Derivation of the generalized rank annihilation method (GRAM)

Klaas Faber, Chemometry Consultancy

Introduction

The generalized rank annihilation method (GRAM) is a method for curve resolution and calibration that requires only a single calibration sample to obtain the so-called second-order advantage, i.e., it can determine the analyte of interest in the presence of unsuspected interferences [1]. GRAM is only one of many competing methods that can be applied to this particular data set-up. However, GRAM stands out for two reasons. First, its error analysis is extensively studied, see e.g. [2,3]. Second, GRAM has been reported to compare favorably with alternating least squares (ALS), alternating trilinear decomposition (ATLD), alternating coupled vectors resolution (ACOVER), alternating slice-wise diagonalization (ASD), alternating coupled matrices resolution (ACOMAR), self-weighted alternating trilinear decomposition (SWATLD) and pseudo alternating least squares (PALS), see [4]. Currently, no useful generalization of GRAM to multiple calibration samples seems to exist [5], however.

Several derivations of GRAM are published in the chemometrics literature, but they tend to be complicated. The remainder of this note consists of (1) a derivation that requires fewer steps and (2) a discussion that intends to shed some light on the properties of the solution.

Derivation

Consider the simplest case where the calibration and prediction sample have the same constituents. This situation could arise from performing standard additions for the analytes of interest [1]. The model for the data matrices can be written as

$$\mathbf{R}_{\mathrm{cal}} = \mathbf{X}\mathbf{\Lambda}\mathbf{Y}^{\mathrm{T}} \qquad (1a)$$

$$\mathbf{R}_{\mathrm{pred}} = \mathbf{X}\mathbf{Y}^{\mathrm{T}} \qquad (1b)$$

where the subscripts 'cal' and 'pred' refer to the particular sample, the columns of $\mathbf{X}$ and $\mathbf{Y}$ contain the loadings in the first and second mode, respectively, and $\mathbf{\Lambda} = \mathbf{C}_{\mathrm{pred}}^{-1}\mathbf{C}_{\mathrm{cal}}$ is a diagonal matrix with relative amounts (the diagonal elements of $\mathbf{C}_{\mathrm{cal}}$ and $\mathbf{C}_{\mathrm{pred}}$ are proportional to amounts). Throughout, the errors in the data have been dropped from the equations to simplify the presentation. In addition, it will be assumed that the noiseless data and true parameters are real-valued, i.e., not complex-valued. To remove scale indeterminacies, which in a formal sense constitute trivial non-identifiabilities (of model parameters), the columns of $\mathbf{Y}$ can be normalized in some convenient way and the columns of **X** scaled accordingly. For ‘spectro-chromatographic’ data, e.g. HPLC-UV or GC-MS, the columns of **X** and **Y** would contain the pure-component chromatograms and spectra, respectively (see Figure 1). Finally, it is noted that the model amounts to two equations with three unknowns, i.e., **X**, **Y** and $\mathbf{\Lambda} = \mathbf{C}_{\text{pred}}^{-1}\mathbf{C}_{\text{cal}}$.

It is often stated that GRAM requires that (1) **X** and **Y** be full column rank matrices and (2) the diagonal elements of $\mathbf{\Lambda}$ be unique. However, it is important to note that these conditions need only hold for the analytes of interest. If, for example, the prediction sample is obtained by performing standard additions, then the (true, not necessarily the estimated) relative amounts for the remaining constituents will be identical. Consequently, GRAM will not work for these ‘background’ constituents, but this hardly invalidates the approach as such for the analytes of interest.

The first step of GRAM consists of calculating base vectors for the (relevant) column and row space using the singular value decomposition (SVD),

$$\mathbf{R}_{\text{pred}} = \mathbf{U\Theta V}^{\text{T}} \tag{2}$$

where **U** and **V** span the column and row space, respectively, and $\mathbf{\Theta}$ is a diagonal matrix that contains the (positive) singular values in descending order. These base vectors are determined up to an unknown (full rank) transformation matrix **T**, because it holds as well that

$$\mathbf{R}_{\text{pred}} = \mathbf{U\Theta T T}^{-1}\mathbf{V}^{\text{T}} \tag{3}$$

Note that **T** is often referred to as a rotation matrix in the literature on curve resolution, but this is not true in a strict sense since a rotation matrix is orthogonal and, in general, $\mathbf{T}^{-1} \neq \mathbf{T}^{\text{T}}$. Comparing (1b) and (3) shows that the loadings can be expressed as

$$\begin{aligned} \mathbf{X} &= \mathbf{U\Theta T} \\ \mathbf{Y}^{\text{T}} &= \mathbf{T}^{-1}\mathbf{V}^{\text{T}} \end{aligned} \tag{4}$$

Inserting (4) in (1a) gives

$$\mathbf{R}_{\text{cal}} = \mathbf{U\Theta T\Lambda T}^{-1}\mathbf{V}^{\text{T}}, \tag{5}$$

It is seen that the problem is reduced to a single equation with two unknowns, i.e., **T** and $\mathbf{\Lambda} = \mathbf{C}_{\text{pred}}^{-1}\mathbf{C}_{\text{cal}}$ since **U**, **V** and $\mathbf{\Theta}$ are known from the SVD of $\mathbf{R}_{\text{pred}}$. Equation (5) can be re-arranged by pre-and post-multiplication with $\mathbf{\Theta}^{-1}\mathbf{U}^{\text{T}}$ and $\mathbf{VT}$, respectively, as

$$\left(\mathbf{\Theta}^{-1}\mathbf{U}^{\text{T}}\mathbf{R}_{\text{cal}}\mathbf{V}\right)\mathbf{T} = \mathbf{T\Lambda}, \tag{6}$$

since the orthogonality of the singular vectors implies that $\mathbf{\Theta}^{-1}\mathbf{U}^{\text{T}}\mathbf{U\Theta} = \mathbf{\Theta}^{-1}\mathbf{I\Theta} = \mathbf{I}$ and $\mathbf{V}^{\text{T}}\mathbf{V} = \mathbf{I}$, with **I** a suitably dimensioned identity matrix.

Discussion

Equation (6) is a standard (as opposed to generalized) real eigenvalue problem, which, in general, is not symmetrical. It stands to reason that the properties of such an eigenvalue problem carry over to the GRAM solution. For example, eigenvectors associated with real eigenvalues can be selected to be real, see e.g. [6]. This is certainly a convenient choice when **T** is used to estimate the (real) loadings according to Equation (4). Matlab, however, uses the Euclidean norm to fix the length of the eigenvectors. (Since one has a single equation with two unknowns, only their direction can be determined.) As a result, the eigenvectors are 'fixed' up to a complex multiplier of modulus unity, whereas the ∞-norm would have given a unique (real-valued) result [6]. (The routines in the old FORTRAN libraries use the ∞-norm.) Complex eigenvalues form another complication. They can, however, not be obtained for constituents that would have given a real eigenvalue in absence of noise, unless the model is seriously mis-specified. This follows directly from perturbation analysis: a (random) perturbation of the input data matrices resulting from real-valued noise leads to a real-valued perturbation of the 'true' eigenvalues, see e.g. [6]. Fortunately, Matlab has a built-in function (`cdf2rdf`) that converts the complex part of the solution into a form that provides valid input for Equation (4), see [7] for further discussion.

References

[1] E. Sánchez, B.R. Kowalski, Generalized rank annihilation factor analysis, *Analytical Chemistry*, **58** (1986) 496-499.

[2] N.M. Faber, J. Ferré and R. Boqué, Iteratively reweighted generalized rank annihilation method. 1. Improved handling of prediction bias, *Chemometrics and Intelligent Laboratory Systems*, **55** (2001) 67-90.

[3] N.M. Faber, R. Boqué and J. Ferré, Iteratively reweighted generalized rank annihilation method. 2. Least squares property and variance expressions, *Chemometrics and Intelligent Laboratory Systems*, **55** (2001) 91-100.

[4] N.M. Faber, Towards a rehabilitation of the generalized rank annihilation method (GRAM), *Analytical and Bioanalytical Chemistry*, **372** (2002) 683-687.

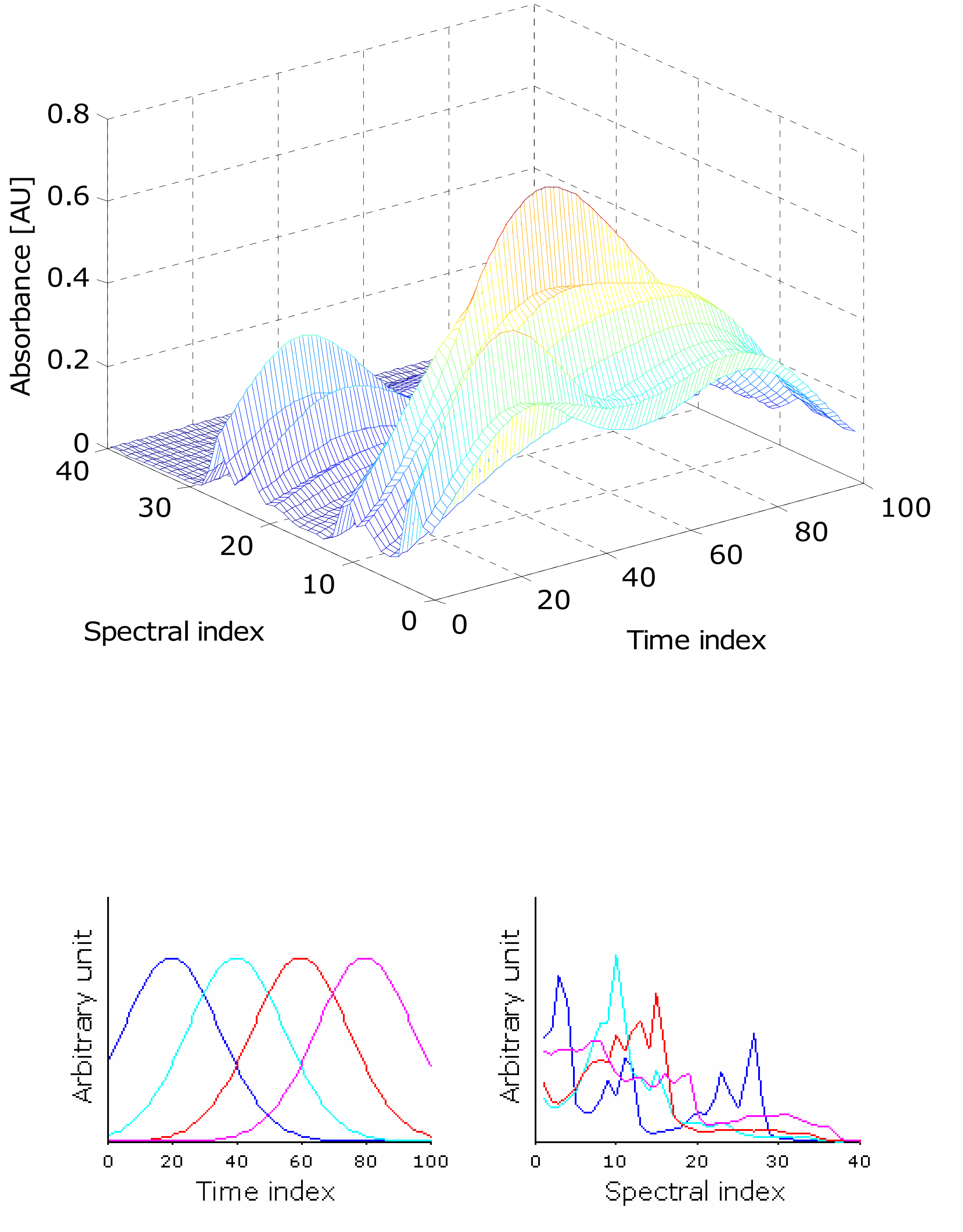
[5] N.M. Faber, R. Bro and P.K. Hopke, Recent developments in CANDECOMP/PARAFAC algorithms: A critical review, *Chemometrics and Intelligent Laboratory Systems*, **65** (2003) 119-137.

[6] J.H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon, Oxford (1988).

[7] K. Faber, On solving generalized eigenvalue problems using Matlab, *Journal of Chemometrics*, **11** (1997) 87-91.

Figure 1

Simulated HPLC-UV data matrix **R** (top) and pure-component chromatograms **X** and spectra **Y** (bottom).


Absorbance [AU]
Spectral index
Time index
Arbitrary unit
Time index
Arbitrary unit
Spectral index