SQUARE ROOT METHOD OF INFORMATION FILTERING

Ibragimov Begovot Sheralievich
Ibragimov Islomnur
Karshi Engineering and Economic Institute

ABSTRACT

The article discusses the performance characteristics of an adaptive filter using a recursive least squares algorithm, as well as an algorithm using a sliding window.

Keywords: Adaptive algorithm, dispersion, adaptive filter, asymptote, simulation, signal, equations, effective window length.

When the adaptive algorithm works with infinite memory $(\lambda = 1)$, the variance of the filter coefficients will decrease in inverse proportion to the number of data samples. In the final analysis, for all practical purposes, the coefficients can be considered constant. In the case where $\lambda < 1$ the coefficients will always include a noise component with a dispersion that depends on the effective length of the τ window . The variance of the filter coefficients will be constrained from below by the value $(\sigma_\varepsilon{}^2/\tau)R^{-1}$. When the window is of sufficient length, the variance of the filter coefficients will be quite satisfactorily predicted by the asymptotic boundary. For short windows, the true variance can be much larger than its predicted cutoff value [1-4]. To date, there are no analytical results for determining the window length above which the asymptotic estimate is valid. Simulation studies are needed to answer this question. A useful criterion for the performance of an adaptive filter is the dispersion of the output signal. The output signal of a filter with an arbitrary coefficient vector is H given by the formula

$$e_H(n) = y(n) - \varphi^T(n)H\ .$$

The output can always be decomposed into the sum of the "optimal" prediction error and another term:

$$e_H(n) = \underbrace{y(n) - \varphi^T(n)H_{opt}}_{\varepsilon(n)} + \varphi^T(n)\left[H_{opt} - H\right]. \tag{1}$$

Hence, it follows

$$\begin{gathered} E\left\{e_H{}^2(n)\right\} = E\left\{e^2(n)\right\} + E\left\{(H - H_{opt})^T\varphi(n)\varphi^2(n)(H - H_{opt})\right\} + \\ + 2E\left\{\varepsilon(n)\varphi^T(n)\left[H_{opt} - H\right]\right\}. \end{gathered} \tag{2}$$

The last term is zero because the prediction error is not correlated with previous data samples. Thus

$$E\left\{e_H{}^2(n)\right\} = \sigma_\varepsilon{}^2 + (H - H_{opt})^T R(H - H_{opt}). \tag{3}$$

Equation (3) can be used to estimate the dispersion of the output signal at any fixed vector of coefficients H. As expected, the output variance will be minimized if and only if $H = H_{opt}$. The minimum variance ${\sigma_\varepsilon}^2$ can be estimated as follows:

$$\begin{aligned} E\left\{\varepsilon^2(n)\right\} &= E\left\{\left[y(n) + \varphi^T(n)H_{opt}\right]^2\right\} = \\ &= E\left\{y^2(n)\right\} + E\left\{H^T{}_{opt}\varphi(n)\varphi^T(n)H_{opt}\right\} - \\ &- 2E\left\{H^T{}_{opt}\varphi(n)y(n)\right\} = r(0) + H^T{}_{opt}RH_{opt} + 2H^T{}_{opt}r \end{aligned} \tag{4}$$

Using the definition , H_{opt} we get

$${\sigma_\varepsilon}^2 = r(0) - H^T{}_{opt}r = (0) - r^T R^{-1} r\,. \tag{5}$$

It should be noted that during the execution of the adaptive algorithm, the filter coefficients themselves are included in the expressions as random variables. Then, to estimate the output variance, we need to use (2), replacing it H with $\overline{H}(n-1)$:

$$\begin{aligned} E\left\{e^2{}_{H(n-1)}(n)\right\} &= {\sigma_\varepsilon}^2 + E\left\{\overline{H}^T(n-1)\varphi(n)\varphi^T(n)\overline{H}(n-1)\right\} = \\ &= {\sigma_\varepsilon}^2 + trE\left\{\left[\varphi(n)\varphi^T(n)\right]\left[\overline{H}(n-1)\overline{H}^T(n-1)\right]\right\} = \\ &= {\sigma_\varepsilon}^2 + tr\left\{E\left[\varphi(n)\varphi^T(n)\right]E\left[\overline{H}(n-1)\overline{H}^T(n-1)\right]\right\} \approx \\ &\approx {\sigma_\varepsilon}^2 + tr\left\{R\frac{{\sigma_\varepsilon}^2}{n}R^{-1}\right\} = {\sigma_\varepsilon}^2 + \frac{N{\sigma_\varepsilon}^2}{n}. \end{aligned} \tag{6}$$

In H φ other words, the increase in the dispersion of the output signal due to the "noise properties" of the coefficients is determined by the formula

$$M = \frac{E\left\{e^2{}_{\overline{H}(n-1)}(n)\right\} - {\sigma_\varepsilon}^2}{{\sigma_\varepsilon}^2} = \frac{N}{n}. \tag{7}$$

In the signal processing literature, this ratio is sometimes referred to as the "detuning factor" of an adaptive filter. The above result was derived for the case of infinite memory. $(\lambda = 1)$ In the case of finite memory, (7) should be used, replaced n by the effective length of the window $\tau \approx 1/(1-\lambda)$ and treated as the lower bound of the true mismatch coefficient.

The main part of the recursive least-squares (RNA) algorithm is the computation of the gain vector, which includes the correction of the covariance error matrix. $K(n)$ Direct correction $P(n)$ with a difference equation can lead to computational difficulties. The main source of difficulty is the fact that the correction formula does not guarantee positive certainty (inverse) of the covariance matrix $P(n)$ $P(n)$. This obstacle can be circumvented by deriving the correction formula for the square root $P^{1/2}(n)$ of the covariance matrix, defining it as any matrix that has the following properties: $(P^{1/2}(n))(P^{1/2}(n))^T = P(n)$.

The use of square root in the least-squares estimation algorithm has been widely studied in the literature [3-6]. Table 1 shows the square root algorithm for correction and $P(n)$ amplification $K(n)$. Matrices $\overline{U}$ and $-$ diagonal matrices, and$U$

$$P(n-1)=\overline{U}\overline{D}\overline{U}^{T} \quad \text{(old covariance matrix)},$$
$$P(n)=\overline{U}\overline{D}\overline{U}^{T} \quad \text{(new covariance matrix)}. \qquad (8)$$

The computational complexity of this square root algorithm is comparable to that of the standard RNA algorithm.

Table 1. RNA Algorithm Using Square Root

- Beginning
  $\overline{U}=1. \quad \overline{D}=diag[\sigma]$
- First stage
  $f=\tilde{U}^{t}\varphi(n)$, Where is $f=[f_1 \dots f_N]$, $v=\tilde{D}i\ (т.е.\ v_i=\tilde{d}_i f_i\ для\ i=1 \dots\dots N)$
- Main Contour

to $j=2,\dots\dots,N$ compute

$\alpha_j=\alpha_{j-1}+v_f f_f,$

$d_j=\dfrac{\tilde{d}_j\alpha_{j-1}}{\alpha_j\lambda(n)},$

$u_i=\overline{\alpha}_i+\alpha_i k_{i-1}, \quad где \quad \alpha_i=\dfrac{-f_i}{\alpha_{j-1}},$

$k_i=\bar{k}_{i-1}+v_i\tilde{u}_i\,/$

- Notation is used $\tilde{u}_i \quad u \quad u_i$ for columns $\tilde{U}_i, U_i$, i.e.

  $\tilde{U}=[\tilde{u}_i \dots \tilde{u}_N], \quad U=[u_i \dots u_N].$

  $K(n)=\dfrac{\bar{k}_N}{a_N}.$

An essential feature of the exponentially weighted RNA algorithm is that the estimation is instantaneous and based on the values of all previous samples, with the later data samples being taken with greater weight than the earlier ones. In some adaptive applications, it is desirable that the estimate $H(n)$ depend only on a finite number of previous data samples. This can be achieved by using a variant of the covariance form, in which the estimate is a function of the data in a rectangular window L length : . $\{y(n-L+1),\dots,y(n)\}$The output of the recursive algorithm for correction $H(n)$ is carried out according to the same method as without entering a window.

An algorithm that uses a sliding window consists of two separate steps. First, a new selection of data is added. $y(n)$ The old data sample is then discarded, $y(n-L)$but the number of active points is retained, which is equal L to . The first step is as follows:

$$P^1(n) = P(n-1) - P(n-1)\varphi(n)[1+\varphi^T(n)P(n-1)\varphi(n)]^{-1}\varphi^T(n)P(n-1),$$
$$H^1(n) = H(n-1) + K(n)[y(n) - \varphi^T(n)P(n-1)], \tag{9}$$
$$K(n) = P(n-1)\varphi(n)[1+\varphi^T(n)P(n-1)\varphi(n)]^{-1},$$

where $P^1(n)$ and $N^1(n)$ are the values based on $(L+1)$ the data samples.
Now let's discard the data sampling $y(n-L)$:

$$P(n) = P^1(n) + P^1(n)\varphi(n-L)[1+\varphi^T(n-L)P^1(n)\varphi(n-L)]^{-1}\varphi^T(n-L)P^1(n),$$

$$H(n) = H^1(n) - K^1(n)[y(n-L) - \varphi^T(y-L)H^1(n)], \tag{10}$$

$$K^1(n) = P^1(n)\varphi(n-L)[1+\varphi^T(n-L)P^1(n)\varphi(n-L)]^{-1}.$$

This algorithm discards old data samples entirely, and proves to be more efficient than the exponentially weighted option for tracking rapid changes in the statistical parameters of the data $y(n)$. In addition, "bad" data samples [i.e., very large values $y(n)$ caused by the noise pulse] are completely "forgotten" after L the signal sampling periods, whereas in the case of the exponentially weighted variant, this effect can last for a long time.
The previous algorithm required the accumulation of past $(L+N)$ data samples. With a large window size, this can become difficult. In addition, the computational complexity of the algorithm is almost 2 times higher than the previous one. For these reasons, the exponentially weighted form of the RNA algorithm is often preferred in practice.

REFERENCES

1. Kostin V.A. Teoriya upravleniya [Control Theory]. Moscow, Gardariki Publ., 2004. - 224 p. (in Russian).
2. Kahaner D, Moler K, Nash S. Numerical methods and software. Moscow, Mir Publ., 2001, 575 p. (in Russian).
3. Kureychik V.M., LEBEDEV B.K., Lebedev O.K. Exploratory Adaptation: Theory and Practice. MOSCOW, Vysshaya shkola PUBL., 2006. 272 P. (in Russian).
4. Kuznetsov E.S. Upravlenie tekhnicheskikh sistemy: Uchebnoe posobie [Management of technical systems: Textbook] / MADI (TU) – M.: 2001. 262 p. (in Russian).
5. Dorf R., Bishop R. Sovremenniye systems upravleniya. -M: Yunimediastayl. 2002, 822s.
6. Bekkulov J., Ibragimov B., Eshonkulov M. MATHEMATICAL MODEL OF THE TRAJECTORY OF MOVING CONTROL OBJECTS. – 2021. P. 110-116.
7. Ibragimov B. Sh. et al. Osnovnye kharakteristiki discretenykh tekhnologicheskikh protsessov massom proizvodstva kak obektiv avtomatizatsii [Main characteristics of discrete technological processes of mass production as objects of automation]. – 2021. – T. 2. – №. 5. P. 42-47.
8. Kurnasov E. V., Ten V. E. Algorithms of Analysis and Control of the Effectiveness of Automated Technological Processes // Automation and Modern Technologies. – 2014. – No.

12. – S. 21-25.

9. Sheralievich, Ibragimov Begovot, Eshkobilov Sirojiddin Berdimurodovich. "BASIC PRINCIPLES OF AUTOMATION OF HEATING REGULATION OF PROTECTED GROUND STRUCTURES". Journal of Analysis and Invention ResearchJet 2.05 (2021): 426-432.
10. Islamnur, I. (2021, April). Implementation of temperature adjustment in the oven working zone with infinite adjustment. In Archive of Conferences (Vol. 20, No. 1, pp. 94-96).
11. Islamnur, I., Murodjon, O., Sherobod, K., & Dilshod, E. (2021, April). Mathematical account of an independent adjuster operator in accordance with unlimited logical principles of automatic pressure control system in the oven working zone. In Archive of Conferences (Vol. 20, No. 1, pp. 85-89).
12. Islamnur, I., Ogli, F. S. U., Turaevich, S. T., & Sherobod, K. (2021, April). The importance and modern status of automation of the fuel burning process in gas burning furnaces. In Archive of Conferences (Vol. 19, No. 1, pp. 23-25).
13. Nurullayevich, K. S., & Islomnur, I. (2022). SELF-REGULATORY CONTROL SYSTEM FOR FURNITURE FIRE. Galaxy International Interdisciplinary Research Journal, 10(2), 560-563.