INFORMATICA, 1998, Vol. 9, No. 3, 315–324



# Simply Invertible Matrices and Fast Prediction

Valdas DIČIŪNAS
*Institute of Mathematics and Informatics, Vilnius University, Department of Informatics*
*Akademijos 4, 2600 Vilnius, Lithuania*
*e-mail: valdas.diciunas@maf.vu.lt*

Received: May 1998

**Abstract.** We study invertibility of big $n \times n$ matrices. There exists a number of algorithms, especially in mathematical statistics and numerical mathematics, requiring to invert step by step large matrices which are closely related to each other. Standard inverting methods require $\mathrm{O}(n^3)$ arithmetical operations therefore using of these algorithms for big values of $n$ becomes problematic. In this paper we introduce some classes of matrices that can be inverted by $\mathrm{O}(n^2)$ operations if we use inverse matrices of other closely related matrices. The most important among them are matrices having big common submatrix and modified sample covariance matrices. We apply our theoretical results constructing a fast algorithm for prediction. This algorithm demonstrates the advantage of our inverting methods and can be used, for example, for safety control in the plant.

**Key words:** inverse matrices, complexity of algorithms, linear regression, real-time prediction, safety control.

## 1. Introduction

There exist a number of algorithms, especially in mathematical statistics (discriminant analysis, regression analysis, prediction, etc.) and numerical mathematics requiring to invert step by step many big matrices. It is well known (Aho *et al.*, 1974) that the problem of inverting of $n \times n$ matrix and the problem of multiplication of two $n \times n$ matrices are asymptotically equivalent and both require $\Omega(n^3)$ arithmetical operations if one uses standard methods. By this reason, using of some statistical algorithms for big values of $n$ becomes problematic. Beginning from pioneering work of Strassen (1969) a number of asymptotically faster methods for multiplication (inversion) of matrices were developed. These methods use $\mathrm{O}(n^\alpha)$ arithmetical operations where $\alpha$ decreases from $\log_2 7 \approx 2.81$ in (Strassen, 1969) to $2.38$ in (Coppersmith and Winograd, 1986). Unfortunately, these nonstandard methods are very complicated and the constants hidden in notation "O" are astronomically large. Therefore, with the only possible exception maid for Strassen's method, the remaining methods have more theoretical than practical significance.

On the other hand, the matrices used in the statistical algorithms mentioned above are very often closely related to each other. They usually have a big common submatrix or are obtained from one another by small modifications. It appears that if we already have one inverse matrix then modified matrices can be inverted in such cases more simply, that is using additionally only $\mathrm{O}(n^2)$ arithmetical operations. Well-known formula of Bartlett

(1952) is a good illustration of such situation. There also exist a lot of matrices that are simply invertible due to their special structures.

An objective of this paper is to investigate some classes of simply invertible matrices and to propose useful analytic solutions. The paper is organised as follows. Section 2 is allocated for definitions and examples. In Section 3 some new cases of simply invertible matrices usefull in statistics and its applications are proposed. In particular, matrices having big common submatrix and so-called modified sample covariance matrices are considered. In Section 4 we apply these new theoretical results to construct an extremely fast algorithm for real-time prediction. Finally, in Section 5 the summary of this paper is given.

## 2. Definitions and Examples

We study matrices whose elements are real numbers. Let $\mathbf{B}$ be an $m \times n$ matrix and $\mathbf{A}_1, \ldots, \mathbf{A}_k$ be matrices of arbitrary dimensions. We call matrix $\mathbf{B}$ *simply computable with respect to* $\mathbf{A}_1, \ldots, \mathbf{A}_k$ if $\mathbf{B}$ may be computed using $\mathbf{A}_1, \ldots, \mathbf{A}_k$ by $\mathrm{O}(mn)$ arithmetical operations (i.e., addition, subtraction, multiplication and division of real numbers). In particular case, when $m = n$, $\mathbf{A}_1$ is a non-singular $n \times n$ matrix and $\mathbf{B} = \mathbf{A}_1^{-1}$ is simply computable with respect to $\mathbf{A}_1, \ldots, \mathbf{A}_k$, we call matrix $\mathbf{A}_1$ *simply invertible with respect to* $\mathbf{A}_2, \ldots, \mathbf{A}_k$. Finally, when $k = 1$ and $\mathbf{A}_1^{-1}$ is simply computable with respect to $\mathbf{A}_1$ we call $\mathbf{A}_1$ *simply invertible*.

Let $\mathcal{K}$ be a class of matrices of similar structure (diagonal, for example). Then an arbitrary matrix $\mathbf{A} \in \mathcal{K}$ represents class $\mathcal{K}$. Therefore, we shall denote $\mathrm{Inv}(\mathbf{A})$ the minimal number of arithmetical operations which is necessary to invert any matrix $\mathbf{B} \in \mathcal{K}$ assuming that $\mathbf{B}$ has the same dimensions as $\mathbf{A}$. Using this notation we can give another definition of simple invertibility considered above: $n \times n$ matrix $\mathbf{A}$ is simply invertible iff $\mathrm{Inv}(\mathbf{A}) = \mathrm{O}(n^2)$.

EXAMPLE 2.1. $\mathbf{C} = \mathbf{A} + \mathbf{B}$ is simply computable with respect to $\mathbf{A}$ and $\mathbf{B}$ but $\mathbf{C} = \mathbf{AB}$ in general is not simply computable with respect to $n \times n$ matrices $\mathbf{A}$ and $\mathbf{B}$ (since an algorithm multiplying arbitrary $n \times n$ matrices by $\mathrm{O}(n^2)$ operations is not known). From the other hand, $\mathbf{AB}$ may be simply computable for some special matrices $\mathbf{A}$ and $\mathbf{B}$. In particular case, when $\mathbf{A}$ or $\mathbf{B}$ is a vector then $\mathbf{AB}$ is simply computable with respect to $\mathbf{A}$ and $\mathbf{B}$. Indeed, let $\mathbf{c} = \mathbf{aB} = (c_1, \ldots, c_n)$. Then each $c_j = \sum_{i=1}^{n} a_i b_{ij}$ requires $2n - 1$ arithmetical operation ($n$ multiplications and $n - 1$ addition).

EXAMPLE 2.2. (Duda and Hart, 1976). Let

$$\mathbf{S} = \mathbf{S}(\mathbf{x}_1, \ldots, \mathbf{x}_N) = \frac{1}{N-1} \sum_{k=1}^{N} (\mathbf{x}_k - \bar{\mathbf{x}})^T (\mathbf{x}_k - \bar{\mathbf{x}}) \tag{1}$$

be *sample covariance matrix* of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$, where

$$\mathbf{x}_k = \left(x_k^1, \ldots, x_k^n\right) \quad \text{and} \quad \bar{\mathbf{x}} = \frac{1}{N}\sum_{k=1}^{N} \mathbf{x}_k.$$

Usually one needs $\mathrm{O}(Nn^2)$ operations to compute $n \times n$ matrix $\mathbf{S}$, therefore, in general $\mathbf{S}$ is not simply computable. Adding one new vector $\mathbf{x}_{N+1}$ we obtain new sample covariance matrix of vectors $\mathbf{x}_1, \ldots, \mathbf{x}_{N+1}$,

$$\mathbf{S}_+ = \frac{1}{N}\sum_{k=1}^{N+1} (\mathbf{x}_k - \bar{\mathbf{x}}_+)^T (\mathbf{x}_k - \bar{\mathbf{x}}_+), \quad \text{where} \quad \bar{\mathbf{x}}_+ = \frac{1}{N+1}\sum_{k=1}^{N+1} \mathbf{x}_k.$$

It is easy to verify (Duda and Hart, 1976) that matrix $\mathbf{S}_+$ can be expressed via matrix $\mathbf{S}$ and vectors $\bar{\mathbf{x}}$ and $\mathbf{x}_{N+1}$ in the following way:

$$\mathbf{S}_+ = \frac{N-1}{N}\mathbf{S} + \frac{1}{N+1}(\mathbf{x}_{N+1} - \bar{\mathbf{x}})^T (\mathbf{x}_{N+1} - \bar{\mathbf{x}}). \tag{2}$$

Formula (2) yields that computing of $\mathbf{S}_+$ requires $3n^2 + \mathrm{O}(n)$ arithmetical operations: (1) $\mathrm{O}(n)$ operations to obtain $\mathbf{z} = \mathbf{x}_{N+1} - \bar{\mathbf{x}}$, $\mathbf{y} = \mathbf{z}/(N+1)$ and $\lambda = (N-1)/N$; (2) $n^2$ multiplications for matrix $\mathbf{U} = \mathbf{y}^T\mathbf{z}$; (3) $n^2$ multiplications for $\lambda\mathbf{S}$; and (4) $n^2$ additions for $\lambda\mathbf{S} + \mathbf{U}$.

Therefore, $\mathbf{S}_+$ is simply computable with respect to $\mathbf{S}$, $\bar{\mathbf{x}}$ and $\mathbf{x}_{N+1}$. In Section 3.4 we will prove that $\mathbf{S}_+$ is simply invertible with respect to $\mathbf{S}$, $\bar{\mathbf{x}}$ and $\mathbf{x}_{N+1}$.

EXAMPLE 2.3. Let $\mathbf{A}$ be a non-singular $n \times n$ matrix. Then $\mathbf{B} = \lambda\mathbf{A}$ is simply invertible with respect to $\mathbf{A}^{-1}$ because $\mathbf{B}^{-1} = (1/\lambda)\mathbf{A}^{-1}$. In particular, matrix $\mathbf{A} = \lambda\mathbf{I}$, where $\mathbf{I}$ is a unit matrix, is simply invertible.

EXAMPLE 2.4. **Block-matrices** (Horn and Johnson, 1989). Let

$$\mathbf{A} = \begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix}$$

be a non-singular $n \times n$ matrix and $\mathbf{A}_{11}$, $\mathbf{A}_{12}$, $\mathbf{A}_{21}$ and $\mathbf{A}_{22}$ be matrices of dimensions $(n-r) \times (n-r)$, $(n-r) \times r$, $r \times (n-r)$ and $r \times r$, respectively. Let $\mathbf{A}_{11}$ and $\mathbf{A}_{22} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}$ be non-singular and $\mathbf{C} = \mathbf{A}_{11}^{-1}\mathbf{A}_{12}$, $\mathbf{D} = \mathbf{A}_{21}\mathbf{A}_{11}^{-1}$, $\mathbf{F} = (\mathbf{A}_{22} - \mathbf{D}\mathbf{A}_{12})^{-1}$. Then

$$\mathbf{A}^{-1} = \begin{pmatrix} \mathbf{A}_{11}^{-1} + \mathbf{CFD} & -\mathbf{CF} \\ -\mathbf{FD} & \mathbf{F} \end{pmatrix}. \tag{3}$$

It is easy to verify that for any $r = \text{const}$ matrix $\mathbf{A}$ is simply invertible with respect to $\mathbf{A}_{11}^{-1}$.

EXAMPLE 2.5. **Bartlett's formula** (Bartlett, 1952). An algorithm inverting the sum $\mathbf{C} = \mathbf{A} + \mathbf{B}$ of two arbitrary matrices $\mathbf{A}$ and $\mathbf{B}$ in general is not known. However, Bartlett (1952) suggested how to invert the sum $\mathbf{C}$ when one of the matrices $\mathbf{A}$ and $\mathbf{B}$ can be expressed as a product of two vectors (column and row) with some coefficients. Let $\mathbf{B} = \mathbf{A} + \lambda \mathbf{u}^T \mathbf{v}$, where $\mathbf{u}$ and $\mathbf{v}$ are vectors (rows). Then

$$\mathbf{B}^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{u}^T\mathbf{v}\mathbf{A}^{-1}}{1/\lambda + \mathbf{v}\mathbf{A}^{-1}\mathbf{u}^T}. \tag{4}$$

It is easy to verify that (4) requires $\mathrm{O}(n^2)$ arithmetical operations.

## 3. Some Classes of Simply Invertible Matrices

In this section we introduce some new classes of simply invertible matrices. In particular, we generalize formula (3) (see Section 3.1) and obtain its inverse formula (Section 3.2). We also will prove some analogues of (2) for modified covariance matrices (Section 3.4). In Section 4 these results will be applied to construct a fast prediction algorithm.

In Section 3.3 we propose the theorem concerning matrices having big common submatrix. We do not give its applications in this paper, however, this result can be useful for some statistical problems.

### 3.1. *Matrix Inverting Using Submatrix*

Let $\mathbf{A}$ be $n \times n$ matrix, $I, J \subseteq \{1, 2, \ldots, n\}$ be any nonempty sets of indices and $\bar{I} = \{1, 2, \ldots, n\} \setminus I$. By $\mathbf{A}_{IJ}$ we denote $|I| \times |J|$ submatrix of $\mathbf{A}$ including (in their natural order) all $\mathbf{A}$ elements $a_{ij}$ satisfying $i \in I$, $j \in J$. Now we are ready to prove the generalization of (3).

**Theorem 3.1.** *Let $|I| = |J| = n - k$, where $k = \mathrm{const}$ and matrices $\mathbf{A}_{IJ}$ and $\mathbf{A}_{\bar{I}\bar{J}} - \mathbf{A}_{\bar{I}J}\mathbf{A}_{IJ}^{-1}\mathbf{A}_{I\bar{J}}$ be non-singular. Then $\mathbf{A}$ is simply invertible with respect to $\mathbf{A}_{IJ}^{-1}$.*

*Proof.* Let $\mathbf{B} = \mathbf{A}^{-1}$, $\mathbf{C} = \mathbf{A}_{IJ}^{-1}\mathbf{A}_{I\bar{J}}$, $\mathbf{D} = \mathbf{A}_{\bar{I}J}\mathbf{A}_{IJ}^{-1}$ and $\mathbf{F} = (\mathbf{A}_{\bar{I}\bar{J}} - \mathbf{A}_{\bar{I}J}\mathbf{A}_{IJ}^{-1}\mathbf{A}_{I\bar{J}})^{-1}$. Since matrix $\mathbf{B}$ is formed by four nonintersecting submatrices $\mathbf{B}_{JI}$, $\mathbf{B}_{J\bar{I}}$, $\mathbf{B}_{\bar{J}I}$ and $\mathbf{B}_{\bar{J}\bar{I}}$, it is enough to prove the following formulae:

$$\begin{aligned} \mathbf{B}_{JI} &= \mathbf{A}_{IJ}^{-1} + \mathbf{CFD}, \\ \mathbf{B}_{J\bar{I}} &= -\mathbf{CF}, \\ \mathbf{B}_{\bar{J}I} &= -\mathbf{FD}, \\ \mathbf{B}_{\bar{J}\bar{I}} &= \mathbf{F}. \end{aligned} \tag{5}$$

Let $I = \{i_1, i_2, \ldots, i_{n-k}\}$, $J = \{j_1, j_2, \ldots, j_{n-k}\}$ be arbitrary subsets of $\{1, 2, \ldots, n\}$ and $\bar{I} = \{i_{n-k+1}, \ldots, i_n\}$, $\bar{J} = \{j_{n-k+1}, \ldots, j_n\}$. Let

$$p_{uv} = \begin{cases} 1, & \text{if } v = i_u, \\ 0, & \text{otherwise;} \end{cases} \quad \text{and} \quad r_{uv} = \begin{cases} 1, & \text{if } u = j_v, \\ 0, & \text{otherwise;} \end{cases} \quad (u, v = 1, 2, \ldots, n).$$

Then $\mathbf{P} = (p_{uv})$ and $\mathbf{R} = (r_{uv})$ are permutation matrices satisfying $\mathbf{P}^{-1} = \mathbf{P}^T$ and $\mathbf{R}^{-1} = \mathbf{R}^T$. Let $\widetilde{\mathbf{A}} = \mathbf{PAR}$. Then $\widetilde{\mathbf{B}} = (\widetilde{\mathbf{A}})^{-1} = \mathbf{R}^{-1}\mathbf{A}^{-1}\mathbf{P}^{-1} = \mathbf{R}^T\mathbf{B}\mathbf{P}^T$ and

$$\widetilde{\mathbf{A}} = \begin{pmatrix} \widetilde{\mathbf{A}}_{11} & \widetilde{\mathbf{A}}_{12} \\ \widetilde{\mathbf{A}}_{21} & \widetilde{\mathbf{A}}_{22} \end{pmatrix}, \qquad \widetilde{\mathbf{B}} = \begin{pmatrix} \widetilde{\mathbf{B}}_{11} & \widetilde{\mathbf{B}}_{12} \\ \widetilde{\mathbf{B}}_{21} & \widetilde{\mathbf{B}}_{22} \end{pmatrix},$$

where submatrices $\widetilde{\mathbf{A}}_{ij}$ and $\widetilde{\mathbf{B}}_{ij}$ satisfy $\widetilde{\mathbf{A}}_{11} = \mathbf{A}_{IJ},\ \widetilde{\mathbf{A}}_{12} = \mathbf{A}_{I\bar{J}},\ \widetilde{\mathbf{A}}_{21} = \mathbf{A}_{\bar{I}J},\ \widetilde{\mathbf{A}}_{22} = \mathbf{A}_{\bar{I}\bar{J}}$ and $\widetilde{\mathbf{B}}_{11} = \mathbf{B}_{IJ},\ \widetilde{\mathbf{B}}_{12} = \mathbf{B}_{I\bar{J}},\ \widetilde{\mathbf{B}}_{21} = \mathbf{B}_{\bar{I}J},\ \widetilde{\mathbf{B}}_{22} = \mathbf{B}_{\bar{I}\bar{J}}$. Now applying Eq. (3) for matrices $\widetilde{\mathbf{A}}$ and $\widetilde{\mathbf{B}}$ we obtain (5).

### 3.2. *Submatrix Inverting*

**Theorem 3.2.** *Let $\mathbf{A}$ be non-singular $n \times n$ matrix, $\mathbf{B} = \mathbf{A}^{-1}$, $|I| = |J| = n - k$, where $k = \text{const}$ and $\mathbf{B}_{\bar{J}\bar{I}}$ be non-singular. Then $\mathbf{A}_{IJ}$ is simply invertible with respect to $\mathbf{A}^{-1}$.*

*Proof.* It is enough to prove the formula

$$\mathbf{A}_{IJ}^{-1} = \mathbf{B}_{JI} - \mathbf{B}_{J\bar{I}}\mathbf{B}_{\bar{J}\bar{I}}^{-1}\mathbf{B}_{\bar{J}I}. \tag{6}$$

From Eq. (5) we obtain $\mathbf{B}_{JI} = \mathbf{A}_{IJ}^{-1} - \mathbf{B}_{J\bar{I}}\mathbf{D}$ and $\mathbf{B}_{\bar{J}I} = -\mathbf{B}_{\bar{J}\bar{I}}\mathbf{D}$. Eliminating $\mathbf{D}$ from the last equation and substituting into the first we get the desired result.

We will apply Eq. (6) in our fast prediction algorithm (see Section 4).

### 3.3. *Inverting of Modified Matrices*

Theorems 3.1 and 3.2 yield more general result. The following theorem can be applied, for example, for two "big" matrices $\mathbf{A}$ and $\mathbf{B}$ such that matrix $\mathbf{B}$ was obtained from matrix $\mathbf{A}$ by substituting of "small" number of its rows by new rows.

**Theorem 3.3.** *Let $\mathbf{A}$ and $\mathbf{B}$ be two non-singular $n \times n$ matrices having $m \times m$ submatrices $\mathbf{A}_{IJ}$ and $\mathbf{B}_{KL}$, respectively, which coincide or differ only in order of rows and columns, i.e., $\mathbf{B}_{KL} = \mathbf{P}\mathbf{A}_{IJ}\mathbf{R}$ for some permutation matrices $\mathbf{P}$ and $\mathbf{R}$. (Here $I, J, K, L \subseteq \{1, 2, \ldots, n\}$ and $|I| = |J| = |K| = |L| = m$.)*

*If $m = n - k$, where $k = \text{const}$ then $\mathbf{B}$ is simply invertible with respect to $\mathbf{A}^{-1}$.*

*Proof.* By Theorem 3.1, $\mathbf{B}$ is simply invertible with respect to $\mathbf{B}_{KL}^{-1}$. By Theorem 3.2, $\mathbf{A}_{IJ}^{-1}$ is simply invertible with respect to $\mathbf{A}^{-1}$. Finally, $\mathbf{B}_{KL}^{-1} = \mathbf{R}^T\mathbf{A}_{IJ}^{-1}\mathbf{P}^T$ is simply computable with respect to $\mathbf{A}_{IJ}^{-1}$.

### 3.4. *Modified Sample Covariance Matrices*

Let us consider sample covariance matrix $\mathbf{S} = \mathbf{S}(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ defined by Eq. (1). In many learning processes the set of vectors changes a little at each iteration, and after any change a new sample covariance matrix is needed. We propose formulae to compute new matrix and its inverse with less arithmetic operations. Denote $\mathbf{S}_+ = \mathbf{S}(\mathbf{x}_1, \ldots, \mathbf{x}_{N+1})$, $\mathbf{S}_- = \mathbf{S}(\mathbf{x}_2, \ldots, \mathbf{x}_N)$ and $\mathbf{S}_\mp = \mathbf{S}(\mathbf{x}_2, \ldots, \mathbf{x}_{N+1})$.

**Theorem 3.4.** *Matrices* $\mathbf{S}_+,\ \mathbf{S}_-$ *and* $\mathbf{S}_\mp$ *are simply computable with respect to* $\mathbf{S},\ \bar{\mathbf{x}},\ \mathbf{x}_1$ *and* $\mathbf{x}_{N+1}$.

*Proof.* Simple computability of $\mathbf{S}_+$ was proved above (Example 2.2 and Eq. (2)). Let us prove a similar formula for $\mathbf{S}_-$:

$$\mathbf{S}_- = \frac{N-1}{N-2}\mathbf{S} - \frac{N}{(N-1)(N-2)}(\mathbf{x}_1 - \bar{\mathbf{x}})^T(\mathbf{x}_1 - \bar{\mathbf{x}}). \tag{7}$$

Let $\mathbf{x}_- = (\mathbf{x}_2 + \cdots + \mathbf{x}_N)/(N-1)$. Then

$$\begin{aligned}
\mathbf{S} &= \frac{1}{N-2}\sum_{k=1}^{N-1}(\mathbf{x}_k - \bar{\mathbf{x}}_-)^T(\mathbf{x}_k - \bar{\mathbf{x}}_-) \\
&= \frac{1}{N-2}\left(\sum_{k=1}^{N}\mathbf{x}_k^T\mathbf{x}_k - \mathbf{x}_1^T\mathbf{x}_1 - (N-1)\bar{\mathbf{x}}_-^T\bar{\mathbf{x}}_-\right) \\
&= \frac{1}{N-2}\left((N-1)\mathbf{S} + N\bar{\mathbf{x}}^T\bar{\mathbf{x}} - \mathbf{x}_1^T\mathbf{x}_1 - \frac{1}{N-1}(N\bar{\mathbf{x}} - \mathbf{x}_1)^T(N\bar{\mathbf{x}} - \mathbf{x}_1)\right)
\end{aligned}$$

and the last equation yields (7). From Eqs. (2) and (7) we obtain

$$\begin{aligned}
\mathbf{S}_\mp &= \mathbf{S} - \frac{N}{(N-1)^2}(\mathbf{x}_1 - \bar{\mathbf{x}})^T(\mathbf{x}_1 - \bar{\mathbf{x}}) \\
&\quad + \frac{1}{N}\left(\mathbf{x}_{N+1} + \frac{1}{N-1}\mathbf{x}_1 - \frac{N}{N-1}\bar{\mathbf{x}}\right)^T \\
&\quad \times \left(\mathbf{x}_{N+1} + \frac{1}{N-1}\mathbf{x}_1 - \frac{N}{N-1}\bar{\mathbf{x}}\right),
\end{aligned} \tag{8}$$

and this ends the proof.

COROLLARY 3.1. Matrices $\mathbf{S}_+,\ \mathbf{S}_-$ and $\mathbf{S}_\mp$ are simply invertible with respect to $\mathbf{S},\ \bar{\mathbf{x}},\ \mathbf{x}_1$ and $\mathbf{x}_{N+1}$.

*Proof.* This follows from Theorem 3.4 and Eqs. (2), (7) and (8). We propose only the expression for $\mathbf{S}_\mp^{-1}$. Let $\mathbf{u} = \mathbf{x}_1 - \bar{\mathbf{x}}$ and $\mathbf{v} = \mathbf{u}/(N-1) + \mathbf{x}_{N+1} - \bar{\mathbf{x}}$. Then

$$\begin{aligned}
&\mathbf{S}_\mp^{-1} = \mathbf{C} + \left(N - 2 + 1/N - \mathbf{u}\mathbf{C}\mathbf{u}^T\right)\mathbf{C}\mathbf{u}^T\mathbf{u}\mathbf{C}, \\
&\mathbf{C} = \mathbf{S}^{-1} + \left(N + \mathbf{v}\mathbf{S}^{-1}\mathbf{v}^T\right)^{-1}\mathbf{S}^{-1}\mathbf{v}^T\mathbf{v}\mathbf{S}^{-1}.
\end{aligned} \tag{9}$$

It is interesting that formula (9) cannot be improved:

PROPOSITION 3.1. It is imposible to construct linear combinations $\mathbf{u}$ and $\mathbf{v}$ of vectors $\mathbf{x}_1,\ \mathbf{x}_{N+1}$ and $\bar{\mathbf{x}}$ such that $\mathbf{S}_{\mp} = \mathbf{S} + \mathbf{u}^T\mathbf{v}$.

*Proof.* Let $\mathbf{u} = a\mathbf{x}_1 + b\mathbf{x}_{N+1} + c\bar{\mathbf{x}},\ \mathbf{v} = d\mathbf{x}_1 + e\mathbf{x}_{N+1} + f\bar{\mathbf{x}}$ and $\mathbf{S}_{\mp} = \mathbf{S} + \mathbf{u}^T\mathbf{v}$. Substituting $\mathbf{u}$ and $\mathbf{v}$ into the last equation and comparing with (9) we obtain incompatible equations: $ce = bf = -1/(N-1),\ cf = 0$.

## 4. Fast Prediction Algorithm

In this section we propose one possible application of theoretical results presented above. Namely, we present a fast algorithm for predicting future values of measurements in some plant. This algorithm can be used for failure detection.

Let us consider the following model. We measure the values of $n$ parameters $x^1, \dots, x^n$ at moments $t_1, t_2, \dots$. Using the last $N$ values at any moment $t_k$ $(k \geqslant N)$ we predict the values for the moment $t_{k+1}$. Then we compare predicted and measured (at the moment $t_{k+1}$) values and raise an alarm if they are strongly different. If our prediction procedure outputs predicted values for the next moment $t_{k+1}$ before this moment (i.e., it runs in time $< \Delta t = t_{k+1} - t_k$) then we call this model *prediction in real-time*. It can be used, for example, for safety control in nuclear power station (compare with (Tylee, 1983)). We take $\Delta t = 1$ s, since according to Tylee (1983, p. 412) "... sampling the plant measurements every second ... provides excellent failure detection results".

We use multiple linear regression for prediction. Let $y = x^i,\ \mathbf{x} = (x^1, \dots, x^{i-1}, x^{i+1}, \dots, x^n)$, $Y = \{i\}$ and $X = \{1, \dots, i-1,\ i+1, \dots, n\}$. Let also denote by $\mathbf{x}_k = (x_k^1, \dots, x_k^n)$ the vector of measured values at the moment $t_k$. Using our notation from Section 3.1 a linear regression equation can be written in the following way:

$$y = \mathbf{x}\mathbf{S}_{XX}^{-1}\mathbf{S}_{XY} + \big(\bar{y} - \bar{\mathbf{x}}\mathbf{S}_{XX}^{-1}\mathbf{S}_{XY}\big),$$

where

$$\bar{y} = \frac{1}{N}\sum_{j=k-N+1}^{k} x_j^i, \qquad \bar{\mathbf{x}} = \frac{1}{N}\sum_{j=k-N+1}^{k}\big(x_j^1, \dots, x_j^{i-1},\ x_j^{i+1}, \dots, x_j^n\big)$$

are means and $\mathbf{S} = \mathbf{S}(\mathbf{x}_{k-N+1}, \dots, \mathbf{x}_k)$ is the sample covariance matrix. Components of the vector $\mathbf{w}_i = \mathbf{S}_{XX}^{-1}\mathbf{S}_{XY}$ and $w_i^0 = \bar{y} - \bar{\mathbf{x}}\mathbf{w}_i$ are called *regression coefficients*. Notice that usually in linear regression a so called matrix of centred sums of squares and products is used (Maindonald, 1988) instead of matrix $\mathbf{S}$:

$$\widetilde{\mathbf{x}}^T\widetilde{\mathbf{x}} = (N-1)\mathbf{S} = \sum_{j=k-N+1}^{k}(\mathbf{x}_j - \bar{\mathbf{x}})^T(\mathbf{x}_j - \bar{\mathbf{x}}).$$

Obviously, in both cases we obtain the same regression coefficients.

Now we propose an algorithm for the prediction in real-time problem. Suppose that at moment $t_k$ we already have matrices $\mathbf{S} = \mathbf{S}(\mathbf{x}_{k-N+1}, \ldots, \mathbf{x}_k)$, $\mathbf{S}^{-1}$ and means

$$\bar{y}^i = \frac{1}{N} \sum_{j=k-N+1}^{k} x_j^i \quad \text{and} \quad \bar{\mathbf{x}}^i = \frac{1}{N} \sum_{j=k-N+1}^{k} \left(x_j^1, \ldots, x_j^{i-1},\ x_j^{i+1}, \ldots, x_j^n\right).$$

**Fast Prediction Algorithm**

*Step* 1. For any $y = x^i$ $(i = 1, \ldots, n)$ compute regression coefficients

$$\mathbf{w}_i = \mathbf{S}_{XX}^{-1}\mathbf{S}_{XY}, \qquad w_i^0 = \bar{y}^i - \bar{\mathbf{x}}^i\mathbf{S}_{XX}^{-1}\mathbf{S}_{XY},$$

computing $\mathbf{S}_{XX}^{-1}$ according to (6).

*Step* 2. For any $y = x^i$ $(i = 1, \ldots, n)$ compute predicted values $\widehat{y}_{k+1} = \mathbf{x}_{k+1}\mathbf{w}_i + w_i^0$ and compare them with measured values $x_{k+1}^i$.

*Step* 3. Using (8) and (9) compute new matrices $\mathbf{S}_{\mp} = \mathbf{S}(\mathbf{x}_{k-N+2}, \ldots, \mathbf{x}_{k+1})$, $\mathbf{S}_{\mp}^{-1}$ and new means $\bar{y}^i := \bar{y}^i + (x_{k+1}^i - x_{k-N+1}^i)/N$ and

$$\begin{aligned}\bar{\mathbf{x}}^i := \bar{\mathbf{x}}^i + \frac{1}{N}\Big[\ &\big(\ x_{k+1}^1, \ldots, x_{k+1}^{i-1},\ x_{k+1}^{i+1}, \ldots, x_{k+1}^n\big)\\ &- \big(x_{k-N+1}^1, \ldots, x_{k-N+1}^{i-1},\ x_{k-N+1}^{i+1}, \ldots, x_{k-N+1}^n\big)\Big],\end{aligned}$$

and return to *Step* 1.

It is easy to verify that this algorithm requires $\mathrm{O}(n^3)$ operations to make one iteration from $t_k$ to $t_{k+1}$ meanwhile the standard methods (Maindonald, 1988) require $\Omega(Nn^2) + \Omega(n^4)$ operations.

EXAMPLE 4.1. Fast prediction algorithm was realised in Matlab on PC with 200 MHz Pentium processor for $N = 100, 200, 800$ and $n = 8, 16, 32, 64, 128$. Table 1 shows CPU time (in seconds) used to make 1 iteration (from $t_k$ to $t_{k+1}$). In this table "fast"

Table 1

Computing time (in seconds) used for 1 iteration

| | $N = 100$ | | $N = 200$ | | $N = 800$ | |
|---|---|---|---|---|---|---|
| $n$ | fast | stand. | fast | stand. | fast | stand. |
| 8 | 0.009 | 0.055 | 0.010 | 0.110 | 0.011 | 0.407 |
| 16 | 0.027 | 0.224 | 0.028 | 0.428 | 0.028 | 1.647 |
| 32 | 0.146 | 1.055 | 0.148 | 2.087 | 0.149 | 8.124 |
| 64 | 0.977 | 5.872 | 0.978 | 9.535 | 0.983 | 43.001 |
| 128 | — | — | 7.402 | 60.122 | 7.460 | 4 min 19 s |

means fast prediction algorithm and "stand." means standard linear regression realised using Matlab tools. From Table 1 we conclude that fast prediction algorithm is much faster than standard one and it can work in real-time even on PC (at least for $n \leqslant 64$ parameters).

## 5. Conclusions

Some classes of simply invertible matrices are studied and formulae for inverse matrices are given. The usefulness of these formulae in statistical computations is demonstrated. As one could easily notice in most of cases considered above there was not any reason to confine ourselves only on complexity $\mathrm{O}(n^2)$. A lot of $n \times n$ matrices can be inverted using $L$ operations for $L$ lying somewhere between $\mathrm{O}(n^2)$ and $\mathrm{O}(n^3)$. The aim of this paper, however, was to consider only the extreme case $L = \mathrm{O}(n^2)$.

We also draw reader's attention that this paper did not touch upon the problems of existence and computation accuracy of inverse matrices. These questions require separate investigation, see, e.g., (Horn and Johnson, 1989).

## Acknowledgements

The author would like to thank Professor Š. Raudys for the suggestion of research topic and helpful discussions. The author is also grateful to the reviewers by their useful comments.

## References

Aho, A.V., J.E. Hopcroft and J.D. Ullman (1979). *The Design and Analysis of Computer Algorithms*. Mir, Moscow. 536 pp. (in Russian).

Bartlett, M.S. (1952). An inverse matrix adjustment arising in discriminant analysis. *Ann. Math. Stat.*, **22**, 207.

Coppersmith, D. and Sh. Winograd (1986). Matrix multiplication via arithmetic progression. IBM Thomas Watson Research Center, Preprint.

Duda, R.O., and P.E. Hart (1976). *Pattern Classification and Scene Analysis*. Mir, Moscow. 511 pp. (in Russian).

Horn, R.A., and C.R. Johnson (1989). *Matrix Analysis*. Mir, Moscow. 655 pp. (in Russian).

Maindonald, J.H. (1988). *Statistical Computation*. Finansy i statistika, Moscow. 349 pp. (in Russian).

Strassen, V. (1969). Gaussian elimination is not optimal. *Numer. Math.*, **13**(4), 354–356.

Tylee, J.L. (1983). On-line failure detection in nuclear power plant instrumentation. *IEEE Trans. Autom. Control*, **28**(3), 406–415.

**V. Dičiūnas** received mathematician diploma from Moscow State University in 1982. He is a researcher at the Department of Informatics of Vilnius University and at the Institute of Mathematics and Informatics. His research interests include artificial neural networks, statistical classification methods and complexity theory.

## Paprastai apgręžiamos matricos ir greitas prognozavimas

Valdas DIČIŪNAS

Straipsnyje nagrinėjama didelių $n \times n$ matricų apgrąža. Žinoma nemažai algoritmų, ypač matematinėje statistikoje ir skaitinėje matematikoje, kuriuose žingsnis po žingsnio reikia apgręžti daug didelių matricų. Dažnai tos matricos būna glaudžiai susijusios tarpusavyje. Standartiniai apgrąžos metodai atlieka O$(n^3)$ aritmetinių operacijų, todėl didelėms $n$ reikšmėms šie algoritmai gali dirbti labai ilgai. Straipsnyje aprašomos kelios matricų klasės, kuriose naudojant kitų (susijusių) matricų atvirkštines, nurodytas matricas galima apgręžti panaudojus tik O$(n^2)$ operacijų. Svarbiausios iš šių matricų klasių yra matricos, turinčios didelę bendrą pomatricę, ir modifikuotos kovariacinės matricos. Teoriniai rezultatai pritaikyti kuriant labai greitą prognozavimo algoritmą, kuris galėtų dirbti realiuoju laiku. Šis algoritmas parodo pasiūlytų apgrąžos metodų privalumus ir galėtų būti naudojamas, pavyzdžiui, gamyklos technologinio proceso saugumui kontroliuoti.