

# Linear Algebraic Equations, SVD, and the Pseudo-Inverse

Philip N. Sabes

October 18, 2001

## 1 A Little Background

### 1.1 Singular values and matrix inversion

For non-symmetric matrices, the eigenvalues and singular values are not equivalent. However, they share one important property:

**Fact 1** A matrix  $\mathbf{A}$ ,  $N \times N$ , is invertible iff all of its singular values are non-zero.

**Proof outline:**

**if:** We have  $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ . Suppose that  $\mathbf{\Sigma}$  has no zeros on the diagonal. Then  $\mathbf{B} = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T$  exists. And  $\mathbf{AB} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T\mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^T = \mathbf{I}$ , and similarly  $\mathbf{BA} = \mathbf{I}$ . Thus  $\mathbf{A}$  is invertible.

**only if:** Given that  $\mathbf{A}$  is invertible, we will construct  $\mathbf{\Sigma}^{-1}$ , which is sufficient to show that SVs are all non-zero:

$$\begin{aligned}\mathbf{A} &= \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \\ \mathbf{\Sigma} &= \mathbf{U}^T\mathbf{A}\mathbf{V} \\ \mathbf{\Sigma}^{-1} &= \mathbf{V}^T\mathbf{A}^{-1}\mathbf{U}\end{aligned}$$

### 1.2 Some definitions from vector calculus

In the following, we will want to find the value of a vector  $\mathbf{x}$  which minimizes a scalar function  $f(\mathbf{x})$ . In order to do this, we first need a few basic definitions from vector calculus.

**Definition 1** The (partial) derivative of a scalar  $a$  with respect to a vector  $\mathbf{x}$ ,  $N \times 1$ , is the  $1 \times N$  vector

$$\frac{\partial a}{\partial \mathbf{x}} = \left[ \frac{\partial a}{\partial x_1} \cdots \frac{\partial a}{\partial x_N} \right]$$

In practice, when only derivatives of scalars are used, people often write  $\frac{\partial a}{\partial \mathbf{x}}$  as an  $N \times 1$  column vector (i.e. the transpose of the definition above). However the row vector definition is preferable, as it's consistent with the following:

**Definition 2** The (partial) derivative of a vector  $\mathbf{b}$ ,  $M \times 1$ , with respect to a vector  $\mathbf{x}$ ,  $N \times 1$ , is the  $M \times N$  matrix

$$\frac{\partial \mathbf{b}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial b_1}{\partial x_1} & \cdots & \frac{\partial b_1}{\partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial b_N}{\partial x_1} & \cdots & \frac{\partial b_N}{\partial x_N} \end{bmatrix}$$

---

©2001, Philip N. Sabes. These notes should not be distributed or posted on the web without permission; requests for permission to [sabes@phy.ucsf.edu](mailto:sabes@phy.ucsf.edu)

**Exercise 1** Let  $\mathbf{x}, \mathbf{y}$  be  $N \times 1$  vectors and  $\mathbf{A}$  be an  $N \times N$  matrix. Show that the following are correct:

$$\begin{aligned} \frac{\partial \mathbf{y}^T \mathbf{x}}{\partial \mathbf{x}} &= \mathbf{y}^T \\ \frac{\partial \mathbf{A} \mathbf{x}}{\partial \mathbf{x}} &= \mathbf{A} \\ \frac{\partial \mathbf{x}^T \mathbf{A} \mathbf{x}}{\partial \mathbf{x}} &= 2\mathbf{x}^T \mathbf{A}, \text{ if } \mathbf{A} \text{ is symmetric} \end{aligned}$$

In ordinary Calculus, to find a local maximum or minimum of a function,  $f(x)$ , we solve the equation  $\partial f(x)/\partial x = 0$ . A similar principle holds for a multivariate function (i.e. a function of a vector):

**Fact 2** Let  $f(\mathbf{x})$  be a scalar function of an  $N \times 1$  vector. Then  $\partial f(\mathbf{x})/\partial \mathbf{x} = \mathbf{0}$  at  $\mathbf{x} = \mathbf{x}^*$  iff  $\mathbf{x}^*$  is a local minimum, a local maximum or a saddle point of  $f$ .

Analogous to the scalar case, second partial derivatives must be checked to determine the kind of extremum. In practice, one often knows that the function  $f$  is either convex or concave.

## 2 Solving Linear Algebraic Equations

From High School algebra, everyone should know how to solve  $N$  coupled linear equations with  $N$  unknowns. For example, consider the  $N=2$  case below:

$$\begin{aligned} 2x + y &= 4 \\ 2x - y &= 8. \end{aligned}$$

First you'd probably add the two equations to eliminate  $y$  and solve for  $x$ :  $4x = 12$  yields  $x = 3$ . Then you'd substitute  $x$  into one of the equations to solve for  $y$ :  $y = 4 - 6 = -2$ . This is easy enough, but it gets somewhat hairy for large  $N$ .

We can simplify the notation, at least, by rewriting the problem as a matrix equation:

$$\underbrace{\begin{bmatrix} 2 & 1 \\ 2 & -1 \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 4 \\ 8 \end{bmatrix}$$

Now, assuming that  $\mathbf{A}$  is invertible, we can write the solution as

$$\begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{A}^{-1} \begin{bmatrix} 4 \\ 8 \end{bmatrix} = \left( \frac{1}{4} \begin{bmatrix} 1 & 1 \\ 2 & -2 \end{bmatrix} \right) \begin{bmatrix} 4 \\ 8 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \end{bmatrix}$$

While this looks like a cleaner approach, in reality it just pushes the work into the computation of  $\mathbf{A}^{-1}$ . And in fact, the basic methods of matrix inversion use "backsubstitution" algorithms which are similar to the eliminate and substitute method we above. Still, this notation shows us something. Every time we compute the inverse of a full-rank matrix  $\mathbf{A}$ , we have essentially solved the whole class of linear equations,  $\mathbf{A} \mathbf{x} = \mathbf{y}$ , for any  $\mathbf{y}$ . The SVD of  $\mathbf{A}$  makes the geometry of the situation clear:

$$\begin{aligned} \mathbf{A} \mathbf{x} &= \mathbf{y} \\ \mathbf{U} \mathbf{\Sigma} \mathbf{V}^T \mathbf{x} &= \mathbf{y} \\ \mathbf{\Sigma} \underbrace{\mathbf{V}^T \mathbf{x}}_{\tilde{\mathbf{x}}} &= \underbrace{\mathbf{U}^T \mathbf{y}}_{\tilde{\mathbf{y}}} \\ \tilde{\mathbf{x}} &= \mathbf{\Sigma}^{-1} \tilde{\mathbf{y}} \\ \tilde{x}_i &= \tilde{y}_i / \sigma_i \end{aligned} \tag{1}$$

Of course not all sets of  $N$  equations with  $N$  unknowns have a solution. An example will illustrate the problem that arises.

**Example**

$$\begin{bmatrix} 1 & 1 \\ 2 & 2 \end{bmatrix} \mathbf{x} = \mathbf{y}$$

Express the matrix in terms of its SVD,  $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ :

$$\begin{bmatrix} 1 & -2 \\ 2 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \mathbf{x} = \mathbf{y}$$

Invert  $\mathbf{U}$  and combine  $\mathbf{\Sigma}$  and  $\mathbf{V}^T$ :

$$\begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x} = \frac{1}{5} \begin{bmatrix} 1 & 2 \\ -2 & 1 \end{bmatrix} \mathbf{y}$$

$$\begin{bmatrix} x_1 + x_2 \\ 0 \end{bmatrix} = \frac{1}{5} \begin{bmatrix} y_1 + 2y_2 \\ -2y_1 + y_2 \end{bmatrix}$$

Clearly, a solution only exists when  $y_2 = 2y_1$ , i.e. when  $\mathbf{y}$  lives in the range of  $\mathbf{A}$ . The problem here is that  $\mathbf{A}$  is “deficient”, i.e. not full-rank.

Consider the general case where  $\mathbf{A}$ ,  $N \times N$ , has rank  $M$ . This means that  $\mathbf{A}$  has  $N-M$  zero-valued SVs:

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_1 & & & & & \\ & \ddots & & & & \\ & & \sigma_M & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{\Sigma}_M & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

If we try to solve the equations as in Equation 1, we hit a snag:

$$\mathbf{Ax} = \mathbf{y}$$

$$\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T \mathbf{x} = \mathbf{y}$$

$$\begin{bmatrix} \mathbf{\Sigma}_M & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tilde{\mathbf{x}} = \tilde{\mathbf{y}}$$

$$\tilde{x}_i = \tilde{y}_i / \sigma_i \text{ for } i \leq M \text{ only}$$

In the SVD-rotated space, we can only “access” the first  $M$  elements of the solution.

When a square matrix  $\mathbf{A}$  is deficient, the columns are linearly dependent, meaning that we really only have  $M < N$  (independent) unknowns. But we still have  $N$  equations/constraints. In the next two sections we’ll further explore the general problem of  $N$  equations,  $M$  unknowns. We’ll start with the case of under-constrained equations,  $M > N$ , and then return to over-constrained equations in Section 4.

### 3 Under-constrained Linear Algebraic Equations

If there are more unknowns  $M$  than equations  $N$ , the problem is under-constrained or “ill-posed”:

$$\begin{matrix} [ & \mathbf{A} & ] \\ & N \times M & \end{matrix} \begin{matrix} \left[ \begin{matrix} \mathbf{x} \\ \end{matrix} \right] \\ & M \times 1 & \end{matrix} = \begin{matrix} [ & \mathbf{y} & ] \\ & N \times 1 & \end{matrix} \quad (N < M) \quad (2)$$

**Definition 3** The “row-rank” of an  $N \times M$  matrix is the dimension of the subspace of  $\mathfrak{R}^M$  spanned by its  $N$  rows. A matrix is said to have “full row-rank” if its row-rank is  $N$ , i.e. if the rows are a linearly independent set. Clearly this can only be the case if  $N \leq M$ .

The row-rank of a matrix is equal to its rank, i.e. the number of non-zero SVs.

In this section, we will address problems of the form of Equation 2 where  $\mathbf{A}$  has full row-rank, i.e. the first  $N$  singular values are non-zero.

**Example** Consider a simple example with 1 equation and 2 unknowns:  $x_1 + x_2 = 4$ . There are an infinite number of solutions to this equation, and they lie along a line in  $\mathfrak{R}^2$ .

#### 3.1 A subspace of solutions

We can determine the solution space for the general case of Equation 2 using the SVD of  $\mathbf{A}$ :

$$\begin{matrix} [ & \mathbf{A} & ] \\ & N \times M & \end{matrix} \begin{matrix} \left[ \begin{matrix} \mathbf{x} \\ \end{matrix} \right] \\ & M \times 1 & \end{matrix} = \begin{matrix} [ & \mathbf{y} & ] \\ & N \times 1 & \end{matrix} \quad (3)$$

$$\begin{matrix} [ & \mathbf{U} & ] & [ & \mathbf{\Sigma}_N & \mathbf{0} & ] \\ & N \times N & & & N \times M & \end{matrix} \begin{matrix} \left[ \begin{matrix} \mathbf{V}^T \\ \end{matrix} \right] \\ & M \times M & \end{matrix} \begin{matrix} \left[ \begin{matrix} \mathbf{x} \\ \end{matrix} \right] \\ & M \times 1 & \end{matrix} = \begin{matrix} [ & \mathbf{y} & ] \\ & N \times 1 & \end{matrix}$$

Using our previous results on SVD, we can rewrite Equation 3 as

$$\sum_{i=1}^N \sigma_i \mathbf{u}_i (\mathbf{v}_i^T \mathbf{x}) = \mathbf{y} \quad (4)$$

In other words, the only part of  $\mathbf{x}$  that matters is the component that lies in the  $N$ -dimensional subspace of  $\mathfrak{R}^M$  spanned by the first  $N$  columns of  $\mathbf{V}$ . Thus, the addition of any component that lies in the null-space of  $\mathbf{A}$  will make no difference: if  $x^*$  is any solution to Equation 3, so is  $x^* + \sum_{i=N+1}^M \alpha_i \mathbf{v}_i$ , for any  $\{\alpha_i\}$ .

**Example** Consider again the equation  $x_1 + x_2 = 4$ :

$$\begin{bmatrix} 1 & 1 \end{bmatrix} \mathbf{x} = 4$$

Replacing the matrix with its SVD,  $\mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ :

$$\begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \mathbf{x} = 4$$

The null-space of  $\mathbf{A} = \begin{bmatrix} 1 & 1 \end{bmatrix}$  is given by  $\alpha \begin{bmatrix} 1 & -1 \end{bmatrix}$ , where  $\alpha$  is a free scalar variable. Since  $x_1 = x_2 = 2$  is one solution, the class of solutions is given by:

$$\begin{bmatrix} 2 \\ 2 \end{bmatrix} + \alpha \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

### 3.2 Regularization

In general, under-constrained problems can be made well-posed by the addition of a “regularizer”, i.e. a cost-function that we’d like the solution to minimize. In the case of under-constrained linear equations, we know that the solution space lies in an  $(M-N)$  dimensional subspace of  $\mathbb{R}^M$ . One obvious regularizer would be to pick the solution that has the minimum square norm, i.e. the solution that is closest to the origin. The new, well-posed version of the problem can now be stated as follows:

$$\text{Find the vector } \mathbf{x} \text{ which minimizes } \mathbf{x}^T \mathbf{x}, \text{ subject to the constraint } \mathbf{A}\mathbf{x} = \mathbf{y} \quad (5)$$

In order to solve Equation 5, we will need to make use of the following,

**Fact 3** *If  $\mathbf{A}$ ,  $N \times M$ ,  $N \leq M$ , has full row-rank, then  $\mathbf{A}\mathbf{A}^T$  is invertible.*

**Proof:**

$$\mathbf{A} = \mathbf{U} \begin{bmatrix} \Sigma_N & 0 \end{bmatrix} \mathbf{V}^T$$

where  $\Sigma_N$  is invertible. Therefore,

$$\begin{aligned} \mathbf{A}\mathbf{A}^T &= \mathbf{U} \begin{bmatrix} \Sigma_N & 0 \end{bmatrix} \mathbf{V}^T \mathbf{V} \begin{bmatrix} \Sigma_N \\ 0 \end{bmatrix} \mathbf{U}^T \\ &= \mathbf{U} \begin{bmatrix} \Sigma_N & 0 \end{bmatrix} \begin{bmatrix} \Sigma_N \\ 0 \end{bmatrix} \mathbf{U}^T \\ &= \mathbf{U} \Sigma_N^2 \mathbf{U}^T, \end{aligned}$$

where  $\Sigma_N^2$  is the  $N \times N$  diagonal matrix with  $\sigma_i^2$  on the  $i$ th diagonal. Since  $\Sigma_N$  has no zero elements on the diagonal, neither does  $\Sigma_N^2$ . Therefore  $\mathbf{A}\mathbf{A}^T$  is invertible (and symmetric, positive-definite).

It is also worth noting here that since that each matrix in the last equation above is invertible, we can write down the SVD (and eigenvector decomposition) of  $(\mathbf{A}\mathbf{A}^T)^{-1}$  by inspection:  $(\mathbf{A}\mathbf{A}^T)^{-1} = \mathbf{U} \Sigma_N^{-2} \mathbf{U}^T$ .

We will now solve the problem stated in Equation 5 using Lagrange multipliers. We assume that  $\mathbf{A}$  has full row-rank. Let

$$H = \mathbf{x}^T \mathbf{x} + \lambda^T (\mathbf{A}\mathbf{x} - \mathbf{y}).$$

The solution is found by solving the equation  $\partial H / \partial \mathbf{x} = 0$  (see Section 1.2) and then ensuring that the constraint  $(\mathbf{A}\mathbf{x} = \mathbf{y})$  holds. First solve for  $\mathbf{x}$ :

$$\begin{aligned} \frac{\partial H}{\partial \mathbf{x}} &= 0 \\ 2\mathbf{x}^T + \lambda^T \mathbf{A} &= 0 \\ \mathbf{x} &= \frac{1}{2} \mathbf{A}^T \lambda \end{aligned} \quad (6)$$

Now using the fact that  $\mathbf{A}\mathbf{A}^T$  is invertible, choose  $\lambda$  to ensure that that the original equation holds:

$$\begin{aligned} \mathbf{A}\mathbf{x} &= \mathbf{y} \\ \mathbf{A} \left( \frac{1}{2} \mathbf{A}^T \lambda \right) &= \mathbf{y} \\ \lambda &= 2 (\mathbf{A}\mathbf{A}^T)^{-1} \mathbf{y} \end{aligned} \quad (7)$$

Finally, substitute Equation 7 into Equation 6 to get an expression for  $\mathbf{x}$ :

$$\mathbf{x} = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}\mathbf{y} \quad (8)$$

### 3.3 The right pseudo-inverse

The  $M \times N$  matrix which pre-multiplies  $\mathbf{y}$  in Equation 8 is called the “right pseudo-inverse of  $\mathbf{A}$ ”:

$$\mathbf{A}_R^+ = \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1}.$$

Why the strange name? Because

$$\mathbf{A}\mathbf{A}_R^+ = \mathbf{A}\mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1} = \mathbf{I},$$

but  $\mathbf{A}_R^+\mathbf{A}$  is generally not equal to  $\mathbf{I}$ . ( $\mathbf{A}_R^+\mathbf{A} = \mathbf{I}$  iff  $\mathbf{A}$  is square and invertible, in which case  $\mathbf{A}_R^+ = \mathbf{A}^{-1}$ ).

**Fact 4** Let  $\mathbf{A}$  be an  $N \times M$  matrix,  $N < M$ , with full row-rank. Then the pseudo-inverse of  $\mathbf{A}$  projects a vector from the range of  $\mathbf{A}$  ( $= \mathbb{R}^N$ ) into the  $N$ -dimensional sub-space of  $\mathbb{R}^M$  spanned by the columns of  $\mathbf{A}$ :

$$\begin{aligned} \mathbf{A}_R^+ &= \mathbf{A}^T(\mathbf{A}\mathbf{A}^T)^{-1} \\ &= \left( \mathbf{V} \begin{bmatrix} \boldsymbol{\Sigma}_N \\ \mathbf{0} \end{bmatrix} \mathbf{U}^T \right) (\mathbf{U}\boldsymbol{\Sigma}_N^{-2}\mathbf{U}^T) \\ &= \mathbf{V} \begin{bmatrix} \boldsymbol{\Sigma}_N \\ \mathbf{0} \end{bmatrix} \boldsymbol{\Sigma}_N^{-2}\mathbf{U}^T \\ &= \mathbf{V} \begin{bmatrix} \boldsymbol{\Sigma}_N^{-1} \\ \mathbf{0} \end{bmatrix} \mathbf{U}^T \\ &= \sum_{i=1}^N \sigma_i^{-1} \mathbf{v}_i \mathbf{u}_i^T \end{aligned} \quad (9)$$

One should compare the sum of outer products in Equation 9 with those describing  $\mathbf{A}$  in Equation 4. This comparison drives home the geometric interpretation of the action of  $\mathbf{A}_R^+$ .

Fact 4 means that the solution to the regularized problem of Equation 5,  $\mathbf{x} = \mathbf{A}_R^+\mathbf{y}$ , defines the unique solution  $\mathbf{x}$  that is completely orthogonal to the null-space of  $\mathbf{A}$ . This should make good sense: in Section 3.1 we found that any component of the solution that lies in the null-space is irrelevant, and the problem defined in Equation 5 was to find the “smallest” solution vector.

Finally then, we can write an explicit expression for the complete space of solutions to  $\mathbf{A}\mathbf{x} = \mathbf{y}$ , for  $\mathbf{A}$ ,  $N \times M$ , with full row-rank:

$$\mathbf{x} = \mathbf{A}_R^+\mathbf{y} + \sum_{i=N+1}^M \alpha_i \mathbf{v}_i, \quad \text{for any } \{\alpha_i\}.$$

**Example** Consider one last time the equation  $\begin{bmatrix} 1 & 1 \end{bmatrix} \mathbf{x} = 4$ . In this case, the cost-function of Section 3.2 is  $x_1^2 + x_2^2$ . Inspection shows that the optimal solution is  $x_1 = x_2 = 2$ .

We can confirm this using Equation 8. The right pseudo-inverse of  $\mathbf{A}$  is

$$\mathbf{A}_R^+ = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \left( \begin{bmatrix} 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \right)^{-1} = \begin{bmatrix} .5 \\ .5 \end{bmatrix},$$

so the solution is  $\mathbf{x} = \begin{bmatrix} .5 \\ .5 \end{bmatrix} 4 = \begin{bmatrix} 2 \\ 2 \end{bmatrix}$ .

## 4 Linear Regression as Over-constrained Linear Algebraic Equations

We now consider the over-constrained case, where there are more equations than unknowns ( $N > M$ ). Although the results presented are generally applicable, for discussion purposes, we focus on a particular common application: linear regression.

Consider an experimental situation in which you measure the value of some quantity,  $y$ , which you believe depends on a set of  $M$  predictor variables,  $\mathbf{z}$ . For example,  $y$  could be number of spikes recorded from a sensory neuron in response to a stimulus with parameters  $\mathbf{z}$ . Suppose that you have a hypothesis that  $y$  depends linearly on elements of  $\mathbf{z}$ :  $y = b_1 z_1 + \dots + b_M z_M = \mathbf{z}^T \mathbf{b}$ , and you want to find the “best” set of model coefficients,  $\mathbf{b}$ . The first step is to repeat the experiment  $N > M$  times, yielding a data set  $\{y_i, \mathbf{z}_i\}_{i=1}^N$ .

In an ideal, linear, noise-free world, the data would satisfy the set of  $N$  linear equations,

$$\mathbf{z}_i^T \mathbf{b} = y_i, \quad i \in [1, N].$$

These equations can be rewritten in matrix form:

$$\begin{aligned} \begin{bmatrix} \mathbf{z}_1^T \\ \vdots \\ \mathbf{z}_N^T \end{bmatrix} [\mathbf{b}] &= \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \\ \mathbf{Z} \quad \mathbf{b} &= \mathbf{y} \\ \text{NxM} \quad \text{Mx1} & \quad \text{Nx1} \quad (\text{N} > \text{M}) \end{aligned} \tag{10}$$

Of course for any real data, no exact solution exists. Rather, we will look for the model parameters  $\mathbf{b}$  that give the smallest summed squared model-error:

$$\text{Find the vector } \mathbf{b} \text{ which minimizes } E = \sum_{i=1}^N (\mathbf{z}_i^T \mathbf{b} - y_i)^2 = (\mathbf{Z}\mathbf{b} - \mathbf{y})^T (\mathbf{Z}\mathbf{b} - \mathbf{y}). \tag{11}$$

**Definition 4** The “column-rank” of an  $N \times M$  matrix is the dimension of the subspace of  $\mathbb{R}^N$  spanned by its  $M$  rows. A matrix is said to have “full column-rank” if its column-rank is  $M$ , i.e. if the columns are a linearly independent set. Clearly this can only be the case if  $N \geq M$ .

The column-rank of a matrix is equal to its row-rank and its rank, and all three equal the number of non-zero SVs.

**Fact 5** If  $\mathbf{Z}$ ,  $N \times M$ ,  $N \geq M$ , has full column-rank, then  $\mathbf{Z}^T \mathbf{Z}$  is invertible.

This fact follows directly from Fact 3, with  $\mathbf{Z}^T$  replacing  $\mathbf{A}$ . From the previous result, we have that if

$$\mathbf{Z} = \mathbf{U} \begin{bmatrix} \Sigma_N \\ 0 \end{bmatrix} \mathbf{V}^T,$$

then

$$\begin{aligned} \mathbf{Z}^T \mathbf{Z} &= \mathbf{V} \Sigma_N^2 \mathbf{V}^T, \text{ and} \\ (\mathbf{Z}^T \mathbf{Z})^{-1} &= \mathbf{V} \Sigma_N^{-2} \mathbf{V}^T. \end{aligned}$$

We will now solve the problem stated in Equation 11. Note that the route to the solution will make use of Fact 5, and so the solution will only exist if  $\mathbf{Z}$  has full column-rank. In the case of

regression, this shouldn't be a problem as long as the data are independently collected and  $N$  is sufficiently larger than  $M$ .

We begin by rewriting the error,

$$E = (\mathbf{Z}\mathbf{b} - \mathbf{y})^T(\mathbf{Z}\mathbf{b} - \mathbf{y}) = \mathbf{b}^T\mathbf{Z}^T\mathbf{Z}\mathbf{b} - 2\mathbf{y}^T\mathbf{Z}\mathbf{b} + \mathbf{y}^T\mathbf{y}.$$

Following Section 1.2, we find the minimum-error  $\mathbf{b}$  by solving the equation  $\partial E/\partial \mathbf{b} = 0$ :

$$\begin{aligned} \frac{\partial E}{\partial \mathbf{b}} &= 2\mathbf{b}^T\mathbf{Z}^T\mathbf{Z} - 2\mathbf{y}^T\mathbf{Z} = 0 \\ \mathbf{Z}^T\mathbf{Z}\mathbf{b} &= \mathbf{Z}^T\mathbf{y} \\ \mathbf{b} &= (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{y} \end{aligned} \tag{12}$$

Equation 12 is the familiar formula for linear regression, which will be derived in a more standard way later in this course.

The  $M \times N$  matrix which pre-multiplies  $\mathbf{y}$  in Equation 12 is called the left pseudo-inverse of  $\mathbf{Z}$ :

$$\mathbf{Z}_L^+ = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T$$

**Fact 6** Let  $\mathbf{Z}$  be a matrix,  $N \times M$ ,  $N > M$ , with full column-rank. Then the pseudo-inverse of  $\mathbf{Z}$  projects a vector  $\mathbf{y} \in \mathbb{R}^N$  into  $\mathbb{R}^M$  by discarding any component of  $\mathbf{y}$  which lies outside (i.e. orthogonal to) the  $M$ -dimensional subspace of  $\mathbb{R}^N$  spanned by the columns of  $\mathbf{Z}$ :

$$\begin{aligned} \mathbf{Z}_L^+ &= (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T \\ &= (\mathbf{V}\Sigma_N^{-2}\mathbf{V}^T)(\mathbf{V}[\Sigma_N \ \mathbf{0}]\mathbf{U}^T) \\ &= \mathbf{V}\Sigma_N^{-2}[\Sigma_N \ \mathbf{0}]\mathbf{U}^T \\ &= \mathbf{V}[\Sigma_N^{-1} \ \mathbf{0}]\mathbf{U}^T \\ &= \sum_{i=1}^N \sigma_i^{-1} \mathbf{v}_i \mathbf{u}_i^T \end{aligned}$$