ESANN'2002 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 24-26 April 2002, d-side publi., ISBN 2-930307-02-1, pp. 155-160

Noise derived information criterion for model selection

Joaquín Pizarro Junquera[1], Pedro L. Galindo Riaño[1],Elisa Guerrero Vázquez[1], Andrés Yáñez Escolano[1]

[1] Universidad de Cádiz, Departamento de Lenguajes y Sistemas Informáticos
Grupo de Investigación "*Sistemas Inteligentes de Computación*"
C.A.S.E.M. 11510 - Puerto Real (Cádiz), Spain
{joaquin.pizarro , pedro.galindo , elisa.guerrero , andres.yaniez}@uca.es

Abstract.. This paper proposes a new complexity-penalization model selection strategy derived from the minimum risk principle and the behavior of candidate models under noisy conditions. This strategy seems to be robust in small sample size conditions and tends to AIC criterion as sample size grows up. The simulation study at the end of the paper will show that the proposed criterion is extremely competitive when compared to other state-of-the-art criteria.

Keywords: Model Selection. Minimum Prediction Risk. Network Size.

1. Introduction

Many data modelling problems are characterized by two difficulties: the absence of a complete a priori model of data generation process and the limited quantity of data. When constructing statistical models for such applications, the issues of model selection and estimation of generalization ability or *prediction risk* are crucial and must be addressed in order to construct a near optimal model. Under the simplest formulation of model selection, the idea is to define a set of candidate hypothesis class $H_0 \subset H_1 \subset ... \subset H_n$ and then choose the class which best fits the data. The notion of best fit can be defined via an objective criterion, such as *maximum a posterior probability* (MAP), *minimum Bayesian information criterion* (SIC), *minimum description length* (MDL) or *minimum prediction risk* (P). We focus on the prediction risk as our selection criterion because it tells us how much confidence to put in predictions produced by our best model.

This paper focuses on regression problems. Consider a set of real-valued input/output data pairs X= {(xi, yi) i=1,..,n} generated according to a density function (true model) plus noise $y_i=f(x_i)+\varepsilon_i$, where y_i is the observed response (dependent variable), x_i and ε_i are drawn independently with unknown distribution p(x) and p(ε), and f(x) is an unknown function. We assume a set of hypothesis classes (candidate models) H_m where m is the number of parameters in the models. The true model f(x) may or not may be contained in any of these classes. One wants to find the model $g(x,\theta_k) \in H_k$ which best approximates the data. The quality of an approximation produced by the candidate model is measured by the loss or discrepancy measure $L(y,g(x, ,\theta_k)$ between the output produced by the model $g(x,\theta_k)$ and the true model f(x).

The expected value of the loss is called the prediction risk

$$P(k) = \int L(y, g(x,\theta_k))dp(x)dp(\varepsilon)$$

The prediction risk can not be calculated directly and must be estimated from available data. The most standard method for estimating prediction risk is test set validation, which includes cross-validation and bootstrap, estimates P(k) as

$$P(k) \approx \left\langle P^{est}(k) \right\rangle = \left\langle \frac{1}{N} \sum_{i=1}^{N} L(y^*, g(x^*,\theta_k)) \right\rangle$$

where Z=(y*,x*) are N data pairs generated from the same density function than X which were not used in the estimation of g(x,θ_k). We use angled brackets 〈〉 to denote expected values. As the size of Z increases, the expected value of the empirical risk approaches the prediction risk. However, for many important problems, data is scarce or expensive, making test set validation impractical or impossible. In these situations, one must use alternative approaches that enable the estimation of prediction risk from the empirical risk (also called resubstitution error) defined as follows:

$$P^{emp}(k) = \frac{1}{n} \sum_{i=1}^{n} L(y, g(x,\theta_k))$$

The first idea that comes to mind is to use the empirical risk directly to estimate P(k). However, this idea does not work, because empirical risk typically decreases with model complexity k. Therefore, choosing the function with minimum training error simply leads to choosing a function from the most complex class. There are main-stream statistics literature contains a number of reliable and computationally cheap corrections to the empirical risk to estimate the prediction risk for the case of quadratic loss and gaussian noise N(0,σ_ε) which can be expressed [1][3][5] in the general form

$$\left\langle P^{est}(k) \right\rangle = \left\langle P^{emp}(k) \right\rangle \cdot T(n,k)$$

where n is the sample size, k is the model complexity, and expectations are taken over all possible training and test sets. As shown above, T(n,k) is the correction factor or penalty term to the empirical risk for performing model selection. Taking logs

$$\log\left\langle P^{est}(k) \right\rangle = \log\left\langle P^{emp}(k) \right\rangle + \log T(n,k) = \log\left\langle P^{emp}(k) \right\rangle + C(n,k)$$

This expression may be more usual to researchers, given that most of the information criteria derived in the past, as AIC, AICc, [2] etc., follows a similar scheme. By adopting the bias/variance decomposition perspective, the penalty term can be interpreted as postulating a particular profile for the variances as a function of model complexity. If the postulated and true profile do not mach, then systematic underfitting or overfitting results, depending on whether the penalty terms are too large or too small. Although the empirical risk typically decreases with model complexity k, the prediction risk first drops and then begins to rise. This is because the bias tend to drop with the model complexity, while the variance increases with the model complexity. Given a sequence of models with increasing k, some model of optimal size will minimize the prediction risk.

2. Noise derived information criterion.

The penalty term T(n,k) only depends on the complexity and nature of the candidates models and the training sample size. We are assuming that the true model is contained in the set of candidate models, or is not very different from the nearest candidate models. Its value will be

$$C_k(n,k) = \log \frac{\left\langle P^{est}(k) \right\rangle}{\left\langle P^{emp}(k) \right\rangle}$$

In order to compute this penalty term, we should estimate the prediction risk and the empirical risk. For the case of the quadratic loss, we have

$$C_k(n,k) = \log \frac{\left\langle \frac{1}{N} \sum (y_i^* - g(x_i^*, \theta_k))^2 \right\rangle}{\left\langle \frac{1}{n} \sum (y_i - g(x_i, \theta_k))^2 \right\rangle} = \log \frac{\left\langle \sigma^2_{k,est} \right\rangle}{\left\langle \sigma^2_{k,emp} \right\rangle}$$

This error variance is a suitable functional to measure the goodness of fit of candidate models for the problem of fitting the Y values. The residual sum of squares is the sum of squares of prediction error. The residual mean square is an estimate of this error variance and as such is a suitable model selection criterion for the problem of fitting a regression model to a data set.

If C(k,n) only depends on the complexity and nature of the candidates models, this relation will hold for any true function f(x), then in particular it will hold for the function f(x)=0. In this case, the true function will be nested to candidate models, and these will all be overfitted models. Then

$$C_k(n,k) = \log \frac{\left\langle \frac{1}{N} \sum (\varepsilon_i^* - g(x_i^*, \theta_k))^2 \right\rangle}{\left\langle \frac{1}{n} \sum (\varepsilon_i - g(x_i, \theta_k))^2 \right\rangle} = \log \frac{\sigma_\varepsilon{}^2 \left\langle \frac{1}{N} \sum (\frac{1}{\sigma_\varepsilon^2} \varepsilon_i^* - \frac{1}{\sigma_\varepsilon^2} g(x_i^*, \theta_k))^2 \right\rangle}{\sigma_\varepsilon{}^2 \left\langle \frac{1}{n} \sum (\frac{1}{\sigma_\varepsilon^2} \varepsilon_i - \frac{1}{\sigma_\varepsilon^2} g(x_i, \theta_k))^2 \right\rangle}$$

Then, assuming that candidate models are linear in the parameters it is possible to get:

$$C(n,k) = \log \frac{\left\langle \frac{1}{N} \sum (\varepsilon_i^{*\prime} - g(x_i^*, \theta_k^\prime))^2 \right\rangle}{\left\langle \frac{1}{n} \sum (\varepsilon_i^\prime - g(x_i, \theta_k^\prime))^2 \right\rangle} = \log \frac{\left\langle \sigma'^2_{k,est} \right\rangle}{\left\langle \sigma'^2_{k,emp} \right\rangle}$$

So we are able to compute C(n,k) analysing the behaviour of candidate models under a noise distribution N(0,ε). This is a very important result, because it allows us to determine the penalty term without knowing the true variance of noise. Note that the true noise variance is cancelled out. We may now use a Monte Carlo approach to calculate C(n,k) from estimates of $<\sigma'^2_{k,est}>$ and $<\sigma'^2_{k,emp}>$ using artificially generated noise samples. Different sets of training(size=n) and test(huge) are generated from the gaussian distribution N(0,1), and all the models are fitted. Then $\sigma'^2_{k,est}$ and $\sigma'^2_{k,emp}$ for each sample are computed and the mean is taken as an estimate of $<\sigma'^2_{k,est}>$ and $<\sigma'^2_{k,emp}>$ respectively. Then we have

$$\log\left\langle\sigma^2_{k,est}\right\rangle=\log\left\langle\sigma^2_{k,emp}\right\rangle+C_k(n,k)$$

The maximum likelihood estimates of $\left\langle\sigma^2_{k,emp}\right\rangle$ is $\mathrm{SSE}_{\mathrm{k,emp}}$/n , where SSE is the sum of squared errors. We define the Noise Derived Information Criterion as

$$NDIC_k=\log\frac{SSE_{k,emp}}{n}+C(n,k)$$

If unbiased estimates of $\left\langle\sigma^2_{k,emp}\right\rangle$ is used, the Noise Derived Information Criterion Unbiased follows immediately

$$NDICu_k=\log\frac{SSE_{k,emp}}{n-k}+C(n,k)$$

3. Experimental results

Different and representative problems have been simulated for linear models: polynomial fitting problem f_1(x)=10*sin(3*x+6)+N(0,0.3std(f1)), univariate autoregressive regression problem [2] f_2(x)= x1+x2+x3+x4+x5+x6+N(0,1), (x1=1), and for non linear models: neural network trained to fit a highly no linear function [6]. Different sample sizes were chosen depending on the complexity of the problems emphasizing small samples behaviour. Different columns show the results in ascending complexity, showing how many times each model is selected. In order to compare different methods, we compute how far the model chosen by the selection strategy deteriorates from the true model, calculating the relative difference between the true model variance (σ_{true}=mse(f^{true}-f^{true}+noise)) and the selected model variance by the model selection strategy ($\sigma_{\mathrm{selected}}$=mse($g_{\mathrm{selected}}$-$f^{\mathrm{true}}$+noise)) where f^{true} is the true function, f^{true}+noise is the true function plus gaussian noise, g_{selected} is the function selected by the model selection strategy and mse is the mean squared error function.

$$D=\frac{\sigma_{selected}-\sigma_{true}}{\sigma_{true}}$$

The lower the difference, the better the generalization capabilities of the model. This value is shown in the last row of the tables. The last column (GE) represents those models selected by the minimum generalization error criterion computed over a huge randomly test set. We are able to compute these values because we know the true model and the true noise distribution The values given in the tables are the averages over 1000 replications of simulation study for linear models and 100 for non linear models.

Table 1. Polynomial fitting to a sinusoidal function f_1 k = polynomial orders = 1..10

	SIZE=15							SIZE=20						
k	NDIC	NDICu	AIC	AICc	SIC	AICu	GE	NDIC	NDICu	AIC	AICc	SIC	AICu	GE
1	0	4	0	23	0	110	2	0	0	0	1	0	3	1
2	0	3	0	87	0	11	1	0	0	0	0	0	0	0
3	3	12	0	20	1	36	6	0	0	0	0	0	1	1
4	83	109	10	108	15	116	11	7	15	1	15	3	21	2
5	912	870	94	695	159	640	232	474	568	79	451	15	571	106
6	2	2	83	114	117	74	212	402	340	114	248	16	214	122
7	0	0	161	31	185	13	375	117	77	262	264	330	178	492
8	0	0	119	1	113	0	117	0	0	137	15	113	11	167
9	0	0	177	0	149	0	35	0	0	153	6	98	1	68
10	0	0	356	0	261	0	9	0	0	254	0	142	0	41
D	**2.614**	**2.830**	**4.3e6**	**4.098**	**4.2e6**	**5.308**	**0.912**	**1.676**	**1.691**	**1.7e3**	**2.438**	**1.6e3**	**2.193**	**0.563**
	SIZE=100							SIZE=500						
D	**0.108**	**0.107**	**0.109**	**0.108**	**0.106**	**0.107**	**0.088**	**0.0059**	**0.0062**	**0.0059**	**0.0059**	**0.0087**	**0.0062**	**0.0053**

Table 2. Univ. autoregressive fitting to f2, k=#vars = 1..12

	SIZE=15							SIZE=20						
k	NDIC	NDICu	AIC	AICc	SIC	AICu	GE	NDIC	NDICu	AIC	AICc	SIC	AICu	GE
1	12	49	0	49	4	187	0	0	5	0	3	0	13	0
2	19	67	0	67	1	133	0	1	6	0	2	1	11	0
3	32	67	1	78	6	99	0	6	15	0	7	1	23	0
4	68	101	1	101	5	104	3	13	31	2	22	4	33	0
5	163	162	10	160	21	129	32	51	70	6	58	17	80	12
6	676	542	173	531	276	340	662	847	834	349	845	568	810	702
7	23	12	42	14	64	8	165	66	35	79	51	86	28	146
8	7	0	64	0	57	0	83	10	4	73	7	57	2	78
9	0	0	52	0	56	0	35	5	0	87	4	52	0	31
10	0	0	81	0	72	0	11	1	0	88	1	66	0	16
11	0	0	151	0	118	0	5	0	0	102	0	52	0	13
12	0	0	425	0	320	0	4	0	0	214	0	96	0	2
D	**0.759**	**1.073**	**3.146**	**1.088**	**2.510**	**1.693**	**0.396**	**0.382**	**0.439**	**0.846**	**0.403**	**0.640**	**0.502**	**0.274**
	SIZE=25							SIZE=100						
D	**0.268**	**0.266**	**0.472**	**0.269**	**0.349**	**0.278**	**0.217**	**0.062**	**0.053**	**0.069**	**0.062**	**0.050**	**0.054**	**0.046**

For non linear models, networks are trained by ordinary least-squares using Levenberg-Marquardt algorithm. For a network with H hidden units, the weights from the previously trained network were used to initialise H-1 of the hidden units, while the weights for the Hth hidden unit were generated from a pseudorandom normal distribution. This sequential network construction method creates a nested set of networks having a monotonously decreasing training error and provides some continuity in the model space which makes a prediction risk minimum more easily noticeable [4]. To compute our penalty term, in order to avoid outliers, the median was used instead of the mean. An advantage of the median over the mean, is that it is less susceptible to the effects of outliers, and is thus more likely to be close to the expected value for skewed distributions.

Table 3. Non linear fitting Heavisine function (medium), k = hidden units

	SIZE=50						SIZE=100						SIZE=250					
K	NDIC	NDICu	AIC	AICc	SIC	GE	NDIC	NDICu	AIC	AICc	SIC	GE	NDIC	NDICu	AIC	AICc	SIC	GE
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
3	29	38	24	38	38	21	29	32	24	30	33	11	17	26	13	15	30	2
4	31	42	24	42	40	26	38	43	32	35	41	28	37	46	34	37	51	9
5	29	18	26	19	19	23	20	21	18	22	23	25	17	15	14	16	14	13
6	9	2	10	1	3	14	12	3	13	12	2	16	9	8	13	13	4	13
7	2	0	3	0	0	5	1	1	4	1	1	11	5	2	5	4	1	10
8	0	0	3	0	0	2	0	0	4	0	0	2	6	3	8	6	0	8
9	0	0	5	0	0	3	0	0	3	0	0	2	5	0	3	5	0	9
10	0	0	1	0	0	1	0	0	1	0	0	1	0	0	3	0	0	8
11	0	0	2	0	0	1	0	0	1	0	0	3	3	0	4	3	0	14
12	0	0	2	0	0	3	0	0	0	0	0	1	1	0	3	1	0	14
D	**0.405**	**0.435**	**1.698**	**0.434**	**0.432**	**0.307**	**0.123**	**0.128**	**0.149**	**0.125**	**0.127**	**0.108**	**0.078**	**0.074**	**0.079**	**0.077**	**0.076**	**0.052**

We can see from tables 1, 2 and 3 that AIC and SIC leads to small-sample overfitting problems The corrected versions, AICc and AICu, outperform their parent criterion AIC from an overfitting perspective. In general, model selection methods with strong penalty function (NDIC, NDICu, AICc, AICu) perform better, we mean, they provide better generalization capabilities, than those with weak penalty functions (AIC, SIC). NDIC provides the best results with small sample sizes. This is a very important characteristic, because of the practical limitations on gathering and using data in real-world situations. It is observed that NDIC and NDICu underfit less than AICc and AICu in small sample sizes. If the size of the sample is large enough, all the methods provide similar result. NDIC penalty term tends to AIC with large samples sizes. NDICu, AICc and AICu provide similar result with enough and large samples sizes thus concluding that, in general, NDICu is an extremely competitive model selection penalty function, independently of sample size. When the complexity of the model is difficult to determine, NDIC may be used instead.

4. References

1. Chapelle, O. Model Selection for Small Sample Regression. Machine Leaning 2001. In Press.
2. McQuarrie, A. Tsai C. L. Regression and times series model selection. World Sientific. Singapore. New Jersey. London. Hong Kong.
3. Moody, J. (1992), The effective number of parameters: An analysis of generalization and regularization in nonlinear learning systems, in J. Moody et Al. eds, `Advances in Neural Information Processing Systems', Vol. 4, Morgan Kaufmann, pp. 847-854.
4. Moody, J. 1994. Prediction Risk and Architecture Selection for Neural Networks. From Statistic to Neural Networks: Theory and Pattern Recognition Applications. V.Cherkassky, J.H. Friedman & H. Wechsler Eds. NATO ASI Series F, Springer.
5. Sclove, S. L. Small-sample and large-sample statistical models selection criteria. Selecting models from data: Artificial Intelligence and Stat. IV (Lect. Notes in Statistics Nº89), pp 31-39. P. Cheeseman and R.W. Oldford (eds), Springer 1994.
6. Warren. S. Sarle. Donoho-Johnstone Benchmarks: Neural Net Results. ftp://ftp.sas.com/ pub/neural/dojo/dojo.html.