data set. Since this data set is independent from the estimated model, it is a fair estimate of $E(\widehat{f} - f)^2$ and consequently of generalization error Err as indicated in (4).

Case 2. A more interesting case is the case in which we do not use new data points to assess the performance of the estimated model, and the traing data is used for both estimating and assessing a model $\widehat{f}$. In this case the cross term in (3) cannot be ignored because $\widehat{f}$ and y_0 are not independent. Therefore the cross term, which is $cov(y_0, \widehat{f})$, is not zero. However the cross term can be estimated by Stein's lemma [5] [8], which was originally proposed to estimate the mean of a Guassian distribution [8].

According to Stein's lemma if $X \sim N(\theta, \sigma^2)$ and $g(x)$ is differentiable function then $E(g(x)(x - \theta)) =$

$\sigma^2 E(g'(x))$. So we let $g(\varepsilon) = \widehat{f} - f = \widehat{f} - y - \varepsilon$ and $x = \varepsilon$. Then by applying Stein's lemma we obtain

$$E(\varepsilon(\widehat{f} - f)) \quad = \quad \sigma^2 E(g'(\varepsilon)) \quad = \quad \sigma^2 E(\frac{d\widehat{f}}{dy})$$

Summing over all y we get

$$\begin{aligned} Err &= \sum_{i=1}^{n} (\widehat{y} - y)^2 - n\sigma^2 + 2\sigma^2 \sum_{i=1}^{n} \frac{d\widehat{f}(x_i)}{dy_i} \\ &= err - n\sigma^2 + 2\sigma^2 \sum_{i=1}^{n} \frac{d\widehat{f}(x_i)}{dy_i} \end{aligned} \tag{5}$$

This is known as Stein's Unbiased Risk Estimator (SURE).

5 Determining the optimum number of clusters

Based on this criterion, the optimum number of clusters should be assigned to have the minimum generalization error Err in (5). However, in $\widehat{f}(x_i)$ all of the parameters corresponding to rules $Y_i(x_0)$ and to $\mu_i(x_0)$ are functions of y_i. Therefore, taking the derivative of $\widehat{f}(x_i)$ with respect to y_i is not easily accomplished. This difficulty can be resolved by applying SURE to the alternative form shown in (1). From the least squared solution of (1) we have:

$$\begin{aligned} P &= (A^T A)^{-1} A^T Y \\ \widehat{f} &= AP = A(A^T A)^{-1} A^T Y = HY \end{aligned} \tag{6}$$

where H is an $N \times N$ matrix that depends on the input vector x_i but not on y_i. From (6) we can easily obtain the required derivative of $\widehat{f}(x_i)$ with respect to y_i.

$$\sum_{i=1}^{n} \frac{d\widehat{f}(x_i)}{dy_i} = \sum_{i=1}^{n} H_{ii}$$

Now, substituting this into (5) we obtain

$$Err \quad = \quad err - n\sigma^2 + 2\sigma^2 \sum_{i=1}^{n} H_{ii}$$

Here we observe that $\sum_{i=1}^{n} H_{ii} = Trace(H)$, the sum of the diagonal elements of H. Thus, we can obtain the further simplification that $Trace(H) = Trace(A(A^T A)^{-1} A^T) = Trace(A^T A(A^T A)^{-1}) = Trace(I) = P$, where P is the dimension of A. Since A is a projection of input matrix X onto a basis set spanned by C, the number of clusters, and the dimension of the original input X is N, one can show generally $P = C(N + 1)$.

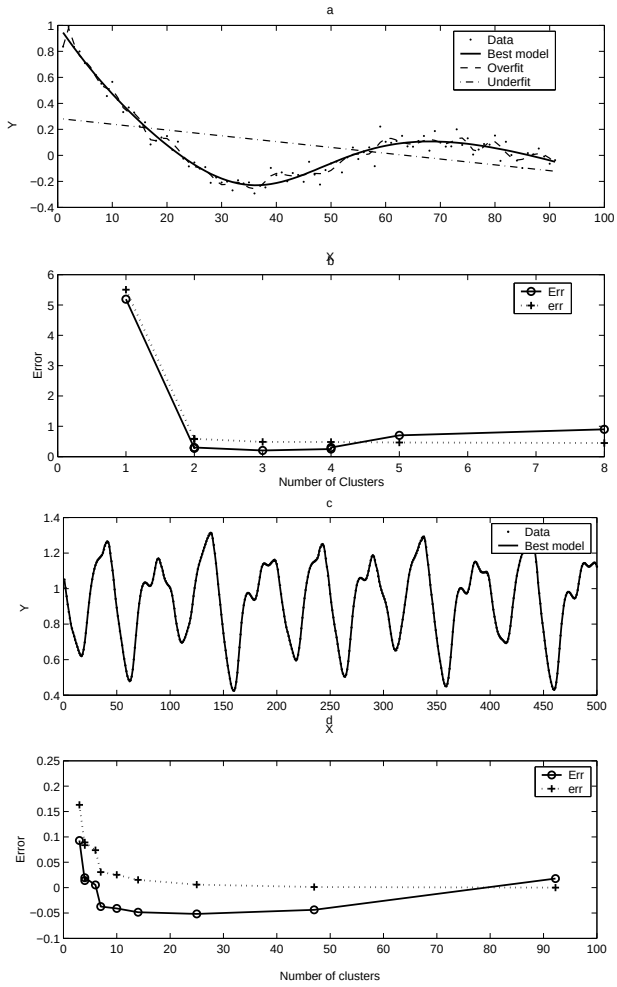

Figure 1: (a) Overfitting, underfitting, and the best estimated model for $y = \frac{sin(x)}{x}$. (b) Err obtained in (7) used to find the optimum number of clusters for model $y = \frac{sin(x)}{x}$. (c) The best estimated model for Mackey-Glass time series. (d) Err otained in (7) used to find the optimum number of clusters for Mackey-Glass time series.

		CV		SURE	
Target function	n	Test error ratio	Cluster diff.	Test error ratio	Cluster diff.
step($x \geq 0.5$)	100	1.027	-0.42	1.017	0.09
$sin(\frac{1}{x})$	100	1.039	-1	1.027	0.06
$sin^2(2\pi x)$	100	1.022	-0.24	1.02	-1.24
step($x \geq 0.5$)	200	1.003	0.05	1.001	0.04
$sin(\frac{1}{x})$	200	1.012	-0.35	1.001	-0.03
$sin^2(2\pi x)$	200	1.004	-0.04	1.008	-0.96

Table 1: Fitting different target functions with $\sigma = 0.25$. Table reports ratio of test errors relative to best possible test error achieved by different methods. A smaller ratio is better. Results are reported at training sample sizes $n = 100$ and $n = 200$ and averaged over 100 repeated trials in each case. Columns 4 and 6 show the difference between the optimum number of clusters and the number of clusters chosen by CV and SURE respectively.

To use this method to find the optimum number of clusters, we simply choose the model that obtains the smallest Err over the set of models considered. Given a set of models $\widehat{f}_C(x)$ indexed by the number of clusters, C, denote the training error for each model by $err(C)$. We then obtain

$$Err(C) \quad = \quad err(C) - n\sigma^2 + 2\sigma^2 C(N+1) \quad (7)$$

where n is the number of training samples and the noise, σ^2, can be estimated from the mean squared error of the model.

6 Experimental results

To explore the effectiveness of our complexity control method, we considered the problem of fitting a first order TSK model to a set of points (Figure 1). The goal is to minimize the squared generalization error Err. To determine the efficacy of the method we compared its performance to the well studied standard cross validation [3].

We first conducted a simple series of experiments by fixing a uniform distribution on the unit interval [0,1], and then fixing various target functions $f : [0, 1] \rightarrow R$. To generate training samples, a sequence of values $x_1, ..., x_t$ is drawn from $[0, 1]$, the target function values $f(x_1), .., f(x_t)$ are computed, and independent Gaussian noise is added to each value. For a given training sample, the series of best fit functions corresponding to a number of clusters C= 1, 2, .. ,*etc.* are computed. Given this sequence, the cross validation strategy will choose some particular model $\widehat{f}_c^*$ on the basis of the observed empirical errors on the validation data set (generated the same way as training data). Our technique will alternatively chose the model corresponding to minimum Err in (7). To determine the effectiveness of these two strategies, the ratio of the test error of the model selected by them to the best test error on a new test data set among the models in sequence $C = 1, 2, ..$ is measured.

Table 1 shows the results obtained for fitting various functions. These results are obtained by repeatedly generating training samples of a fixed size, and recording the ratio of test error achieved relative to the best possible test error, for each technique (CV and SURE).

In another experiment we considered a benchmark problem in system identification: predicting the time series generated by the chaotic Mackey-Glass (MG) differential delay equation [6]. Here the goal is to predict the value of x at some time $t + \Delta t$, using the past values of x up to the time t. We use $\tau = 17$, $N = 4$, $S = 6$, $\Delta t = 6$ and same data set that was used in [4]. In this experiment, our proposed criterion SURE and cross validation CV chose the same number of clusters, and therefore they achieved the same generalization error 0.005 on the test data. However, the SURE technique does not require an extra validation set to choose its cluster number, and in fact used half the data available to CV in this case. Therefore, we claim that it achieved more effective performance on this example. Figure 1.*d* shows a comparison of the training error for each number of clusters and compares this to the Err for each.

7 Conclusion

We have proposed a new approach to choosing the optimum number of clusters for FCM in system identification. Our approach minimizes a theoretically unbiased estimate of generalization error of the model. Our experimental results validate the effectiveness of this approach. A comparison cross validation illustrates that the generalization error of the models selected by our approach is usually less than models se-

lected by cross validation technique. Importantly, this is achieved while requiring much less computation than cross validation. The utility of our method is greatest when there is insufficient data to hold out a validation set for cross validation.

References

[1] Bezdek J.C.: Fuzzy mathematics in pattern classification. Ph.D. dissertation. Cornell University. Ithaca. NY. (1973)

[2] Bezdek J.C.: A convergence theorem for the fuzzy ISODATA clustering algorithms. IEEE Transactions on Pattern Analysis and Machine Intelligence. **PAMI-2(1)** (1980) 1–8

[3] Craven P., Wahba G.: Smoothing noisy data with spline functions. Number. Math **31** (1979) 377–403

[4] Jang JSR: ANFIS: Adaptive-network-based fuzzy inference system. IEEE Trans. On systems. Man & Cybernetics **23(3)** (1993)665–685

[5] Ker-Chau L.: From Stein's Unbiased Risk Estimates to the Method of Generalized Cross Validation. Annals of Statistics. **13(4)** (1985) 1352-1377

[6] Mackey M., Glass L.: Oscillation and chaos in physiological control systems. Sience **197** (1977) 287-289

[7] Pal N.R and Bezdek J.C.: On cluster validity for the fuzzy c-mean model. TFS. **5(1)** (1997) 152–153

[8] Stein M. C.: Estimation of the Mean of a Multivariate Normal Distribution. Annals of Statistics.**9(6)** (1981) 1135–1151

[9] Sugeno M. and Yasukawa T.: A fuzzy-logic-based approach to qualitative modeling. IEEE Trans. Fuzzy Syst. **1** (1993) 7-31

[10] Takagi T. , Sugeno M.: Fuzzy identification of systems and its application to modeling and control. IEEE Trans. Syst. Man & Cybern. **SMC-15(3)** (1985) 116–132