In Proc. of Computer Security Symposium'99, pp.81-86, Oct. 1999.

Effects of Data Hiding on Remote Data Analysis

Kanta Matsuura

Institute of Industrial Science, the University of Tokyo

Roppongi 7-22-1, Minato-ku, 106-8558 JAPAN

Abstract: Let us suppose digital data distributed over the network do not exactly keep their original bit-string: digitally-watermarked images, for example. Usually, the difference between the original one and the watermarked one is evaluated by human recognition. If a remote computer analyzes the digital data, however, the evaluation criteria might be different. This paper studies how data-hiding effects on remote data analysis. Specifically, a design commitment competition is simulated. The result suggests the importance of how to represent the design problem; smaller condition numbers provide better robustness of the competition when the SNR is sufficiently high.

1 Introduction

Once equipped with security mechanisms, open networks can be used for wider range of applications. Let us suppose an application where digital data distributed over the network do not exactly keep their original bit-string like digitally-watermarked images[1],[2]. In this example, the difference between the original one and the watermarked one is usually evaluated by human recognition; if the user does not notice the difference, the watermarking is acceptable, *i.e.*, he/she regards it as good as the original one.

In different applications where not a human user but a computer itself uses the digital data, however, the evaluation criteria might also be different. For instance, we may watermark our own measured data and then commit the analysis of the data to a remotely-located third party (see Fig. 1). What if the watermarked data give us results quite different from those given by the original data? We may also watermark our own design criteria and then commit the specific design to a third party. What if the watermarked criteria give us a different solution? In accordance with the application, we should use an appropriate strategy to evaluate the effects of watermarking or data-hiding.

This paper studies the effects of data-hiding on remote data analysis. The basic procedures are introduced in Section 2. Section 3 then shows a simulation by using those procedures. After a discussion in Section 4, Section 5 gives concluding remarks.

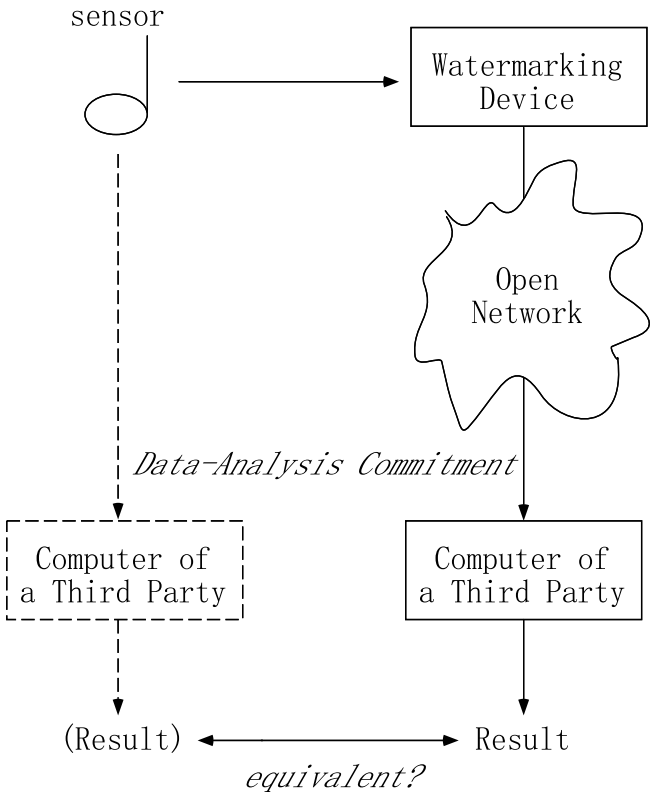


Figure 1: Committed analysis of remotely-measured and then watermarked data.

2 Basic Procedures

2.1 Random Data Hiding

Let $\boldsymbol{y} = (y_1, y_2, \cdots, y_N)^T$ be a set of measurements or design criteria. Not devoted to a specific data-hiding algorithm, we consider the following procedures:

1. Randomly choose a specified number of the measurements $(y_{j_1}, y_{j_2}, \cdots, y_{j_n})^T$.
2. The chosen measurements are disturbed by white noise.

In the following, n/N is referred to as *density* of this data-hiding. Denoting the disturbed data by $\tilde{\boldsymbol{y}}$, SNR (Signal to Noise Ratio) is defined as

$$SNR = 20 \log \frac{\|\boldsymbol{y}\|}{\|\tilde{\boldsymbol{y}} - \boldsymbol{y}\|}. \tag{1}$$

2.2 Generalized Inversion

Let us consider a system equation

$$A\boldsymbol{x} = \boldsymbol{y} \tag{2}$$

where $A = (a_{i,j})$ is a system matrix and $\boldsymbol{x} = (x_1, x_2, \cdots, x_M)^T$ is a set of parameters to be designed by a third party. For simplicity, we assume that A is full-ranked.

When the dimension of $\boldsymbol{x}$ is smaller than that of $\boldsymbol{y}$ (*i.e.*, $M < N$), the system equation (2) can not be completely satisfied. In other words, the system is overdetermined. In this overdetermined case, a generalized-inverse (or pseudoinverse) matrix

$$A^+ \equiv \left(A^T A\right)^{-1} A^T \tag{3}$$

of A is used to obtain a minimum square-error estimate

$$\boldsymbol{x}^+ = A^+ \boldsymbol{y}. \tag{4}$$

When $M = N$, the system equation (2) has a unique solution $\boldsymbol{x}{=}A^{-1}\boldsymbol{y}$. In other words, the system is determined. Depending on the design constraint, however, the third party encounters an overdetermined problem even if $M = N$; if the client wants to reduce the number of parts, the number of non-zero parameters may be limited[1]. Let us suppose that only $m(< M)$ parameters are allowed to be non-zero. In this case, the third party would search for $(i_1, i_2, \cdots, i_m)$ which minimizes the error

$$\left\| \begin{pmatrix} a_{1,i_1} & a_{1,i_2} & \cdots & a_{1,i_m} \\ a_{2,i_1} & a_{2,i_2} & \cdots & a_{2,i_m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N,i_1} & a_{N,i_2} & \cdots & a_{N,i_m} \end{pmatrix} \begin{pmatrix} x_{i_1} \\ x_{i_2} \\ \vdots \\ x_{i_m} \end{pmatrix} - \boldsymbol{y} \right\|. \tag{5}$$

The coefficient matrix in (5) is represented by $A_{(i_1, i_2, \cdots, i_m)}$, in the following. An exhaustive search would be

[1] In designing beamforming arrays, for example, the number of arrays is saved as the solution gets more sparse[3].

1. Choose $(i_1, i_2, \cdots, i_m)$ and set the error $E = \infty$.
2. Compute a minimum-square error solution for $(i_1, i_2, \cdots, i_m)$ as $\boldsymbol{x}^+_{(i_1,i_2,\cdots,i_m)}$= (0, 0, $\cdots$, 0, x_{i_1}, 0, $\cdots$, 0, x_{i_2}, 0, $\cdots$, 0, x_{i_m}, 0, $\cdots$, 0)^T where $(x_{i_1}, x_{i_2}, \cdots, x_{i_m})^T = A^+_{(i_1,i_2,\cdots,i_m)}\boldsymbol{y}$.
3. If $\left\|A\boldsymbol{x}^+_{(i_1,i_2,\cdots,i_m)} - \boldsymbol{y}\right\| < E$, then take $\boldsymbol{x}^+_{(i_1,i_2,\cdots,i_m)}$ as a temporary solution and set $E = \left\|A\boldsymbol{x}^+_{(i_1,i_2,\cdots,i_m)} - \boldsymbol{y}\right\|$.
4. Change $(i_1, i_2, \cdots, i_m)$ and return to 2.

3 Effects of Data Hiding

3.1 Situation

In order to avoid disturbance by computation error, a small scale situation $M = N = 10$ is considered. A and $\boldsymbol{y}$ are randomly generated. $\boldsymbol{y}$ is then watermarked to be $\tilde{\boldsymbol{y}}$ with a density of 0.4. A client commits the design to 10 third parties. These parties are called servers, in the following. The servers are allowed to have at most $m = 3$ non-zero design parameters and search for a better solution by the random search. Each random search is iterated 10 times. Among the results from the servers, the client finds the best one in terms of the square error. The server which gives this minimum error is called a *winner*.

For the final parameter selection by each server, the error not for $\tilde{\boldsymbol{y}}$ but for $\boldsymbol{y}$ is computed. If the winner is still best in terms of this error, the data-hiding is regarded as *acceptable*.

3.2 Results

For each set of A and $\boldsymbol{y}$, the simulation is carried out $K = 100$ times. Let the number of acceptance (= how many times the data-hiding is accepted) be L. The rate of acceptance L/K is then given as in Fig. 2. The simulation were carried out by using two different system matrices whose condition numbers[2] were different by far; one is 1530.4 while the other is 11.97. Since we consider random matrices of small size (10×10), we assume that computation of condition numbers is precise enough for the discussion in the next section.

4 Discussion

In linear estimation theory, the condition number of a system matrix is in close relation to the robustness against errors [4]; larger condition numbers cause larger disturbance or error distribution.

The results in Fig. 2 suggests that

(I) for high SNRs, smaller condition number is better

[2] A condition number of a matrix is defined as the ratio of the largest to the smallest singular values of the matrix.

and that

(II) for low SNRs, the difference in condition number has insignificant effects.

In the simulated situation, a threshold SNR between (I) and (II) is estimated to be around 18dB.

The condition number depends on how the design problem is represented. When data-hiding algorithms are efficient enough to yield SNRs much higher than the threshold, the particular representation of the problem is important for reliable analysis commitment.

5 Concluding Remarks

Effects of data-hiding were analyzed in a context of remote data analysis. Specifically, a design commitment competition was simulated. The result suggests the importance of the representation of design problem; better-conditioned system matrices contribute to more robust competition if the SNR is high enough.

This work is in its infancy and just studied basic characteristics. Quantitative analyses will use not random setting of the problem but particular one such as image processing.

Acknowledgments: This work is supported by the 16th Research-Promotion Fund (No. 28) from Casio Science Promotion Foundation.

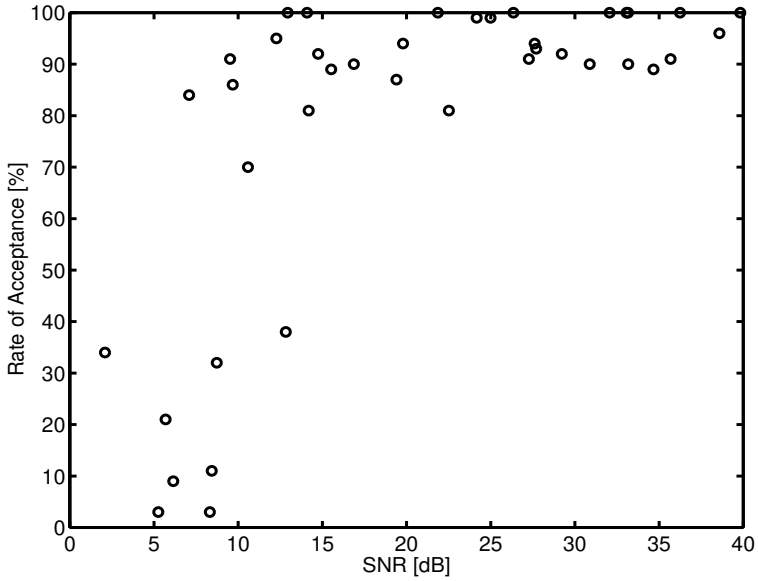
References

[1] J. C. Davis. "Protecting Intellectual Property in Cyberspace". *IEEE Technology and Society Magazine*, Vol. 17, No. 2, pp. 12–25, 1998.

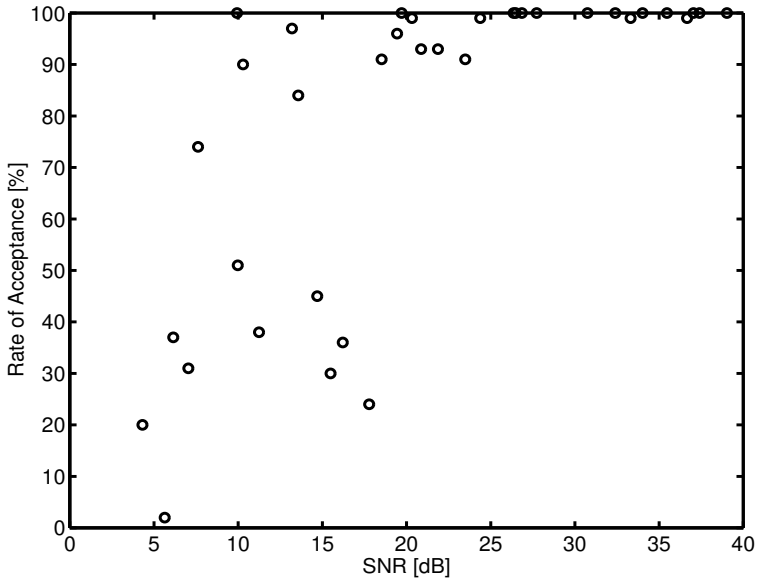
[2] N. Morimoto. "Digital Watermarking Technology". *Journal of IEICE*, Vol. 82, No. 8, pp. 836–838, August 1999. (in Japanese).

[3] R. M. Leahy and B. D. Jeffs. "On the Design of Maximally Sparse Beamforming Arrays". *IEEE Trans. Antennas Propagat.*, Vol. 39, No. 8, pp. 1178–1187, August 1991.

[4] M. J. R. Healy. *Matrices for Statistics*. Oxford University Press, New York, 1986.



(a) $cond(A)$=1530.4



(b) $cond(A)$=11.97

Figure 2: Rate of acceptance vs. SNR. $cond(A)$ represents the condition number of system matrix A.