TQ Analyst Algorithms: ValPro for Antaris

Thermo SCIENTIFIC

The information in this publication is provided for reference only. All information contained in this publication is believed to be correct and complete. Thermo Fisher Scientific shall not be liable for errors contained herein nor for incidental or consequential damages in connection with the furnishing, performance or use of this material. All product specifications, as well as the information contained in this publication, are subject to change without notice.

This publication may contain or reference information and products protected by copyrights or patents and does not convey any license under our patent rights, nor the rights of others. We do not assume any liability arising out of any infringements of patents or other rights of third parties.

We make no warranty of any kind with regard to this material, including but not limited to the implied warranties of merchantability and fitness for a particular purpose. Customers are ultimately responsible for validation of their systems.

For technical assistance, please contact:
Technical Support
Thermo Fisher Scientific
5225 Verona Road
Madison, WI 53711-4495
U.S.A.

Telephone: 1 800 532 4752 (U.S.A.) or +1 608 273 5017 (worldwide)
Fax: +1 608 273 5045 (worldwide)
E-mail: us.techsupport.analyze@thermofisher.com
World Wide Web: http://www.thermo.com/spectroscopy

269-124500, Rev B

Contents

This page intentionally left blank

Introduction

This manual describes the TQ Analyst™ algorithms used by the RESULT™ and ValPro™ software packages. You can use the algorithm descriptions along with the provided algorithm verification spreadsheets to verify the results produced by any TQ Analyst method executed alone or in conjunction with RESULT or ValPro. The descriptions in this manual are more detailed versions of the corresponding algorithm descriptions included in the spreadsheets.

Verifying a TQ Analyst method externally may be required to demonstrate regulatory compliance in the validation of your Thermo Scientific Industrial Solutions system. The spreadsheets let you reproduce the calculations performed on a spectrum when it is quantified by a TQ Analyst method. The results you obtain for a particular test using a spreadsheet should be mathematically equivalent to the results obtained using the corresponding algorithm qualification test in ValPro. If the results are different, see "Verifying an algorithm with your own data" for more information. See the "Using the Spreadsheets" chapter for instructions for opening and viewing a spreadsheet for a particular algorithm. See "Algorithm Qualification Test Descriptions" and "Running ValPro Qualification Tests" in the "Operation Qualification (OQ)" section of your *ValPro System Qualification* manual for more information about the ValPro tests.

Note To read the spreadsheets, you need Microsoft® Excel 97 or 2000 (or a later version). ▲

This manual covers the algorithms used by the quantitative analysis types listed in the following table. For each analysis type the table shows the section in the "Algorithms" chapter that contains a detailed description of the algorithm, and a reference to the section in the "Principles of TQ Analyst" chapter of the *TQ Analyst User's Guide* that describes the corresponding calibration or classification technique. See the "Parameter Settings" chapter for a description of how various parameter settings in TQ Analyst affect the calculations performed during a method calibration or prediction.

Analysis Type	*Section in "Algorithms" Chapter*	*Section in* TQ Analyst User's Guide
Simple Beer's law	"Simple Beer's law calibration and prediction"	"Simple Beer's Law calibration technique"
Classical least squares (CLS)	"Classical least squares (CLS) method for multivariate calibration and prediction"	"Classical least squares calibration technique"
Stepwise multiple linear regression (SMLR)	"Inverse least squares (ILS) method for multivariate calibration and prediction"	"Stepwise Multiple Linear Regression calibration technique"
Partial least squares (PLS)	"Partial least squares (PLS) method for multivariate calibration"	"Partial least squares calibration technique"
Principal component regression (PCR)	"Principal component regression (PCR) method for multivariate calibration"	"Principal component regression calibration technique"
Similarity match	"Similarity match for material identification"	"Similarity Match classification technique"
Distance match	"Distance match for material identification"	"Distance Match classification technique"
Discriminant analysis	"Discriminant analysis method for classifying materials"	"Discriminant Analysis classification technique"
Search standards	"Search method for identifying materials"	"Search Standards classification technique"
QC Compare search	"QC Compare method for classifying materials"	"QC Compare search classification technique"
Measurement only	"Measurement only method for measuring spectral features"	"Calibration models for Measurement Only methods"

Using the Spreadsheets

This chapter explains how to open and view the provided algorithm verification spreadsheets to verify the results produced by TQ Analyst methods. The table below shows the algorithm qualification tests available in ValPro, descriptions of the tests, the corresponding algorithms described in this manual, and the filenames of the corresponding spreadsheets. The ValPro tests appear in the table in the same order as in the Algorithm Qualification Report produced using your system.

ValPro Test	*Algorithm*	*Spreadsheet Filename*
Polystyrene Beers Law1 – Measures thickness (in sheets) of polystyrene sample by Beer's law using large peak in spectrum. Expected result is 2.9267.	Simple Beer's law	ALG_BEERS.xls
Polystyrene Beers Law2 – Measures thickness (in sheets) of polystyrene sample by Beer's law using small peak in spectrum. Expected result is 2.9736.	Simple Beer's law	ALG_BEERS2.xls
Polystyrene CLS – Measures thickness (in sheets) of polystyrene sample using classical least squares algorithm using spectrum region. Expected result is 0.9640.	Classical least squares (CLS)	ALG_CLS.xls
Polystyrene CLS-C – Measures standard error of thickness of polystyrene sample using classical least squares algorithm using spectrum region. Expected result is 0.0006.	Classical least squares (CLS): error term	ALG_CLS.xls
Polystyrene SMLR0 – Measures thickness (in sheets) of polystyrene sample by stepwise multiple linear regression performing analysis on spectrum data. Expected result: 2.9531.	Inverse least squares (ILS)	ALG_SMLR0.xls
Polystyrene SMLR1 – Measures thickness (in sheets) of polystyrene sample by stepwise multiple linear regression performing analysis on first derivative of spectrum data. Expected result: 2.9796.	Inverse least squares (ILS)	ALG_SMLR1.xls

(continued on next page)

ValPro Test	*Algorithm*	*Spreadsheet Filename*
Polystyrene SMLR2 – Measures thickness (in sheets) of polystyrene sample by stepwise multiple linear regression performing the analysis on second derivative of spectrum data. Expected result: 2.9876.	Inverse least squares (ILS)	ALG_SMLR2.xls
Polystyrene PLS0 – Measures thickness (in sheets) of polystyrene sample by partial least squares performing analysis on spectrum data. Expected result: 2.9236.	Partial least squares (PLS)	ALG_PLS0.xls
Polystyrene PLS0-C – Measures uncertainty in thickness of polystyrene sample by partial least squares performing analysis on spectrum data. Expected result: 0.0810.	Partial least squares (PLS): error term	ALG_PLS0.xls
Polystyrene PLS-S – Measures thickness (in sheets) of polystyrene sample by partial least squares performing analysis on Savitsky-Golay smoothed spectrum. Expected result: 2.9671.	Partial least squares (PLS) (with Savitzky-Golay smoothing)	ALG_PLSS.xls
Polystyrene PLS-N1 – Measures thickness (in sheets) of polystyrene sample by partial least squares performing analysis on Norris first derivative spectrum data. Expected result: 2.9653.	Partial least squares (PLS) (with Norris first derivative)	ALG_PLSN1.xls
Polystyrene PLS-N2 – Measures thickness (in sheets) of polystyrene sample by partial least squares performing analysis on Norris second derivative spectrum data. Expected result: 2.9598.	Partial least squares (PLS) (with Norris second derivative)	ALG_PLSN2.xls
PCR – Measures thickness (in sheets) of polystyrene sample by principal component regression. Expected result: 0.9988.	Principal component regression (PCR)	ALG_PCR.xls
PCR-C – Evaluates uncertainty of thickness (in sheets) of polystyrene sample by principal component regression. Expected result: 0.0477.	Principal component regression (PCR): error term	ALG_PCR.xls

(continued on next page)

ValPro Test	*Algorithm*	*Spreadsheet Filename*
PCR-F – Evaluates full spectrum check value computed when measuring thickness of polystyrene sample by principal component regression. Expected result: 99.9906.	Fit values for spectrum evaluation	ALG_PCRFullFit.xls
PCR-R – Evaluates region check value computed when measuring thickness of polystyrene sample by principal component regression. Expected result: 99.9999.	Fit values for spectrum evaluation	ALG_PCRRegFit.xls
Similarity Match – Measures similarity index for polystyrene sample using similarity match algorithm. Expected result: 93.8080.	Similarity match	ALG_SIMIL.xls
Distance Match – Measures class similarity for polystyrene sample using distance match algorithm. Spectrum is preprocessed with Norris second derivative. Expected result: 1.3468.	Distance match	ALG_DM.xls
Distance Match-C – Measures best class or category for polystyrene sample using distance match algorithm. Spectrum is preprocessed with Norris second derivative. Expected class index result: 0.0000.	Distance match	ALG_DM.xls
Distance Match-D – Measures maximum deviation (in unit of standard deviations) between any standard and polystyrene sample using distance match algorithm. Spectrum is preprocessed with Norris second derivative. Expected result: 14.4369.	Distance match	ALG_DM.xls
Discriminant – Measures class similarity for polystyrene sample using discriminant analysis algorithm. Spectrum is Savitzky-Golay smoothed. Expected result: 1.6260.	Discriminant analysis	ALG_DA.xls

(continued on next page)

ValPro Test	*Algorithm*	*Spreadsheet Filename*
Discriminant-M – Measures best class or category for polystyrene sample using discriminant analysis algorithm. Spectrum is Savitzky-Golay smoothed. Expected class index result: 2.0000.	Discriminant analysis	ALG_DA.xls
QC Compare Search – Determines unitless search metric polystyrene spectrum using QC Compare search algorithm. Expected result: 99.9789.	QC Compare	ALG_QCC.xls
QC Compare Search-C – Determines best class assigned for polystyrene sample using QC Compare search algorithm. Expected class index result: 1.0000.	QC Compare	ALG_QCC.xls
Search – Determines search metric for polystyrene sample using Search algorithm. Expected result: 99.9981.	Search	ALG_SEA.xls
Search-C – Determines most similar spectral library entry for polystyrene sample using Search algorithm. Expected class index result: 3.0000.	Search	ALG_SEA.xls
Search-F – Determines full spectrum check result calculated when analyzing polystyrene sample using Search algorithm. Expected result: 92.6164.	Fit values for spectrum evaluation	ALG_SEAFullFit.xls
Fixed Location Height – Measures ratios of heights of two absorbance peaks at fixed locations in polystyrene spectrum. Expected result: 0.2834.	Measurement only: fixed location height	ALG_MO_FIXED.xls
Average Height in Range – Measures average height of absorbance peak over range in polystyrene spectrum. Expected result: 1.8934.	Measurement only: average height in range	ALG_MO_AVGHGT.xls

(continued on next page)

ValPro Test	*Algorithm*	*Spreadsheet Filename*
Maximum Height in Range – Measures maximum height of absorbance peak over range in polystyrene spectrum. Expected result: 0.3891.	Measurement only: maximum height in range	ALG_MO_MAXHGT.xls
Minimum Height in Range – Measures minimum height of absorbance peak over range in polystyrene spectrum. The expected result is 0.1894.	Measurement only: minimum height in range	ALG_MO_MINHGT.xls
Absolute Maximum Height in Range – Measures absolute maximum height of absorbance peak over range in polystyrene spectrum. Expected result: 0.6989.	Measurement only: absolute maximum in range	ALG_MO_ABSHGT.xls
Area – Measures area of absorbance peak in polystyrene spectrum. Expected result: 29.8170.	Measurement only: area	ALG_MO_AREA.xls
Computed Area – Measures computed area of absorbance peak in polystyrene spectrum. Expected result: 67.7205.	Measurement only: computed area	ALG_MO_CAREA.xls
RMS Noise – Measures root mean square noise in milliabsorbance units over range in polystyrene spectrum. Expected result: 0.6949.	Measurement only: rms noise	ALG_MO_RMS.xls
Peak-to-Peak Noise – Measures peak-to-peak noise in milliabsorbance units over range in polystyrene spectrum. Expected result: 2.3458.	Measurement only: peak-to-peak noise	ALG_MO_PP.xls
Height at Exact Location – Measures interpolated height in absorbance units at exact location in polystyrene spectrum. Expected result: 0.1663.	Measurement only: interpolated height at exact location	ALG_MO_EXACT.xls
Peak Height – Measures interpolated peak height in absorbance units for peak in polystyrene spectrum. Expected result: 0.4302.	Measurement only: peak height (interpolated)	ALG_MO_INTERP.xls

(continued on next page)

ValPro Test	*Algorithm*	*Spreadsheet Filename*
Peak Location – Measures interpolated peak location for peak in polystyrene spectrum. Expected result in wavenumbers: 5948.235.	Measurement only: peak location (interpolated)	ALG_MO_LOCN.xls
Peak Width – Measures peak width at half maximum for peak in polystyrene spectrum. Expected result in wavenumbers: 116.542.	Measurement only: peak width (at half maximum)	ALG_MO_FWHM.xls
Single Beam 1% of Maximum – Measures location at which intensity is 1% of peak maximum in single-beam spectrum. Expected result in wavenumbers: 5622.465.	Measurement only: location at 1% of peak	ALG_MO_1PCT.xls
Single Beam 2% of Maximum – Measures location at which intensity is 2% of peak maximum in single-beam spectrum. Expected result in wavenumbers: 5676.673.	Measurement only: location at 2% of peak	ALG_MO_2PCT.xls
Single Beam 5% of Maximum – Measures location at which intensity is 5% of peak maximum in single-beam spectrum. Expected result in wavenumbers: 5742.054.	Measurement only: location at 5% of peak	ALG_MO_5PCT.xls
Single Beam 10% of Maximum – Measures location at which intensity is 10% of peak maximum in single-beam spectrum. Expected result in wavenumbers: 5794.285.	Measurement only: location at 10% of peak	ALG_MO_10PCT.xls

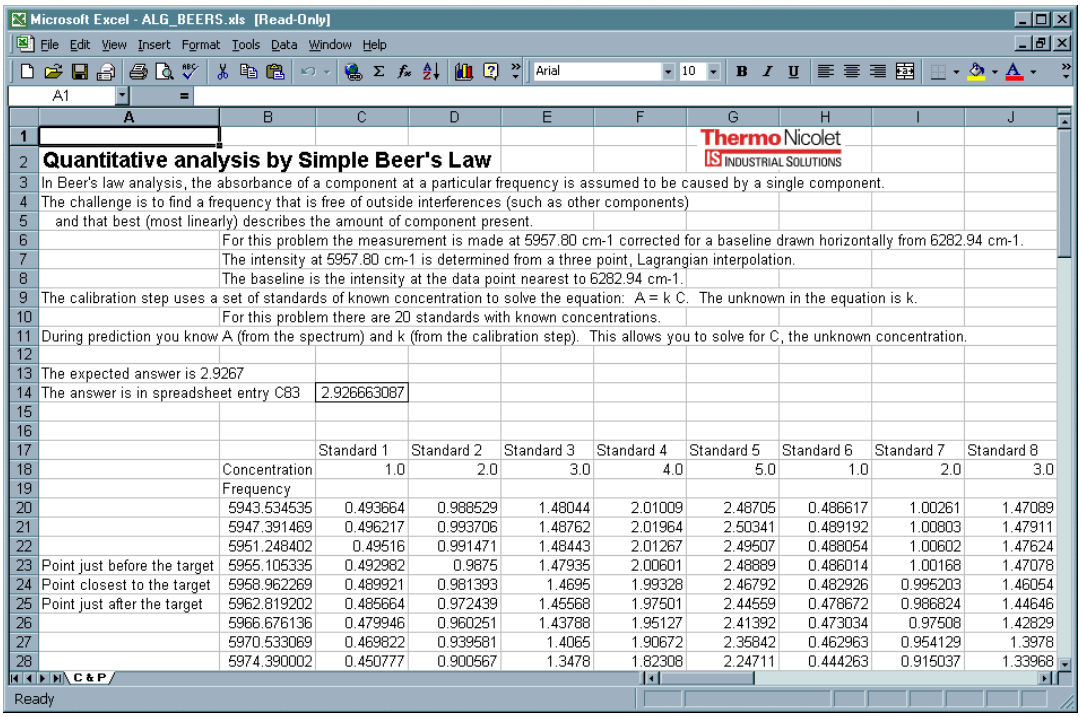
Each spreadsheet contains a description of the algorithm, data from a provided spectral data file, and the expected and actual results obtained for that data using the calculations specified by the algorithm.

To open and view a spreadsheet for the algorithm you want to verify, follow the steps below. The next section explains how to use the spreadsheets to verify an algorithm with your own data.

1. Start Microsoft Excel.

2. **Use Open in the File menu to open the desired spreadsheet.**

Here is an example of an opened spreadsheet:



Microsoft Excel - ALG_BEERS.xls [Read-Only]

File Edit View Insert Format Tools Data Window Help

Arial 10

A1 =

	A	B	C	D	E	F	G	H	I	J
1							Thermo Nicolet			
2	**Quantitative analysis by Simple Beer's Law**						INDUSTRIAL SOLUTIONS			
3	In Beer's law analysis, the absorbance of a component at a particular frequency is assumed to be caused by a single component.									
4	The challenge is to find a frequency that is free of outside interferences (such as other components)									
5	and that best (most linearly) describes the amount of component present.									
6		For this problem the measurement is made at 5957.80 cm-1 corrected for a baseline drawn horizontally from 6282.94 cm-1.								
7		The intensity at 5957.80 cm-1 is determined from a three point, Lagrangian interpolation.								
8		The baseline is the intensity at the data point nearest to 6282.94 cm-1.								
9	The calibration step uses a set of standards of known concentration to solve the equation: A = k C. The unknown in the equation is k.									
10		For this problem there are 20 standards with known concentrations.								
11	During prediction you know A (from the spectrum) and k (from the calibration step). This allows you to solve for C, the unknown concentration.									
12										
13	The expected answer is 2.9267									
14	The answer is in spreadsheet entry C83		2.926663087							
15										
16										
17			Standard 1	Standard 2	Standard 3	Standard 4	Standard 5	Standard 6	Standard 7	Standard 8
18		Concentration	1.0	2.0	3.0	4.0	5.0	1.0	2.0	3.0
19		Frequency								
20		5943.534535	0.493664	0.988529	1.48044	2.01009	2.48705	0.486617	1.00261	1.47089
21		5947.391469	0.496217	0.993706	1.48762	2.01964	2.50341	0.489192	1.00803	1.47911
22		5951.248402	0.49516	0.991471	1.48443	2.01267	2.49507	0.488054	1.00602	1.47624
23	Point just before the target	5955.105335	0.492982	0.9875	1.47935	2.00601	2.48889	0.486014	1.00168	1.47078
24	Point closest to the target	5958.962269	0.489921	0.981393	1.4695	1.99328	2.46792	0.482926	0.995203	1.46054
25	Point just after the target	5962.819202	0.485664	0.972439	1.45568	1.97501	2.44559	0.478672	0.986824	1.44646
26		5966.676136	0.479946	0.960251	1.43788	1.95127	2.41392	0.473034	0.97508	1.42829
27		5970.533069	0.469822	0.939581	1.4065	1.90672	2.35842	0.462963	0.954129	1.3978
28		5974.390002	0.450777	0.900567	1.3478	1.82308	2.24711	0.444263	0.915037	1.33968

C & P

Ready

At the top of the spreadsheet is a general description of the algorithm, including its purpose; for example, to determine the concentration of a component, or the noise level in a spectral region.

The expected and calculated answers are shown next. The closeness of these values is what verifies the algorithm. (See the Note following this procedure.) There may be more than one pair of values depending on the number of answers produced by the algorithm.

Depending on the complexity of the algorithm, and therefore the spreadsheet, the rest of the algorithm information may appear on the same page or on additional pages. (To see a page, click its tab at the bottom of the window.) This information includes the algorithm steps, the X and Y values of the spectral data points, and various calculation results. You may need to scroll horizontally or vertically to see all of the cells containing information. The calculated answer appears near the end of the information. This value and the calculated answer shown near the top of the spreadsheet are identical.

3. **When you are finished viewing the spreadsheet, close it or exit Excel.**

Note When you compare the results from the ValPro algorithm test with the results computed in the Excel spreadsheets, you will find that some of the values are slightly different. The three tests that produce a different result in fewer than four decimal digits are shown in the table below.

ValPro Test	*ValPro Result*	*Spreadsheet Result*
Distance Match-D	14.4369	14.3677
QC Compare Search	99.9805	99.97954
Search-F	92.6170	92.61628

The differences illustrated here are accounted for by a difference in machine floating point precision between the Excel calculation and the calculation in TQ Analyst. The magnitude of difference is a measure of the uncertainty in computing these values.

In the distance match technique, a value is computed at each frequency in an unknown spectrum. The value is computed by subtracting the average intensity at a particular frequency for a class of reference spectra from the intensity measured at that frequency in the unknown. This value is then divided by the standard deviation of the intensities of the reference spectra measured at the frequency. The result shown above is produced because the standard deviation at the specific frequency is small relative to the measured value. The small differences between the spectrum and the mean spectrum are magnified by the division process; therefore, the final results calculated by the spreadsheet and ValPro are different.

In the QC Compare technique, the search metric is scaled to enhance very subtle differences when it is close to 100. Because the search metric is raised to the twenty-first power, even small differences in the metric are magnified and the final results calculated by the spreadsheet and the ValPro results shown above are different.

The difference between the full spectrum check (Search-F) values is very small. The algorithm used to compute the metric requires that the first derivative of the spectrum be estimated. This is done by using a three-point quadratic derivative. Because the derivative is computed by subtracting two nearly equal values, the floating point precision of the difference is less than the precision of the original measurement. This causes the search metrics to be slightly different and the final results calculated by the spreadsheet and the ValPro results shown above to be different. ▲

Verifying an algorithm with your own data

The primary purpose of the algorithm verification spreadsheets is to let you independently confirm the algorithms used by TQ Analyst methods. However, you can also use these Microsoft Excel spreadsheets to help you verify a TQ Analyst method you developed using data collected on your spectrometer.

If you plan to verify a method and algorithm used with a software tool other than Excel, you must write the needed code based on the algorithm description given in this manual. You can use the associated Excel spreadsheet as a reference to verify any steps that are not clear.

If you plan to use Excel to verify a method and algorithm, you can use the associated spreadsheet as a template for the verification. Many of the steps in the spreadsheets are simple calculations where each cell is calculated independently. However, some operations are performed on a vector or matrix of data. These steps make it difficult to paste data into a new spreadsheet for instant verification.

Since most of the "measurement only" spreadsheets do not use vector or matrix calculations, they are good candidates for quickly verifying calculations.

If the answers you produce with your method do not match the result you got using TQ Analyst, check the following items.

1. First, make sure you have completely described the method used by TQ Analyst. For example, is a required baseline included in the spectral data? Does your method perform a derivative calculation specified in the TQ Analyst method?

2. If the answers differ by only a small amount in a late decimal place, there may be a rounding difference between the algorithm created in Excel (or another tool) and the TQ Analyst algorithm.

3. Any differences between the results obtained with TQ Analyst and your method will likely be small. Check to see if you are correcting the spectral data either with a linear or quadratic correction or by taking a derivative. Both these techniques may produce only slight deviations between TQ Analyst results and your own.

This page intentionally left blank

Algorithms

This chapter describes the algorithms used by the TQ Analyst and RESULT software packages. The algorithms are in sections named for the quantitative analysis types that use them. For information on parameter settings that affect these algorithms, see the "Parameter Settings" chapter.

Notation

The following notation conventions are used in the algorithm descriptions:

- Boldface, upper-case letters are used to denote matrices; for example, **X** and **Y**.
- Primes (') are used to denote transposed matrices.
- Boldface, lower-case letters are used to denote vectors; for example, **x** and $\mathbf{y_m}$.
- Scalars are shown in italics; for example, x_i and y_k.

Simple Beer's law calibration and prediction

The principle that drives the simple Beer's law model is that there is some feature in the spectrum (a measurement at a particular frequency) that is uniquely related to each component (concentration) to be measured. Beer's law states that the absorbance value, A_i, at frequency i is proportional to the concentration of the component, c, and is also proportional to the pathlength, b, through which the measurement is made. This model is expressed by the following equation, where a is the proportionality constant.

$A_i = abc$ (equation 1)

One simple enhancement to this model assumes that there is an offset to the model. That is, the absorbance measured when the concentration is zero does not equal zero. This offset changes the model as shown below.

$A_i = abc + A_0$ (equation 2)

When there are many standard spectra, these equations may be written as shown below.

$$\mathbf{A} = \mathbf{CK}$$

A, the absorbance measurement, is a column matrix of size s x 1 where s is the number of standards.

In the case of equation 1, **C**, the concentration values, is a column matrix of size s x 1 and **K** is a matrix of size 1 x 1. The value in the **K** matrix is the proportionality constant that relates the absorbance value to the concentration.

For equation 2, **C**, the concentration and baseline values, is a matrix of size s x 2 where the second column is all ones. **K** is a matrix of size 2 x 1. The first value in the **K** matrix is the proportionality constant that relates the absorbance value to the concentration. The second value in **K** is the offset term that describes the blank effect.

Calibration

The system is calibrated by solving the matrix equation shown above for the calibration matrix, **K**.

$$\mathbf{K} = (\mathbf{C'C})^{-1} \mathbf{C'A}$$

Note that in this step we invert (**C'C**), which is either a 1 x 1 or 2 x 2 matrix. Thus s, the number of reference standards, must be one only for the case in equation 1. However, we must have at least two standards if our solution includes an offset term.

Prediction

We have the calibration matrix to predict absorbance values from concentrations. However, in prediction we want to do just the opposite. We have measured a spectral intensity, A_i, and wish to estimate the concentration of the individual component.

In the case of equation 1, the **K** matrix contains only one term, K_1. In this case the unknown calibration is estimated as follows:

$$c = A_i / K_1$$

In the case of equation 2, the **K** matrix contains two terms. The first term, K_1, is the proportionality constant, and the second term, K_2, is the offset. The unknown concentration is estimated as follows:

$$c = (A_i - K_2) / K_1$$

Classical least squares (CLS) method for multivariate calibration and prediction

The CLS method is a quantitative analysis technique based on the least squares algorithm. CLS is typically used when each component of interest produces a peak or combination of peaks in the sample spectrum, but the peaks overlap significantly.

Data pretreatment

The matrix **X** contains the concentration data for a set of calibration standards. **X** is *r* x *c*, where there are *r* references and *c* components.

The matrix **Y** contains the spectral data for a set of calibration standards. **Y** is *r* x *p*, where there are *r* references and *p* spectral data points.

Setting up the data that will be used in the method

The classical least squares model states that the measured signal intensity (the **Y** values) is linearly proportional to the concentration values (the **X** values), as shown by this equation:

$$\mathbf{Y} = \mathbf{XC'} + \mathbf{E}$$

Y is the *r* x *p* spectral data point matrix, *r* references by *p* data points.

X is the *r* x *c* component matrix, *r* references by *c* components.

C is the *p* x *c* calibration matrix, *p* data points by *c* components.

E contains the errors (uncertainties) not explained by the model and is of size *r* x *p*.

Calibration

The system is calibrated by solving the matrix equation shown above for the calibration matrix, **C**.

$$\mathbf{C} = \mathbf{Y'X} \left(\mathbf{X'X} \right)^{-1}$$

Note that in this step we invert (**X'X**), which is a (*c* x *c*) matrix created by multiplying a (*c* x *r*) matrix by an (*r* x *c*) matrix. Therefore *r*, the number of references, must be greater than or equal to *c*, the number of components.

The calibration matrix, **C**, has *p* rows and *c* columns. Each column is an estimate of the pure component spectrum that is associated with a component in the mixture.

Prediction

We have the calibration matrix to predict absorbance values from concentrations. However, in prediction we want to do just the opposite. We have measured a spectrum and wish to estimate the concentrations. Therefore, we solve the calibration matrix equation for **x**, the set of concentrations that will best produce the unknown spectrum.

$$\mathbf{x} = \mathbf{yC} \left(\mathbf{C'C} \right)^{-1}$$

y is the vector that contains the unknown spectrum (of size 1 x *p*).

x is the vector that contains the estimate of the unknown concentrations (of size 1 x *c*).

Note that in this step we invert (**C'C**), which is a (*c* x *c*) matrix created by multiplying a (*c* x *p*) matrix by a (*p* x *c*) matrix. Therefore, *p*, the number of spectral data points, must be greater than or equal to *c*, the number of components.

In practice the set of terms [$\mathbf{C} \left(\mathbf{C'C} \right)^{-1}$], which is of size *p* x *c*, may be precalculated. This permits the prediction step to be done as a simple matrix multiplication.

Principal component regression (PCR) method for multivariate calibration

The PCR method is a quantitative analysis technique based on the principal component regression algorithm.

Data pretreatment

The matrix $\mathbf{Y_o}$ contains the concentration data for a set of calibration standards. $\mathbf{Y_o}$ is r x c, where there are r references and c components.

The matrix $\mathbf{X_o}$ contains the spectral data for a set of calibration standards. $\mathbf{X_o}$ is r x p, where there are r references and p spectral data points.

Several data pretreatments are possible. Two examples are described below.

In **mean centering** the column-wise mean (the average over all references for each data point) may be computed and subtracted from each element of the column. If the general terms are x_{ij}, the averages are x_j. x_j is subtracted from each x_{ij}.

In **variance scaling** the column-wise variance (the variance over all references for each data point) may be computed. The square root of the variance is used to scale each element of the column. This ensures that each column (data point) has equal weight before starting the analysis.

Setting up the data that will be used in the method

After the spectral data in $\mathbf{X_o}$ has been rescaled, the resultant data is called $\mathbf{X}$. $\mathbf{X}$ is the starting point for the principal component regression.

Calibration

The following steps are performed in a PCR calibration for dimensions $h = 1, 2, ..., a$:

1. Initialize a matrix $\mathbf{U}$ that has r rows and has $h = 0$ columns.

2. Increase h by 1 and select the column of $\mathbf{X}$ (actually $\mathbf{X_h}$ since it depends on the value of h) with the greatest sum of squares. This is a first estimate for the principal component scores (or latent variables). Call this vector $\mathbf{u_h}$. $\mathbf{u_h}$ is of size r x 1.

3. Compute the squared norm of $\mathbf{u_h}$.

   $u_h^2 = \mathbf{u_h}' \mathbf{u_h}$

4. Calculate the row vector $\mathbf{b_h}$ as $\mathbf{b_h}' = \mathbf{u_h}'\mathbf{X} / u_h^2$. $\mathbf{b_h}$ is of size p x 1.

5. Compute the squared norm of $\mathbf{b_h}$.

   $b_h^2 = \mathbf{b_h}'\mathbf{b_h}$

6. Scale $\mathbf{b_h}$ to unit length according to $\mathbf{b_h} = \mathbf{b_h} / b_h$.

7. Rename u_h^2 as u_{old}^2. Calculate a new estimate of the principal component scores. $\mathbf{u_h}$ is still of size r x 1.

   $\mathbf{u_h} = \mathbf{X}\mathbf{b_h} / b_h^2$

8. Recalculate the squared norm of $\mathbf{u_h}$.

   $u_h^2 = \mathbf{u_h}'\mathbf{u_h}$

9. If norm($\mathbf{u_h} - \mathbf{u_{old}}$) > 10^{-6} * norm($\mathbf{u_h}$), return to step 4 and continue. Otherwise, the scores have converged and step 10 is performed. When convergence has been reached, $\mathbf{u_h}$ equals the scores or latent variables and $\mathbf{b_h}$ equals the loadings for factor h.

10. Compute the part of $\mathbf{X}$ that is not described by this much of the model. The residual is calculated as shown below.

    $\mathbf{X_{h+1}} = \mathbf{E_x} = \mathbf{X_h} - \mathbf{u_h}\mathbf{b_h}'$

11. Add column h to the $\mathbf{U}$ matrix such that $\mathbf{U} = \mathbf{U} \mid \mathbf{u_h}$.

12. Return to step 2 until a factors (principal components) have been computed.

13. If $\mathbf{X_{a+1}}$ is small, $\mathbf{U}$ is a very good approximation for $\mathbf{X}$ except that a different, orthogonal basis set is being used. Replace $\mathbf{X}$ in the normal inverse regression model with $\mathbf{U}$ and assume that an inverse least squares relationship between $\mathbf{Y}$ and $\mathbf{U}$ exists. The model is defined by this equation:

    $\mathbf{Y} = \mathbf{U}\mathbf{C}' + \mathbf{E}$

    $\mathbf{Y}$ is the r x c component matrix, r references by c components.

U is the *r* x *a* spectral data point matrix, *r* references by *a* transformed data points.

C is the *c* x *a* calibration matrix, *c* components by *a* transformed data points.

E contains the "random" residuals not explained by the systematic model and is *r* x *c*.

Estimate **C'** as shown below.

$$\mathbf{C'} = (\mathbf{U'U})^{-1} \mathbf{U'Y}$$

Note that the inverse of the product (**U'U**) is taken. This is an (*a* x *a*) matrix created by multiplying an (*a* x *r*) matrix by an (*r* x *a*) matrix. This means that *r* must be greater than or equal to *a*. (There must be more references than there are transformed data points!)

Prediction

These steps are performed in a PCR prediction:

1. Scale and center **x** as indicated by the method. Note that the scaling and centering constants are taken from the calibration data. Create a vector, **u**, that has $h = 0$ elements.

2. Set $h = 0$ and set $\mathbf{y} = 0$.

3. Increase *h* by 1 and compute the following:

   $$u_h = \mathbf{x'b_h} / b_h^2$$

   This is the solution to the model $\mathbf{x} = u_h\mathbf{b_h}$.

4. Remove this contribution from **x**. This is done only so that an estimate of degree of fit may be determined. **x** may be removed since only orthogonal components are removed at any time.

   $$\mathbf{x} = \mathbf{x} - u_h\mathbf{b_h}$$

5. Augment the **u** vector with u_h.

   $$\mathbf{u} = \mathbf{u} \mid u_h$$

6. If $h > a$, continue to step 7. Otherwise, return to step 3.
7. Predict the concentrations according to this equation:

$$\mathbf{y} = \mathbf{uC'}$$

The predicted value(s) of **y** is (are) now available. They must be unscaled and uncentered as required.

8. Measure the degree of fit of the unknown sample as the variance of the residual spectrum, the **x** that remains after all *a* factors are removed.

$$s^2 = (\mathbf{x'x}) / (p - a)$$

If the fit is acceptable, $s^2 < s_x^2 (\mathbf{E_x})$ * F. This implies that the predicted value, **y**, can be reliably computed by this algorithm. $s_x^2 (\mathbf{E_x})$ is the residual variance of the $\mathbf{X}_{aopt+1}$ matrix (after removal of *a* factors). F is approximately the F statistic at probability level (alpha = 0.05) and (*p* - *a*) / 2 and (*p* - *a*) (*r* - *a* - 1) / 2 degrees of freedom. In general this is an optimistic estimate of the degrees of freedom.

Partial least squares (PLS) method for multivariate calibration

The PLS method is a statistical approach to quantitative analysis based on the partial least squares algorithm.

Data pretreatment

The matrix $\mathbf{Y_o}$ contains the concentration data for a set of calibration standards. $\mathbf{Y_o}$ is r x c, where there are r references and c components.

The matrix $\mathbf{X_o}$ contains the spectral data for a set of calibration standards. $\mathbf{X_o}$ is r x p, where there are r references and p spectral data points.

Several data pretreatments are possible. Two examples are described below.

In **mean centering** the column-wise mean (the average over all references for each component, or data point) may be computed and subtracted from each element of the column. If the general terms are x_{ij} and y_{ik}, the averages are $x_{.j}$ and $y_{.k}$. $x_{.j}$ is subtracted from each x_{ij}, and $y_{.k}$ is subtracted from each y_{ik}.

In **variance scaling** the column-wise variance (the variance over all references for each component, or data point) may be computed. The square root of the variance is used to scale each element of the column. This ensures that each column (component or data point) has equal weight before starting the analysis.

Setting up the data that will be used in the method

After the spectral data in $\mathbf{X_o}$ has been rescaled, the resultant data is called **X**. Similarly, $\mathbf{Y_o}$ becomes **Y**.

An inverse least squares relationship between **Y** and **X** is assumed to exist, giving the following equation for the model:

$$\mathbf{Y} = \mathbf{XC'} + \mathbf{E}$$

Y is the r x c component matrix, r references by c components.

X is the r x p spectral data point matrix, r references by p data points.

C is the c x p calibration matrix, c components by p data points.

E contains the "random" residuals not explained by the systematic model and is r x c.

Calibration

In the PLS1 method, the number of components, c, is assumed to be one. The algorithm is shown below.

The following steps are performed in a PLS1 calibration for dimensions $h = 1, 2, ..., a$:

1. Set $h = 0$.

2. Increase h by 1. Since the **Y** matrix has only one column, select **Y** matrix as the latent variable, or $\mathbf{u_h} = \mathbf{Y}$. $\mathbf{u_h}$ is of size r x 1.

3. Determine weights for the **X** variables as $\mathbf{w_h} = \mathbf{u_h}'\mathbf{X}$. Normalize $\mathbf{w_h}$ to unit length by dividing each element by the norm $\mathbf{w_h}\mathbf{w_h}'$. $\mathbf{w_h}$ is of size 1 x p. Also, **X** and **Y** are actually a function of h, so **X** is $\mathbf{X_h}$ and **Y** is $\mathbf{Y_h}$.

4. Compute the latent variable for **X** as $\mathbf{t_h} = \mathbf{X}\mathbf{w_h}'$. $\mathbf{t_h}$ is of size r x 1.

   We now have a latent variable from **Y** as $\mathbf{u_h}$ and a latent variable from **X** as $\mathbf{t_h}$. $\mathbf{u_h}$ is a (component) projection containing one value per reference, and $\mathbf{t_h}$ is also a projection (this time spectral) containing one value per reference. A model that relates $\mathbf{u_h}$ as a linear function of $\mathbf{t_h}$ is assumed to exist, giving the following equation:

   $$u_h = f(t_h)$$

   This model may be $u_h = v_h * t_h$ (a model of the form y = mx).

   The model may be more complicated; for example, $u_h = a_h t_h^3 + b_h t_h^2 + c_h t_h + d_h$ (a cubic model). This is the basis of the "Non-linear PLS" option.

5. Compute the coefficients by least squares (to estimate, for example, v_h or a_h, b_h, c_h, d_h).

6. Compute the loadings for **X** as $\mathbf{b_{hx}}$ from $\mathbf{b_{hx}} = \mathbf{t_h}'\mathbf{X} / (\mathbf{t_h}'\mathbf{t_h})$. $\mathbf{b_{hx}}$ is of size 1 x p.

7. Compute the part of **X** and **Y** that are not described by this much of the model. These residuals are calculated as shown below.

   $$\mathbf{X_{h+1}} = \mathbf{E_x} = \mathbf{X_h} - \mathbf{t_h}\mathbf{b_{hx}}$$

   $$\mathbf{Y_{h+1}} = \mathbf{E_y} = \mathbf{Y_h} - \mathbf{f(t_h)}$$

8. Return to step 2, unless the last value of *h* has been reached.

Optimum number of factors or dimensions

The number of significant dimensions, a_{opt}, is best determined by cross validation in which these steps are performed:

1. Remove a single calibration sample, or a set of calibration samples, from the set of calibration standards.

2. Recalibrate the system without this sample set.

3. For each *h* from 1 to *a*, predict the remaining sample using the new model.

4. Save the difference between the actual and calculated concentration(s). Perform this same procedure for each calibration sample set (so that as many as *r* calibrations are performed). For each *h* square and sum the concentration differences for all calibration standards.

5. Step through the *a* values for the predicted error sum of squares. If the resulting sum of squares is smaller than the previous value, that dimension has predictive relevance. Continue until you find a value $a_{opt} + 1$ that is not significant.

Prediction

These steps are performed in a PLS prediction:

1. Scale and center **x** as indicated by the method. Note that the scaling and centering constants are taken from the calibration data.

2. Set $h = 0$ and set $\mathbf{y} = \mathbf{y_{avg}}$. $\mathbf{y_{avg}}$ is the average value taken by y in the calibration data after scaling and centering.

3. Increase *h* by 1 and compute the following:

   $t_h = xw_h'$

   $\mathbf{y} = \mathbf{y} + f(t_h)\ \mathbf{b_{hy}}$

   ($\mathbf{b_{hy}}$ is unity if there is only one component!)

   $\mathbf{x} = \mathbf{x} - t_h\mathbf{b_{hx}}$

4. If $h > a_{opt}$, continue to step 5. Otherwise, return to 3.

5. The predicted value(s) of **y** is (are) now available. Unscale and uncenter it (them) as required.

6. Use the equation below to measure the degree of fit of the unknown sample as the variance of the residual spectrum, the **x** that remains after all a_{opt} factors are removed.

   $$s^2 = (\mathbf{x'x}) / (p - a_{opt})$$

   If the fit is acceptable, $s^2 < s_x^2 (\mathbf{E_x}) * F$. This implies that the predicted value, **y**, can be reliably computed by this algorithm. $s_x^2 (\mathbf{E_x})$ is the residual variance of the $\mathbf{X}_{aopt+1}$ matrix (after removal of a_{opt} factors). F is the F statistic at probability level (alpha = 0.05) and $(p - a_{opt}) / 2$ and $(p - a_{opt}) (r - a_{opt} - 1) / 2$ degrees of freedom. In general this is an optimistic estimate of the degrees of freedom.

References

W. Lindberg, J. Persson and S. Wold, *Analytical Chemistry*, Vol. 55, 1983, pp. 643-648.

Inverse least squares (ILS) method for multivariate calibration and prediction

This is the basic algorithm used for stepwise multiple linear regression (SMLR). However, this does not include the calibration step where the best data points are identified. In the case of SMLR, the number of components is one (at a time) and the number of data points is a small number (for each component).

Data pretreatment

The matrix **Y** contains the concentration data for a set of calibration standards. **Y** is *r* x *c*, where there are *r* references and *c* components.

The matrix **X** contains the spectral data for a set of calibration standards. **X** is *r* x *p*, where there are *r* references and *p* spectral data points.

Setting up the data that will be used in the method

The inverse least squares model states that the concentration values (the **Y** values) are a linear function of the spectral data (the **X** values), as shown by this equation:

$$\mathbf{Y} = \mathbf{XC'} + \mathbf{E}$$

Y is the *r* x *c* component matrix, *r* references by *c* components.

X is the *r* x *p* spectral data point matrix, *r* references by *p* data points.

C is the *c* x *p* calibration matrix, *c* components by *p* data points.

E contains the errors not explained by the model and is of size *r* x *c*.

Calibration

The system is calibrated by solving the matrix equation shown above for the calibration matrix, **C**.

$$\mathbf{C} = \mathbf{Y'X} \left(\mathbf{X'X} \right)^{-1}$$

Note that in this step we invert (**X'X**), which is a (*p* x *p*) matrix created by multiplying a (*p* x *r*) matrix by an (*r* x *p*) matrix. Therefore *r*, the number of references, must be greater than or equal to *p*, the number of data points. In data rich analyses such as the multipoint spectra generated in FT-IR spectroscopy, this is a severe restriction. In general, the inverse least squares model is used when the number of data points has been greatly reduced; for example, to peak heights or areas.

Prediction

The calibration matrix allows us to predict concentration values from spectral data. Therefore, we solve the following equation to predict the concentrations in an unknown sample.

$$\mathbf{y} = \mathbf{xC'}$$

y is the vector that contains the estimate of the unknown concentrations (of size 1 x *c*).

x is the vector that contains the unknown spectrum (of size 1 x *p*).

Search method for identifying materials

The Search method of identifying materials is used to identify compounds that are similar to an unknown compound. The infrared spectrum of the material contains the information used to describe the material.

This section describes the problems and algorithms that are used to build collections, or libraries, of spectra and to measure the similarity between an unknown material and the materials in the library.

Setting up the data that will be used in the method

The Search method depends on comparing the spectrum of an unknown material with the spectra measured on a large number of known materials. This collection of known materials is frequently called a “spectral library,” or a “library” for short. Each spectrum in a library has the same registration. That is, each spectrum is represented by intensity information measured (or interpolated) at the same ordinate values.

The first step in adding a spectrum to a spectral library is to ensure that the abscissa values share the same unit. That is, a transmittance spectrum may not be added to an absorbance library. The conversion step is done automatically before adding a spectrum to the library. If the conversion is not available, the spectrum cannot be added to the library. The conversion algorithms will not be described in this document.

When the spectrum of a material is added to a spectral library, the registration of the new spectrum must be consistent with the registration required by the library. The registration is defined by three values. The starting ordinate value, the ending ordinate value and the spacing between consecutive ordinate values completely define the registration. To pass the registration test, a new spectrum must have the same data spacing and contain the same ordinate values as the spectral library.

There are several conditions that will cause a spectrum to fail the registration test:

- The new spectrum has the correct data spacing and ordinate range, but data points are shifted along the X-axis. This can be fixed by interpolation.

- The new spectrum has the correct data spacing, but its ordinate range does not enclose the ordinate range of the library. Since the spectrum does not contain enough data to describe the range of the library, it cannot be added.

- The ordinate range of the new spectrum encloses the ordinate range of the library, but the data spacing is different. This can be fixed by interpolation.

- Both the data spacing and ordinate range of the new spectrum are incorrect.

See “Spectral interpolation” later in this section for information about the use of interpolation to address the cases listed above.

Analyzing an unknown spectrum

A Search method includes an algorithm used to generate a metric describing the similarity between the spectrum measured on an unknown material and each spectrum in the library. This metric predicts which individual material in the library has a spectrum that is most similar to the spectrum of the unknown material.

The displayed results of a search are a list of the materials in the library that are most similar to the unknown material. The metric is also reported for each of the materials in the library. The magnitude of the metric is used subjectively to determine the confidence in a particular match.

Five algorithms are available for measuring the similarity between an unknown spectrum and each entry in the spectral library. These algorithms are described in the next sections.

When the spectrum of an unknown material is presented to the library, the unknown spectrum must be registered to the spectral library. The problems that can occur when a spectrum is added to a library (see “Setting up the data that will be used in the method”) can also occur when a search is performed on an unknown spectrum. The same spectral interpolation algorithm is used to address these problems (see the next section).

Spectral interpolation

The unknown spectrum must be registered, or “lined up,” with the spectra in the library. This is done in two main steps.

First, if the data spacing of the unknown spectrum is a larger number than the data spacing of the library, intermediate points are generated to more closely match up the data point spacings. This is the case where we need to “super-resolve” the unknown.

In the procedure that follows, “fUnkSpacing” is the distance between two successive data points in the unknown spectrum, and “fLibSpacing” is the distance between two successive data points in a spectrum in the library.

1. Add a data point between each pair of points in the unknown spectrum.

2. Between the first two points, estimate the new value as the average of the two nearest points. Use this same rule to interpolate a data point between the last two points. For all other points use a four-point interpolation where the interpolation weights are { -0.0625, +0.5625, +0.5625. -0.625 }. The four data points are the two natural points before the interpolated point and the two natural points after the interpolated point. The new unknown spectrum has a data spacing that is one-half that of the previous unknown spectrum; that is, fUnkSpacing = 0.5 * fUnkSpacing.

3. Perform these steps as long as fUnkSpacing > 0.9 * fLibSpacing.

The second main step makes sure that the data spacing and data point alignment are exact. This is done by using a well defined interpolation algorithm taken from *Numerical Recipes in C: The Art of Scientific Computing*, second edition, by William H. Press, Saul A. Teukolsky, William T. Vettering, and Brian P. Flannery (Cambridge University Press, 1992, pp. 113-116). The algorithm and code to implement this algorithm are available in the text.

Algorithms for measuring spectral similarity

Several algorithms, described in the next sections, are available for measuring the similarity between an unknown spectrum and any spectrum in a spectral library. When one of these algorithms is selected, the assumption is made that the unknown spectrum has been registered to the library and that the ordinate values that will be used for the measurement have been identified. Thus a vector of (registered) data values for both the unknown and library spectrum can be presented to the algorithm.

Correlation algorithm

This is the Thermo Scientific preferred algorithm. (Any of the algorithms described in the next sections also can be used.) It depends on computing a metric, a correlation coefficient, that relates an unknown spectrum to each spectrum in the library. The library spectra are ranked in descending order based on the magnitude of the metric. Finally, the spectra that have the largest metric are reported as most similar to the unknown compound.

The algorithm for computing the correlation value between an unknown spectrum and a specific library spectrum is described below.

The algorithm uses these definitions and conditions:

- **X** is the vector that represents the spectrum of the specific library compound. **X** has the form $\{ x_1, x_2, x_3, \ldots, x_{n-1}, x_n \}$ where x_i is the intensity at the location indicated by i.

- **Y** is the vector that represents the spectrum of the unknown compound. The length of vector **Y** is the same as the length of vector **X**, and **X** and **Y** line up.

- An important step is performed before the correlation calculation: An estimate of the derivative of **X** is computed and is called **X'**. Also, the estimated derivative of **Y** is computed and is called **Y'**.

- The derivative of **X** that is computed is a three-point smoothed derivative. As a result, the **X** vector entries become the **X'** vector entries according to the rules shown in the following algorithm procedure.

These steps are performed when the correlation algorithm is used:

1. Perform the calculations listed below as long as fUnkSpacing > 0.9 * fLibSpacing.

   $x'_1 = 0$
   $x'_2 = (x_3 - x_1) / 2$
   $x'_3 = (x_4 - x_2) / 2$
   …
   $x'_{n-1} = (x_n - x_{n-2}) / 2$
   $x'_n = 0$

2. Compute the square of the correlation value according to the equation shown below, where **X' * Y'** denotes the dot product between vectors **X'** and **Y'**. (The dot product gives a scalar.)

   $$r^2 = (\mathbf{X'} * \mathbf{Y'}) (\mathbf{X'} * \mathbf{Y'}) / (\mathbf{X'} * \mathbf{X'}) (\mathbf{Y'} * \mathbf{Y'})$$

   The metric that is reported is one hundred times the square root of r^2. The minimum value that this metric may attain is zero. This value implies that there is no relationship between the unknown spectrum and the particular library spectrum. The maximum value for the metric is one hundred. If the metric is one hundred, the unknown spectrum is within a multiplicative constant, either identical to, or the negative of, the particular library spectrum.

Absolute difference algorithm

The absolute difference algorithm depends on computing a distance metric that relates an unknown spectrum to each spectrum in the library. The library spectra are ranked in descending order based on the magnitude of the metric. Finally, the spectra that have the largest metric are reported as most similar to the unknown compound.

The algorithm for computing the absolute difference between an unknown spectrum and a specific library spectrum is described below.

The algorithm uses these definitions and conditions:

- **X** is the vector that represents the spectrum of the specific library compound. **X** has the form $\{ x_1, x_2, x_3, \ldots, x_{n-1}, x_n \}$ where x_i is the intensity at the location indicated by i.

- **Y** is the vector that represents the spectrum of the unknown compound. The length of vector **Y** is the same as the length of vector **X**, and **X** and **Y** line up.

- Before the absolute distance calculation is performed, a scale factor is computed to scale the unknown spectrum to the library spectrum. The scale factor is computed into the value "m" according to an algorithm that is described in the "Scaling algorithm" section later in this chapter.

These steps are performed when the absolute difference algorithm is used:

1. Determine the minimum value of the library vector and call it x_{min}.

2. Determine the minimum value of the unknown vector and call it y_{min}.

3. Use these values to compute a difference minimum according to this equation:

   $$d_{min} = y_{min} - mx_{min}$$

4. Use the equations below to compute two values: "Difference" is related to the absolute distance between the library spectrum and the scaled unknown spectrum. "Spectrum" is a measure of the absolute length of the unknown spectrum.

   $$\text{Difference} = | y_1 - mx_1 - d_{min} | + | y_2 - mx_2 - d_{min} | + \ldots + | y_n - mx_n - d_{min} |$$

   $$\text{Spectrum} = | y_1 - y_{min} | + | y_2 - y_{min} | + \ldots + | y_n - y_{min} |$$

5. Compute the metric according to this equation:

   $$\text{Metric} = 100 - 100 * \text{Difference} / \text{Spectrum}$$

   The maximum value for the metric is one hundred. If the metric is one hundred, the unknown spectrum is within a multiplicative constant, either identical to, or the negative of, the particular library spectrum.

Squared difference algorithm

The squared difference algorithm depends on computing a distance metric that relates an unknown spectrum to each spectrum in the library. The library spectra are ranked in descending order based on the magnitude of the metric. Finally, the spectra that have the largest metric are reported as most similar to the unknown compound.

The algorithm for computing the squared difference between an unknown spectrum and a specific library spectrum is described below.

The algorithm uses these definitions and conditions:

- **X** is the vector that represents the spectrum of the specific library compound. **X** has the form $\{ x_1, x_2, x_3, \ldots, x_{n-1}, x_n \}$ where x_i is the intensity at the location indicated by i.

- **Y** is the vector that represents the spectrum of the unknown compound. The length of vector **Y** is the same as the length of vector **X**, and **X** and **Y** line up.

- Before the absolute distance calculation is performed, a scale factor is computed to scale the unknown spectrum to the library spectrum. The scale factor is computed into the value "m" according to an algorithm that is described in the "Scaling algorithm" section later in this chapter.

These steps are performed when the squared difference algorithm is used:

1. Determine the minimum value of the library vector and call it x_{min}.

2. Determine the minimum value of the unknown vector and call it y_{min}.

3. Use these values to compute a difference minimum according to the following equation.

   $$d_{min} = y_{min} - mx_{min}$$

4. Use the equations below to compute two values: "Difference" is related to the squared distance between the library spectrum and the scaled unknown spectrum. "Spectrum" is a measure of the squared length of the unknown spectrum.

   $$\text{Difference} = (y_1 - mx_1 - d_{min})^2 + (y_2 - mx_2 - d_{min})^2 + \ldots + (y_n - mx_n - d_{min})^2$$

   $$\text{Spectrum} = (y_1 - y_{min})^2 + (y_2 - y_{min})^2 + \ldots + (y_n - y_{min})^2$$

5. Compute the metric according to this equation:

   $$\text{Metric} = 100 - 100 * \text{Difference} / \text{Spectrum}$$

The maximum value for the metric is one hundred. If the metric is one hundred, the unknown spectrum is within a multiplicative constant, either identical to, or the negative of, the particular library spectrum.

Absolute derivative algorithm

The absolute derivative algorithm depends on computing a distance metric that relates an unknown spectrum to each spectrum in the library. The library spectra are ranked in descending order based on the magnitude of the metric. Finally, the spectra that have the largest metric are reported as most similar to the unknown compound.

The algorithm for computing the absolute derivative between an unknown spectrum and a specific library spectrum is described below.

The algorithm uses these definitions and conditions:

- **X** is the vector that represents the spectrum of the specific library compound. **X** has the form $\{ x_1, x_2, x_3, \ldots, x_{n-1}, x_n \}$ where x_i is the intensity at the location indicated by i.

- **Y** is the vector that represents the spectrum of the unknown compound. The length of vector **Y** is the same as the length of vector **X**, and **X** and **Y** line up.

- Before the absolute distance calculation is performed, a scale factor is computed to scale the unknown spectrum to the library spectrum. The scale factor is computed into the value “m” according to an algorithm that is described in the “Scaling algorithm” section later in this chapter.

- An important step is performed before the distance calculation: An estimate of the derivative of **X** is computed and is called **X'**. Also, the estimated derivative of **Y** is computed and is called **Y'**.

- The derivative of **X** that is computed is a three-point smoothed derivative. As a result, the **X** vector entries become the **X'** vector entries according to these rules:

$$x'_1 = 0$$
$$x'_2 = (x_3 - x_1) / 2$$
$$x'_3 = (x_4 - x_2) / 2$$
$$\ldots$$
$$x'_{n-1} = (x_n - x_{n-2}) / 2$$
$$x'_n = 0$$

These steps are performed when the absolute derivative algorithm is used:

1. Use the equations below to compute two values: "Difference" is related to the absolute distance between the derivative of the library spectrum and the derivative of the scaled unknown spectrum. "Spectrum" is a measure of the absolute length of the derivative of the unknown spectrum.

   $$\text{Difference} = | y'_1 - mx'_1 | + | y'_2 - mx'_2 | + \ldots + | y'_n - mx'_n |$$

   $$\text{Spectrum} = | y'_1 | + | y'_2 | + \ldots + | y'_n |$$

2. Compute the metric according to this equation:

   $$\text{Metric} = 100 - 100 * \text{Difference} / \text{Spectrum}$$

   The maximum value for the metric is one hundred. If the metric is one hundred, the unknown spectrum is within a multiplicative constant, either identical to, or the negative of, the particular library spectrum.

Squared derivative algorithm

The squared derivative algorithm depends on computing a distance metric that relates an unknown spectrum to each spectrum in the library. The library spectra are ranked in descending order based on the magnitude of the metric. Finally, the entries that have the largest metric are reported as most similar to the unknown compound.

The algorithm for computing the squared derivative between an unknown spectrum and a specific library spectrum is described below.

The algorithm uses these definitions and conditions:

- **X** is the vector that represents the spectrum of the specific library compound. **X** has the form $\{ x_1, x_2, x_3, \ldots, x_{n-1}, x_n \}$ where x_i is the intensity at the location indicated by i.

- **Y** is the vector that represents the spectrum of the unknown compound. The length of vector **Y** is the same as the length of vector **X**, and **X** and **Y** line up.

- Before the absolute distance calculation is performed, a scale factor is computed to scale the unknown spectrum to the library spectrum. The scale factor is computed into the value "m" according to an algorithm that is described in the "Scaling algorithm" section later in this chapter.

- An important step is performed before the distance calculation: An estimate of the derivative of **X** is computed and is called **X'**. Also, the estimated derivative of **Y** is computed and is called **Y'**.

- The derivative of **X** that is computed is a three-point smoothed derivative. As a result, the **X** vector entries become the **X'** vector entries according to these rules:

$$x'_1 = 0$$
$$x'_2 = (x_3 - x_1) / 2$$
$$x'_3 = (x_4 - x_2) / 2$$
$$\ldots$$
$$x'_{n-1} = (x_n - x_{n-2}) / 2$$
$$x'_n = 0$$

These steps are performed when the squared derivative algorithm is used:

1. Use the equations below to compute two values: "Difference" is related to the squared distance between the derivative of the library spectrum and the derivative of the scaled unknown spectrum. "Spectrum" is a measure of the squared length of the derivative of the unknown spectrum.

$$\text{Difference} = (y'_1 - mx'_1)^2 + (y'_2 - mx'_2)^2 + \ldots + (y'_n - mx'_n)^2$$

$$\text{Spectrum} = (y'_1)^2 + (y'_2)^2 + \ldots + (y'_n)^2$$

2. Compute the metric according to this equation:

$$\text{Metric} = 100 - 100 * \text{Difference} / \text{Spectrum}$$

   The maximum value for the metric is one hundred. If the metric is one hundred, the unknown spectrum is within a multiplicative constant, either identical to, or the negative of, the particular library spectrum.

Scaling algorithm

The scaling algorithm is used to produce a factor that will put the unknown spectrum and the library spectrum onto the same scale. The following model is used.

"Unknown" = m * "Library"

The algorithm for computing the factor "m" uses these definitions and conditions:

- **X** is the vector that represents the spectrum of the specific library compound. **X** has the form $\{ x_1, x_2, x_3, \ldots, x_{n-1}, x_n \}$ where x_i is the intensity at the location indicated by i.

- **Y** is the vector that represents the spectrum of the unknown compound. The length of vector **Y** is the same as the length of vector **X**, and **X** and **Y** line up.

- An important step is performed before the scale factor calculation: An estimate of the derivative of **X** is computed and is called **X'**. Also, the estimated derivative of **Y** is computed and is called **Y'**.

- The derivative of **X** that is computed is a three-point smoothed derivative. As a result, the **X** vector entries become the **X'** vector entries according to these rules:

$$x'_1 = 0$$
$$x'_2 = (x_3 - x_1) / 2$$
$$x'_3 = (x_4 - x_2) / 2$$
$$\dots$$
$$x'_{n-1} = (x_n - x_{n-2}) / 2$$
$$x'_n = 0$$

This step is are performed when the scaling algorithm is used:

Compute the scale factor “m” according to the equation shown below, where **X' * Y'** denotes the dot product between vectors **X'** and **Y'**.

$$m = (\mathbf{X'} * \mathbf{Y'}) / (\mathbf{X'} * \mathbf{X'})$$

QC Compare method for classifying materials

The QC Compare method of classifying materials is a special case of the Search method used to identify compounds that are similar to an unknown compound used to produce a spectrum. See “Search method for identifying materials” for information about the algorithm used.

In this section we will concentrate on the differences between the QC Compare method and the Search method. For more detailed information about the algorithms, see “The Search method for identifying materials” later in this chapter.

Setting up the data that will be used in the method

Both the QC Compare and Search methods depend on comparing the spectrum of an unknown material with the spectra measured on a large number of known materials. This collection of known materials is frequently called a “spectral library,” or a “library” for short. The first difference between the two methods is the amount of information that is saved with the spectrum of each known material.

In a Search method, each material is defined by its name or description and the spectrum that is measured for the material. Both of these items are required for a QC Compare method. However, there is an additional piece of information that is saved for each material. In the QC Compare problem it is assumed that each material comes from a larger population that represents some attribute or feature of the material. This is called the “class” of the material.

When a new material is added to a spectral library in a Search method, it is necessary only to enter the name of the material and the spectrum associated with that material.

There is an additional step in the QC Compare method. Before creating a material library, it is necessary to define the classes (groupings) that will be present in the materials in the library. Then, when a new material is added to a QC Compare library, it is necessary to enter the name of the material, the class to which the material belongs, and the spectrum associated with that material.

Analyzing an unknown spectrum

A Search method includes an algorithm used to generate a metric describing the similarity between the spectrum measured on an unknown material and each spectrum in the library. The displayed results of a search are a list of the materials in the library that are most similar to the unknown material. The metric is also reported for each of the materials in the library. The magnitude of the metric is used subjectively to determine the confidence in a particular match.

The QC Compare method also includes these steps. However, there are some differences. First, in a Search method there are five algorithms available to generate a metric. (See the "Search method for identifying materials" section for descriptions of the algorithms.) In the QC Compare method the correlation algorithm is always used. The metric that is generated is different, since a monotonic scaling function is applied to the result before it is reported. The scaling function is described below, where r_{old} is the metric generated in a Search method, and r_{new} is the metric generated in the QC Compare method.

$$x = r_{old} / 100$$
$$r_{new} = 100 (x^{21} + x) / 2$$

Also, in the QC Compare method a filter is used as the list of similar materials is generated. By definition the list will contain only one entry for each class that has been defined. That is, for each class only the material that is most similar to the unknown material will be reported.

Thus, in a Search method the number of entries in the reported list will equal the number of entries in the spectral library. In a QC Compare method, the reported list will contain only a number of entries equal to the number of classes that have been defined for the spectral library.

In the literature the QC Compare method is referred to as a "nearest neighbor" classification. The similarity to a class is not measured as the average distance to the entries representing a class. Instead, it is measured as the distance to the entry that is most similar to the spectrum of the unknown.

References

T.M. Cover and P.E. Hart, "Nearest Neighbor Pattern Classification," *IEEE Transactions on Information Theory*, Vol. IT-13(1), 1967, pp. 21-27.

A tutorial for nearest neighbor classification is available on the World Wide Web at this address:

http://www-cgrl.cs.mcgill.ca/~soss/cs644/projects/simard

Similarity match for material identification

The objective of the similarity match algorithm is to report a value that measures how similar an unknown spectrum is to a set of standard spectra. The algorithm uses the residual spectrum method.

Calibration

The starting point for calibration is a set of standard spectra that is represented in matrix form by **X**. **X** is *r* x *p* where *r* is the number of references and *p* is the number of spectral data points per spectrum.

The following steps are performed in the calibration:

1. Initialize a matrix **Z** that has *r* rows and *p* columns. This matrix will hold the Gram-Schmidt calibration array. All entries in **Z** should be initialized to zero.

2. Select the first column of **X**. This standard spectrum is used to determine the first term of the Gram-Schmidt expansion. Call this vector **u**. **u** is of size 1 x *p*.

3. Compute the squared norm of **u**.

   $u^2 = \mathbf{u'u}$

4. Scale **u** to unit length according to **u** = **u** / *u*.

5. Insert **u** into the first row of **Z**.

6. Select the next column of **X**. Call this vector **u**. **u** is of size 1 x *p*.

7. Use the equation below to compute the amount of each spectrum in **Z** that is found in vector **u**. **t** is of size 1 x *r*.

   $\mathbf{t} = \mathbf{uZ'}$

8. Determine the amount of **u** that is left over after the contribution from **Z** is removed, according to this equation:

   $\mathbf{u} = \mathbf{u} - \mathbf{tZ}$

9. Use the equation below to compute the squared norm of **u**. If u^2 equals zero, set it to unity.

   $u^2 = \mathbf{uu'}$

10. Scale **u** to unit length according to **u** = **u** / *u*.

11. Insert **u** into the next row of **Z**.

12. Return to step 6. Repeat this sequence until all columns of **X** have been extracted. After all columns of **X** have been extracted, **Z** represents the Gram-Schmidt expansion of the **X** matrix. **Z** will be used during the prediction step.

Prediction

These steps are performed in a similarity match prediction:

1. Use the equation below to compute the squared norm or magnitude of an unknown spectrum called **x**. **x** is of size 1 x *p*.

   $x^2 = \mathbf{xx'}$

2. Use the equation below to compute the amount of each spectrum in **Z** that is found in vector **x**. **t** is of size 1 x *r*.

   $\mathbf{t} = \mathbf{xZ'}$

3. Use the equation below to determine the amount of **x** that is left over after the contribution from **Z** is removed. **u** is of size 1 x *p*.

   $\mathbf{u} = \mathbf{x} - \mathbf{tZ}$

4. Compute the squared norm or magnitude of **u** according to this equation:

   $u^2 = \mathbf{uu'}$

5. Compute the similarity from x^2 and u^2 according to the rules shown below. The final similarity value is *S*.

   $V = (x^2 - u^2) / x^2$
   $U = (V^{21} + V) / 2$
   $S = 100 * U$

   Using this measure a perfect match has a similarity of 100. If you want a perfect match to have a value of zero (that is, zero distance between the unknown and the standards), do the following transformation.

   $S = 100 - S$

Distance match for material identification

The objective of the distance match algorithm is to report a value that measures how similar unknown spectrum is to a set of standard spectra. The algorithm is based on wavenumber distance and uses the residual spectrum method.

Calibration

The starting point for calibration is a set of standard spectra that is represented in matrix form by **X**. **X** is *r* x *p* where *r* is the number of references and *p* is the number of spectral data points per spectrum.

The following steps are performed in the calibration:

1. Initialize a matrix **M** that has 1 row and *p* columns. This matrix will hold the mean spectrum for all of the standards.

2. Initialize a matrix **D** that has 1 row and *p* columns. This matrix will hold the estimated standard deviation for all of the standards.

3. Compute the mean spectrum of **X** according to the equation below and save the result into **M**.

   $$m_{1,i} = (\sum x_{j,i}) / r$$

4. Compute the standard deviation of **X** according to the equation below and save the result into **D**.

   $$d_{1,i}^{\,2} = (\sum (x_{j,i} - m_{1,i})^2) / (r - 1)$$

   M and **D** will be used during the prediction step.

Prediction

The steps below are performed in a distance match prediction. For the prediction we use a scalar value *q* that defines the number of standard deviations from the mean spectrum that are allowed for each data point in the spectrum.

1. Initialize a scalar counter, called *s*, that will be the number of times that the unknown spectrum **x** deviates from the mean spectrum by too great a value. **x** is of size 1 x *p*.

2. For each of the p data points in the spectrum, compute the deviations from the mean according to this equation:

   $$z = (x_{1,I} - m_{1,I}) / d_{1,I}$$

3. If the absolute value of z is greater than q, increase s by 1.

4. Return to step 2. Repeat this sequence until all of the data points have been considered.

5. Compute the similarity measure as S according to this equation:

   $$S = 100 * s / p$$

   This implies that a perfect score is zero.

Discriminant Analysis method for classifying materials

The discriminant analysis method is a classification analysis technique. It applies the spectral information in the specified regions of an unknown sample spectrum to a stored method model to determine which class of standards is most similar to the unknown. A measurement of the Mahalanobis distance between the unknown sample and each reported class is also provided.

Data pretreatment

The matrix $\mathbf{Y_0}$ contains the classification data for a set of calibration standards. There is one class per standard. Therefore, $\mathbf{Y_0}$ is r x 1 where there are r references and 1 category.

The class value will take one of c values; that is, there are c categories, or classes.

The matrix $\mathbf{X_0}$ contains the spectral data for a set of calibration standards. $\mathbf{X_0}$ is r x p where there are r references and p spectral data points.

Several data pretreatments are possible. Some examples are described below.

In **mean centering** the column-wise mean (the average over all references for each data point) may be computed and subtracted from each element of the column. If the general terms are x_{ij}, the averages are x_j. x_j is subtracted from each x_{ij}.

In **variance scaling** the column-wise variance (the variance over all references for each data point) may be computed. The square root of the variance is used to scale each element of the column. This ensures that each column (data point) has equal weight before starting the analysis.

Setting up the data that will be used in the method

After the spectral data in $\mathbf{X_0}$ have been rescaled, the resultant data is called $\mathbf{X}$. $\mathbf{X}$ is the starting point for the discriminant analysis calibration and is of size r x p (r references by p data points).

Calibration

The following steps are performed in a discriminant analysis calibration for dimensions $h = 1, 2, ..., a$:

1. Decompose the **X** matrix into c submatrices, one for each category. Denote the submatrix for the first class as X_1. It will be of size r_1 x p, where r_1 is the number of references assigned as being in the first category. Subsequent submatrices are denoted $\mathbf{X_2}$, $\mathbf{X_3}$, … $\mathbf{X_c}$. The number of rows in each submatrix will be r_2, r_3, … r_c. The number of columns is always p.

2. For each category compute a mean spectrum. The mean spectrum for category i is denoted $\mathbf{m_i}$ and it is of size 1 x p.

3. For each $\mathbf{X_i}$ subtract $\mathbf{m_i}$ from each row. Although $\mathbf{X_i}$ has changed, still call the result of this operation $\mathbf{X_i}$.

4. If the intent is to compute a unique distribution for each class, continue with step 5. However, if the intent is to have one distribution that is used for all categories, skip to step 7.

5. For each $\mathbf{X_i}$ compute its eigenvalues and eigenvectors. Either compute a maximum number of eigenvalues ($k \leq r_i$) or compute the number of eigenvectors necessary to describe 99.99% of the total variance in $\mathbf{X_i}$. Denote the number of eigenvalues and vectors that are produced as q. The eigenvectors for class i, called Vi, make up a q x p matrix. The eigenvalues for class i make up a q x 1 vector.

6. Convert the eigenvalue vector into a covariance matrix according to the rules described below. The resulting matrix, called $\mathbf{C_i}$, will be of size q x q. Each nondiagonal entry will be zero, and the diagonal entry will be computed according to the equation shown below. c_{kk} is the entry in the kth diagonal in the $\mathbf{C_i}$ matrix, and e_k is the kth entry in the eigenvalue vector.

   $$c_{kk} = ((r_i - 1) / e_k) (1 / q)$$

   e_k is the kth eigenvalue.

   The calibration for computing a unique distribution for each class is finished. See the Note after step 9.

7. Recombine the spectral submatrices (with the category mean removed) into a new spectral matrix, according to the following equation. The result, called $\mathbf{X_+}$, will be the same size as the original **X** matrix.

   $$\mathbf{X_+ = X_1 \mid X_2 \mid X_3 \mid \ldots X_c \mid}$$

8. For $\mathbf{X}_+$ compute its eigenvalues and eigenvectors. Either compute a maximum number of eigenvalues ($k \leq r_i$) or compute the number of eigenvectors necessary to describe 99.99% of the total variance in $\mathbf{X}_+$. Denote the number of eigenvalues and vectors that are produced as q. The eigenvectors, called **V**, make up a q x p matrix. The eigenvalues make up a q x 1 vector.

9. Convert the eigenvalue vector into a covariance matrix according to the rules described below. The resulting matrix, called **C**, will be of size q x q. Each nondiagonal entry will be zero, and the diagonal entry will be computed according to the equation shown below. c_{kk} is the entry in the kth diagonal in the **C** matrix, and e_k is the kth entry in the eigenvalue vector.

   $$c_{kk} = (r_i / q) (1 / e_k)$$

   e_k is the kth eigenvalue.

   Since each class, or category, is assumed to have the same distribution, the **V** and **C** matrices are valid for every category. Thus, for any class i, $\mathbf{C_i} = \mathbf{C}$ and $\mathbf{V_i} = \mathbf{V}$.

   The calibration for determining one distribution for all categories is finished. See the following Note.

Note When the calibration is completed, each category, i, is described by one vector, $\mathbf{m_i}$, and two matrices, $\mathbf{C_i}$ and $\mathbf{V_i}$. This is true regardless of whether the calibration computed a unique distribution for each class or determined one distribution for all categories. The next section explains how the vector and matrices are used during a prediction. ▲

Prediction

These steps are performed in a discriminant analysis prediction:

1. Scale and center **x** as indicated by the method. The scaling and centering constants are taken from the calibration data. (**x** is assumed to be of size 1 x p.)

2. Remove the mean for the first class according to $\mathbf{x}_+ = \mathbf{x} - \mathbf{m_i}$.

3. Give the projection result by computing the matrix product $\mathbf{p} = \mathbf{V_i x_+}'$. **p** is of size q x 1. (Note that $\mathbf{x}_+'$ is the transposition of $\mathbf{x}_+$.)

4. Calculate the Mahalanobis distance (or M-distance), r, of the sample from the category mean according to following equation. The result, r_i, is a scalar.

   $$r_i^2 = \mathbf{p'C_i\,p}$$

Repeat steps 2 through 4 for the rest of the classes. When the process is completed for all the classes, we have c values of r_i. The most similar category has the minimum M-distance.

5. Choose r_z, the smallest value, from the list of values. The most likely choice is category z. The magnitude of the r_i gives information about how similar the unknown sample is to any of the categories.

Measurement only method for measuring spectral features

In quantitative analysis methods the spectral intensity or location information is used to estimate how much of an individual component is present. That is, the spectral information is mapped or transformed to a concentration or amount of component.

In qualitative analysis methods the same intensity or location information is used to determine a standard of performance for the spectrum. That standard of performance describes how similar an unknown spectrum is to a set of standard reference spectra. In this case the spectral information is mapped or transformed to a category for the spectrum or to a performance value.

In a measurement only method the spectral data is not mapped to any other value or property. The value that is reported is the actual measurement taken from the spectrum. The measurement types are taken from the measures described in the "Region type" section in the "Parameter Settings" chapter. The measurement only method can be used to measure any of the region types that produce a single value.

Calibration

Little is done in the calibration step for a measurement only method. Since the result of the method is a measurement that may be taken from any spectrum, there are no calibration reference standards associated with the method. Thus there is no dependence on the resolution of the spectral data. The only check made during calibration is to make sure that valid data are entered for the spectral region.

Prediction

In the prediction step the specific value is computed from the unknown spectrum. For example, if a peak area from 3200 to 2800 cm^{-1} is requested, this value is computed for the current spectrum and the result is presented in the report.

It is possible to report an error in a measurement if the unknown spectrum does not include data in the region where the measurement is performed. In that case no valid measurement may be performed.

Fit values for spectrum evaluation

Three types of fit value are computed in the TQ Analyst software. In each case the objective is to determine how similar an unknown spectrum is to the set of reference standards used to calibrate the TQ Analyst method. The reported value is a quality measure of the unknown spectrum. If the unknown spectrum is very similar to the reference standard spectra, we should expect the answer from the TQ Analyst method to be accurate and precise. If the unknown spectrum is unlike any of the reference standards, we may be less confident about the quality of the answer.

The reason that there are three different fit values is that there are three slightly different situations where a spectrum quality value is needed:

- For quantitative methods, we need to determine how similar the entire unknown spectrum is to the set of reference standard spectra. Since quantitative analysis assumes that the effects of the individual components are (approximately) additive, the algorithm for determining fit should assume that the linear combination of the reference standards is important.

- Differences between the unknown spectrum and the reference standard spectra in spectral regions that are not used in the analysis may not be important. Spectral artifacts or compositional impurities or interferences are unlikely to be important if they appear in sections of the spectrum that are not analyzed. Thus we need to determine how similar the unknown spectrum is to the reference standard spectra in the analytical regions of interest.

- For classification methods, the fit value is a measure of how similar the unknown spectrum is to the individual reference standards. In this case it is more important to compute a metric that relates the unknown spectrum to each of the reference standard spectra.

The algorithms used to calibrate and predict with these methods are described below.

Full spectrum fit value

This value is used for TQ Analyst quantitative methods.

Calibration

The following steps are performed in a full spectrum fit value calibration. The starting point for the calibration is a set of standard spectra that in matrix form is represented by **X'**. **X'** is r x p' where r is the number of references and p' is the number of spectral data points per spectrum.

1. Deresolve the spectral data in **X'** to produce the reference data that is used to compute the full spectrum fit value. A data matrix **X** of size r x p is formed by summing the values from four sequential columns to produce one new value. If p' is not evenly divisible by four, the last sum is computed from fewer than four values. The number of columns in **X**, denoted by p, equals the integer part of $(p' + 3) / 4$.

2. Read a value "Fit type" from the user interface to determine if any spectral preprocessing should be done. If the "Fit type" is set to measure from a zero baseline, no spectral preprocessing is done on the **X** matrix.

3. If the "Fit type" is set to measure from the mean of the reference standard spectra, average the rows of the **X** matrix to give a vector **m**. **m** is 1 x p.

4. Subtract **m** from each row of **X**.

5. Finish the calibration by using the calibration algorithm described in the "Similarity match for material identification" section.

Prediction

The following steps are performed in a full spectrum fit value prediction. The unknown spectrum, **x'**, is taken as the starting point for computing the fit value. **x'** is 1 x p'.

1. Deresolve the spectrum in **x'** to produce the spectrum that is used to compute the full spectrum fit value. The deresolved spectrum **x** is of size 1 x p and is formed by summing the values from four sequential columns in **x'** to produce one new value. If p', the number of columns or data points in the original data, is not evenly divisible by four, the last sum is computed from fewer than four values.

2. If the “Fit type” is set to measure from a zero baseline, no spectral preprocessing is done on **x**. If the “Fit type” is set to measure from the mean of the reference standard spectra, subtract **m** from **x**.

3. Finish the prediction by using the prediction algorithm described in the “Similarity match for material identification” section. The described technique is used to give the reported result.

Region fit value

This value is used for TQ Analyst quantitative methods.

Calibration

The following steps are performed in a region fit value calibration. The starting point for calibration is a set of standard spectra that in matrix form is represented by **X**. **X** is *r* x *p* where *r* is the number of references and *p* is the number of spectral data points per spectrum.

1. Read a value “Fit type” from the user interface to determine if any spectral preprocessing should be done. If the “Fit type” is set to measure from a zero baseline, no spectral preprocessing is done on the **X** matrix.

2. If the “Fit type” is set to measure from the mean of the reference standard spectra, average the rows of the **X** matrix to give a vector **m**. **m** is 1 x *p*.

3. Subtract **m** from each row of **X**.

4. Finish the calibration by using the calibration algorithm described in the “Similarity match for material identification” section.

Prediction

The following steps are performed in a region fit value prediction. The unknown spectrum, **x**, is taken as the starting point for computing the fit value. **x** is 1 x *p*.

1. If the “Fit type” is set to measure from a zero baseline, no spectral preprocessing is done on **x**. If the “Fit type” is set to measure from the mean of the reference standard spectra, subtract **m** from **x**.

2. Finish the prediction by using the prediction algorithm described in the “Similarity match for material identification” section. The described technique is used to give the reported result.

Search fit value

This value is used for TQ Analyst classification methods.

The algorithm for computing the search fit value is described in the “QC Compare method for classifying materials” section. For computing this fit value, it is assumed that only one category is defined and that all reference standards are assigned to that category.

Parameter Settings

This chapter describes how various parameter settings in TQ Analyst affect calibration and prediction calculations performed by the algorithms described in the "Algorithms" chapter.

Pathlength type

The following sections describe how the available settings of the Pathlength Type parameter in TQ Analyst affect the calculations performed during a method calibration or prediction.

Constant

A constant pathlength means that all concentrations of the standards are multiplied by one (a constant value).

Known

A known pathlength implies that each standard and each sample have been assigned a unique (and possibly different) pathlength value. This value is multiplied by the concentration values during calibration.

The known pathlength value is divided out of the calculated concentrations during prediction.

Predict

For this setting a pathlength is known for each standard during calibration and is treated as a separate component. The pathlength is also multiplied by the other component concentrations.

This setting implies that the pathlength will be estimated as an additional, separate component during prediction. This pathlength value is divided out of the calculated concentration for the other components.

Internal Reference (A=k*b*c)

An internal reference pathlength implies that a value, proportional to pathlength, is computed from a (single-valued) region. This value is multiplied by the concentration values during calibration.

During prediction this value is divided out of the calculated concentrations.

Peak Ratio Or Normalize (A/b=k*c)

A peak ratio or normalize pathlength implies that a value, proportional to pathlength, is computed from a (single-valued) region. This value is used to normalize the spectrum (by dividing by the value) before calibration or prediction.

Multiplicative Signal Correction (MSC)

For this setting a set of standard spectra is used to calibrate the method being developed. A mean spectrum is computed by averaging the standard spectra, point by point.

During prediction a linear (slope and intercept) model is assumed. These steps are performed:

1. Compute slope and intercept coefficients for the points in the spectral region regressed against the equivalent points from the mean spectrum.

2. Compute the difference spectrum as the difference between the unknown spectrum and the intercept value.

3. Divide this difference by the slope value to produce the MSC-corrected spectrum.

Standard Normal Variate (SNV)

A standard normal variate pathlength correction implies that the spectral region is scaled, removing mean and variability information. The mean value and the standard deviation are computed over all of the data points taken from a spectral region. Each data point is corrected by subtracting the mean value and then dividing the result by the standard deviation.

Region type

The following sections describe how the available settings of the Region Type parameter in TQ Analyst affect the calculations performed during a method calibration or prediction.

Fixed Location Height

For this setting the intensity at a fixed location is computed as follows:

1. Perform any baseline correction that is specified.
2. Find the data point closest to the fixed location.
3. Use a three-point (quadratic) interpolation about that point to determine the location of the maximum.
4. Compute the interpolated height at the maximum location. The result is a single value.

Average Height In Range

For this setting the average height in a spectral range is computed as follows:

1. Perform any baseline correction that is specified.
2. Sum all data points in the range.
3. Divide the total by the number of points. The result is a single value.

Maximum Height In Range

For this setting the maximum height in a spectral range is computed as follows:

1. Perform any baseline correction that is specified.
2. Find the maximum intensity over the points in the spectral range. The result is a single value.

Minimum Height In Range

For this setting the minimum height in a spectral range is computed as follows:

1. Perform any baseline correction that is specified.

2. Find the minimum intensity over the points in the spectral range. The result is a single value.

Absolute Maximum In Range

For this setting the absolute maximum height in a spectral range is computed as follows:

1. Perform any baseline correction that is specified.

2. Take the absolute value of all intensities in the spectral range.

3. Find the maximum of the absolute values. The result is a single value.

Area

For this setting the area over a spectral range is computed according to the trapezoidal rule. These steps are performed:

1. Perform any baseline correction that is specified.

2. Sum the intensities of all points in the spectral range. If a point is the first or last in the range, add only half its intensity.

3. Divide the sum by one less than the number of points in the spectral range. The result is a single value.

Computed Area

For this setting the baseline-corrected area over a spectral range is computed according to the trapezoidal rule. The endpoints of the area region define the two fixed baseline points. These steps are performed:

1. Set the baseline to a fixed-location, two-point baseline with the area endpoints as the frequency values.

2. Perform a baseline correction as described in “Fixed Location” in the “Two Points” section in “Baseline type” later in this chapter.

3. Sum the intensities of all points in the spectral range. If a point is the first or last in the range, add only half its intensity. (This is not important because these points will, by definition, have zero intensity.)

4. Divide the sum by one less than the number of points in the spectral range. The result is a single value.

RMS Noise

For this setting the root mean squared intensity over a spectral range is computed as follows:

1. Perform any baseline correction that is specified.

2. Determine the average least squares straight line over the spectral range.

3. Subtract the fitted straight line from each data point in the spectral range.

4. Sum the square of the intensity values from step 3 over all points in the spectral range.

5. Divide the total by one less than the number of data points.

6. Take the square root of the quotient to find the rms noise value. The result is a single value.

Peak-To-Peak Noise

For this setting the peak-to-peak noise intensity over a spectral range is computed as follows:

1. Perform any baseline correction that is specified.

2. Determine the least squares straight line over the spectral range.

3. Subtract the fitted straight line from each data point in the spectral range.

4. Find the maximum and minimum intensity values from step 3 over all points in the spectral range.

5. Compute the difference between the maximum value and the minimum value to find the peak-to-peak noise value. The result is a single value.

Interpolated Height At Exact Location

For this setting the intensity at the exact location specified is computed as follows:

1. Perform any baseline correction that is specified.

2. Determine the data point closest to the location given.

3. Using this data point and the points on either side of it, use a three-point (quadratic) interpolation to determine the height at the exact location. The result is a single value.

Peak Location (Interpolated)

For this setting the interpolated peak location in a spectral region is computed as follows:

1. Perform any baseline correction that is specified.

2. Find the data point associated with the maximum intensity over the points in the spectral range.

3. Using this data point and the points on either side of it, use a three-point (quadratic) interpolation to determine the location of the intensity maximum. (Set the derivative of the quadratic function to zero and solve for the location.) The result is a single value.

Peak Height (Interpolated)

For this setting the interpolated peak location in a spectral region is computed as follows:

1. Perform any baseline correction that is specified.

2. Find the data point associated with the maximum intensity over the points in the spectral range.

3. Using this data point and the points on either side of it, use a three-point (quadratic) interpolation to determine the intensity maximum. (Set the derivative of the quadratic function to zero, solve for the location and then compute the intensity at that location.) The result is a single value.

Peak Width (At Half Maximum)

For this setting the software determines the full width at half peak maximum for the largest peak in a spectral range. These steps are performed:

1. Perform any baseline correction that is specified.

2. Find the data point associated with the maximum intensity over the points in the spectral range.

3. Using this data point and the points on either side of it, use a three-point (quadratic) interpolation to determine the intensity maximum.

4. From the data point of maximum intensity, move in the direction of lower frequency until reaching a point whose intensity is less than half the maximum intensity.

5. Use that point and the previous point (whose intensity is greater than half the maximum intensity) to perform a linear interpolation to find the location where the intensity is exactly one-half of the maximum intensity.

6. Save this point as the first width point.

7. From the data point of maximum intensity, move in the direction of higher frequency until reaching a point whose intensity is less than half the maximum intensity.

8. Use that point and the previous point (whose intensity is greater than half the maximum intensity) to perform a linear interpolation to find the location where the intensity is exactly one-half of the maximum intensity.

9. Save this point as the second width point.

10. Find the peak width by taking the difference between the locations of the second width point and the first width point. The result is a single value.

Location At 1% (or 2%, 5% or 10%) Of Peak

For these settings the software finds the location where the intensity is 1% (or 2%, 5% or 10%, depending on the setting) of the maximum intensity in a spectral range. These steps are performed:

1. Perform any baseline correction that is specified.

2. Find the data point associated with the maximum intensity over the points in the spectral range.

3. Using this data point and the points on either side of it, use a three-point (quadratic) interpolation to determine the intensity maximum.

4. From the data point of maximum intensity, move in the direction of lower frequency until reaching a point whose intensity is less than 1% (or 2%, 5% or 10%) of the maximum intensity.

5. Use that point and the previous point—whose intensity is greater than 1% (or 2%, 5% or 10%) of the maximum intensity—to perform a linear interpolation to find the location where the intensity is exactly 1% (or 2%, 5% or 10%) of the maximum intensity.

6. Save this point as the first width point. If the 1% point (or 2%, 5% or 10% point) does not occur within the spectral range, save a large negative number as the first width point.

7. From the data point of maximum intensity, move in the direction of higher frequency until reaching a point whose intensity is less than 1% of the maximum intensity.

8. Use that point and the previous point—whose intensity is greater than 1% (or 2%, 5% or 10%) of the maximum intensity—to perform a linear interpolation to find the location where the intensity is exactly 1% (or 2%, 5% or 10%) of the maximum intensity.

9. Save this point as the second width point. If the 1% point (or 2%, 5% or 10% point) does not occur within the spectral range, save a large positive number as the second width point.

10. Determine whether the first width point or second width point is closer to the location of maximum intensity. The closer point is the location of the 1% (or 2%, 5% or 10%) peak. The result is a single value.

Spectrum Range

For this setting the points in a spectral range are determined as follows:

1. Perform any baseline correction that is specified.

2. Determine the data point closest to the first spectral region value. This is the first point.

3. Determine the data point closest to the second spectral region value. This is the last point.

4. Extract all data points from the (baseline-corrected) spectrum between, and including, the first point and the last point. The result is a vector of data (intensity) values.

Baseline type

The following sections describe how the available settings of the Baseline Type parameter in TQ Analyst affect the calculations performed during a method calibration or prediction.

In the “Region type” section each procedure starts with the step “Perform any baseline correction that is specified.” This section describes the baseline corrections that can be specified.

None

When this baseline type is applied, no change is made to the intensity of the spectrum.

One Point

When Baseline Type is set to One Point, the settings described in the next sections are available.

Fixed Location

This setting uses a single specified frequency value. These steps are performed:

1. Determine the data point closest to the single frequency value.
2. Save the intensity at this point as the baseline intensity.
3. Subtract the baseline intensity from each point in the spectral range before performing the region calculation.

Average In Range

This setting uses two specified frequency values, which define a baseline region in the spectrum. These steps are performed:

1. Average the intensity at all data points in the range enclosed by the baseline frequencies.
2. Save this value as the baseline intensity.

3. Subtract the baseline intensity from each point in the spectral range before performing the region calculation.

Maximum In Range

This setting uses two specified frequency values, which define a baseline region in the spectrum. These steps are performed:

1. Find the maximum intensity value in the range enclosed by the baseline frequencies.

2. Save this value as the baseline intensity.

3. Subtract the baseline intensity from each point in the spectral range before performing the region calculation.

Minimum In Range

This setting uses two specified frequency values, which define a baseline region in the spectrum. These steps are performed:

1. Find the minimum intensity value in the range enclosed by the baseline frequencies.

2. Save this value as the baseline intensity.

3. Subtract the baseline intensity from each point in the spectral range before performing the region calculation.

Two Points

When Baseline Type is set to Two Points, the settings described in the next sections are available.

Fixed Location

This setting uses two specified baseline regions, each defined by a single frequency value. These steps are performed:

1. Determine the data point closest to the first frequency value (from the first baseline region).

2. Save the location and intensity of this point as the first baseline intensity.

3. Determine the data point closest to the second frequency value (from the second baseline region).

4. Save the location and intensity of this point as the second baseline intensity.

5. Compute the linear baseline passing through the two baseline points from steps 2 and 4.

6. At each frequency in the spectral region, compute the linear baseline value and subtract it from the spectrum intensity.

Average In Range

This setting uses two specified baseline regions, each defined by two frequency values. These values define the region of the corresponding baseline point. The following steps are performed:

1. Find the data points closest to the two frequency values that define the first baseline region.

2. Average the intensities of all data points in the first baseline region.

3. Save this average as the first baseline intensity. The location of the first baseline point is the average of the X-axis values of the points found in step 1.

4. Find the data points closest to the two frequency values that define the second baseline region.

5. Average the intensities of all data points in the second baseline region.

6. Save this average as the second baseline intensity. The location of the second baseline point is the average of the X-axis values of the points found in step 4.

7. Compute the linear baseline passing through the two baseline points.

8. At each frequency in the spectral region, compute the linear baseline value and subtract it from the spectrum intensity.

Maximum In Range

This setting uses two specified baseline regions, each defined by two frequency values. These values define the region of the corresponding baseline point.

The following steps are performed:

1. Find the data points closest to the two frequency values that define the first baseline region.

2. Find the maximum intensity value among all data points in the first baseline region.

3. Save this value as the first baseline intensity. The data point where this value occurs is the first baseline point.

4. Find the data points closest to the two frequency values that define the second baseline region.

5. Find the maximum intensity value among all data points in the second baseline region.

6. Save this value as the second baseline intensity. The location where this value occurs is the second baseline point.

7. Compute the linear baseline passing through the first and second baseline points.

8. At each frequency in the spectral region, compute the linear baseline value and subtract it from the spectrum intensity.

Minimum In Range

This setting uses two specified baseline regions, each defined by two frequency values. These values define the region of the corresponding baseline point. The following steps are performed:

1. Find the data points closest to the two frequency values that define the first baseline region.

2. Find the minimum intensity value among all data points in the first baseline region.

3. Save this value as the first baseline intensity. The data point where this value occurs is the first baseline point.

4. Find the data points closest to the two frequency values that define the second baseline region.

5. Find the minimum intensity value among all data points in the second baseline region.

6. Save this value as the second baseline intensity. The data point where this value occurs is the second baseline point.

7. Compute the linear baseline passing through the first and second baseline points.

8. At each frequency in the spectral region, compute the linear baseline value and subtract it from the spectrum intensity.

Baseline Offset

The baseline offset is defined by a number that defines the level of the baseline. The software subtracts the offset value from each point in the spectral region before performing the region calculation.

Linear Removed

The calculations for this setting are performed over the limits of an individual spectral region. These steps are performed:

1. For a spectrum compute the best least squares linear baseline over the limits of the spectral region.

2. Subtract this baseline from the original spectrum before performing the region calculation.

Quadratic Removed

The calculations for this setting are performed over the limits of an individual spectral region. These steps are performed:

1. For a spectrum compute the best least squares quadratic baseline over the limits of the spectral region.

2. Subtract this baseline from the original spectrum before performing the region calculation.

Data normalization

The following sections describe how the data normalization parameters in TQ Analyst affect the calculations performed during a method calibration or prediction.

Use Mean Centering Technique

This option uses the set of standard spectra used to calibrate the method being developed. A mean spectrum is computed by averaging these spectra, point by point. A mean concentration is computed by averaging the concentration values.

During calibration and prediction the mean spectrum is subtracted (point by point) from the unknown spectrum. Then the mean concentration is added back to the predicted concentration.

Use Variance Scaling Technique

This option uses the set of standard spectra used to calibrate the method being developed. A standard deviation spectrum is computed by calculating the standard deviation of these spectra, point by point. A standard deviation concentration is computed by calculating the standard deviation of the concentration values of the standards.

During calibration and prediction the unknown spectrum is divided (point by point) by the standard deviation spectrum. Then the predicted concentration is multiplied by the standard deviation of the concentration values.

Smoothing and derivatives

The following sections describe how the smoothing parameters in TQ Analyst affect the calculations performed during a method calibration or prediction.

Simple derivatives

The First Derivative and Second Derivative settings of Data Format on the Spectra tab convert spectra to simple derivatives if no smoothing filter is selected. The next sections give the equations for these conversions.

First derivative

The first derivative of a spectrum at point i is computed using the following equation, where "spacing" is the distance, in wavenumbers, between consecutive data points. The spacing value can be found in the collection and processing information displayed when you click the Information button (labeled "i") in a spectral window of TQ Analyst.

$$x_i' = (x_{i+1} - x_{i-1}) / (2 * \text{spacing})$$

Second derivative

The second derivative of a spectrum at point i is computed using the following equation. (See "First derivative" above for a discussion of the "spacing" value.)

$$x_i'' = (x_{i+1} - 2 * x_i + x_{i-1}) / (6 * \text{spacing}^2)$$

Savitzky-Golay smoothing and derivatives

There are three factors that determine the coefficients to compute the Savitzky-Golay (S-G) functions:

- The order, k, of the derivative, where a zero implies that no derivative is taken.
- The polynomial order, p, for the fitting polynomial
- The number of points, n, in the filter.

When these values are specified, the S-G coefficients are defined by a linear filter independent of the dependent variable. The coefficients are calculated by solving for the best least squares fit of a polynomial of order p using n evenly spaced data points. Using these points, the center point of the range is approximated.

For derivatives, the appropriate derivative of the polynomial is taken. This means that for a cubic interpolating polynomial (p = 3) and a simple smooth (k = 0), the interpolated value is calculated as shown below.

$$y = ax^3 + bx^2 + cx + d$$

and

$$y(0) = d$$

This means that for a cubic interpolating polynomial (p = 3) and a first derivative (k = 2), the interpolated value is calculated as shown below.

$$y' = 3ax^2 + 2bx + c$$

and

$$y'(0) = c$$

This means that for a cubic interpolating polynomial (p = 3) and a second derivative (k = 1), the interpolated value is calculated as shown below.

$$y'' = 6ax + 2b$$

and

$$y''(0) = 2b$$

Norris derivatives

There are three factors that determine the coefficients to compute the Norris derivative functions:

- The order, d, of the derivative. A zero is not appropriate since this function always produces a derivative.
- The segment length, ℓ, which is the number of points combined into one number.
- The gap, g, which is the number of points between the segments.

The Norris derivative function is defined by a linear filter independent of the dependent variable. The specific coefficients are distributed over (2 * ℓ + g) points for a Norris first derivative and over (3 * ℓ + 2 * g) points for a Norris second derivative. The general model for the coefficients assumes a quadratic polynomial with a three point filter.

General References

Kenneth R. Beebe, Randy J. Pell and Mary Beth Seasholtz, *Chemometrics: A Practical Guide* (Wiley, 1998).

Richard G. Brereton, *Chemometrics: Applications of Mathematics and Statistics to Laboratory Systems* (Chichester, U.K.: Ellis Horwood Limited, 1990).

Richard Kramer, *Chemometric Techniques for Quantitative Analysis* (New York: Marcel Dekker, 1998).

H. Martens and T. Næs, *Multivariate Calibration* (Chichester, U.K.: Wiley, 1989).

William H. Press, Saul A. Teukolsky, William T. Vetterling and Brian P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing*, Second edition (Cambridge, U.K.: Cambridge University Press, 1992).

This page intentionally left blank

Index

D

E

F

H

I

K

L

M

N

O

P

Q

R

S

T

V