Assessment of Two-Dimensional Separative Systems Using Nearest Neighbor Distances Approach. Part I: Orthogonality Aspects.

Witold Nowik,*,†,‡ Sylvie Héron,† Myriam Bonose,† Mateusz Nowik,§ and Alain Tchapla†

† Univ. Paris-Sud, Groupe de Chimie Analytique de Paris-Sud EA 4041, LETIAM, IUT d'Orsay, Plateau de Moulon, 91400 Orsay, France.
‡ Laboratoire de Recherche des Monuments Historiques, 29 rue de Paris, 77420 Champs-sur-Marne, France.
§ AGH University of Science and Technology, Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering, Department of Measurement and Electronics, Al. A. Mickiewicza 30, 30-059 Kraków, Poland.
* Corresponding author: Tel: +33(0)140205651. Fax: +33(0)147033246. E-mail: witold.nowik@culture.gouv.fr. Present address: Centre de Recherche et de Restauration des Musées de France (C2RMF), 14 quai François Mitterrand, 75001 Paris, France.

Table of contents:

Table S-1. **Standards of anthraquinoids.**

Abbr.	Name	Supplier
14Cl	1,4-dichloroanthraquinone	TCI Europe (Zwijndrecht, Belgium)
15Br	1,5-dibromoanthraquinone	TCI Europe (Zwijndrecht, Belgium)
15Cl	1,5-dichloroanthraquinone	TCI Europe (Zwijndrecht, Belgium)
18Cl	1,8-dichloroanthraquinone	TCI Europe (Zwijndrecht, Belgium)
Aba	1-bromoanthraquinone	TCI Europe (Zwijndrecht, Belgium)
Aca	1-chloroanthraquinone	TCI Europe (Zwijndrecht, Belgium)
Afv	Anthraflavic acid	Maybridge (Trevillet, Great Britain)
Agl	Anthragallol	H. Schweppe (Frankenthal, Germany)
Aha	1-hydroxyanthraquinone	TCI Europe (Zwijndrecht, Belgium)
Ale	Aloe-emodin	Extrasynthese (Genay, France)
Ali	Alizarin	Acros Organics (Geel, Belgium)
Anq	Anthraquinone	Sigma Aldrich (Saint Louis, MO, USA)
Arf	Anthrarufin	Sigma Aldrich (Saint Louis, MO, USA)
Bba	2-bromoanthraquinone	TCI Europe (Zwijndrecht, Belgium)
Bca	2-chloroanthraquinone	TCI Europe (Zwijndrecht, Belgium)
Bha	2-hydroxyanthraquinone	TCI Europe (Zwijndrecht, Belgium)
Bua	2-*tert*-butylanthraquinone	Sigma Aldrich (Saint Louis, MO, USA)
Can	2-anthraquinonecarboxylic acid	Sigma Aldrich (Saint Louis, MO, USA)
Chr	Chrysophanol (Chrysophanic acid)	Extrasynthese (Genay, France)
Dan	Danthrone (Chrysazin)	Sigma Aldrich (Saint Louis, MO, USA)
Dma	1,4-dimethylanthraquinone	Extrasynthese (Genay, France)
Emo	Emodine	Acros Organics (Geel, Belgium)
Eta	2-ethylanthraquinone	Sigma Aldrich (Saint Louis, MO, USA)
Flk	Flavokermesic acid (Laccaic acid D)	H. Schweppe (Frankenthal, Germany)
FraA	Frangulin A	H. Schweppe (Frankenthal, Germany)
FraB	Frangulin B	H. Schweppe (Frankenthal, Germany)
Hma	2-hydroxymethylanthraquinone	Acros Organics (Geel, Belgium)
Hys	Hystazarin	H. Schweppe (Frankenthal, Germany)
Ker	Kermesic acid	H. Schweppe (Frankenthal, Germany)
Moh	Methoxyhystazarin	H. Schweppe (Frankenthal, Germany)
Oma	2,3-dimethylanthraquinone	Extrasynthese (Genay, France)
Phy	Physcion (Parietin)	Extrasynthese (Genay, France)
Pur	Purpurin	Sigma Aldrich (Saint Louis, MO, USA)
Qlz	Quinalizarin	H. Schweppe (Frankenthal, Germany)
Qza	Quinizarin	Sigma Aldrich (Saint Louis, MO, USA)
Rba	Ruberythric acid	H. Schweppe (Frankenthal, Germany)
Rgl	Rufigallol	H. Schweppe (Frankenthal, Germany)
Rhe	Rhein	Extrasynthese (Genay, France)
Tec	Tectoquinone	Sigma Aldrich (Saint Louis, MO, USA)
Xpu	Xanthopurpurin	H. Schweppe (Frankenthal, Germany)

Table S-2. **Stationary phases.**

Stationary phase	Manufacturer	Column L x i.d. [mm]	Particle size [µm]	Bonding	Abbr.
ChiraDex Gamma	Merck	250 x 2.0	5	γ-cyclodextrin	*CDG*
Cosmosil 5PYE	Nacalai-Tesque	250 x 4.6	5	2-(1-pyrenyl)ethyl	*PYE*
Discovery HS-PEG	Supelco	250 x 4.6	5	Polyethyleneglycol	*PEG*
Fluorosep-RP	ES industries	250 x 4.6	5	Pentafluorophenyl	*FSP*
Hypersil Gold	Thermo Hypersil	150 x 2.1	5	C18 monolayer, endcapped	*Gold*
Uptisphere NEC	Interchim	150 x 2.0	5	C18 monolayer, non endcapped	*NEC*

Table S-3. **Retention coefficients (χ).**

	Stationary phase					
Compound	**NEC**	**Gold**	**FSP**	**PYE**	**PEG**	**CDG**
14Cl	0.727	0.735	0.911	0.877	0.788	0.139
15Br	0.817	0.813	0.988	0.971	0.940	0.122
15Cl	0.741	0.745	0.880	0.909	0.817	0.128
18Cl	0.752	0.720	0.923	0.851	0.799	0.107
Aba	0.716	0.713	0.800	0.744	0.780	0.139
Aca	0.675	0.673	0.726	0.708	0.703	0.133
Afv	0.168	0.181	0.323	0.109	0.717	0.154
Agl	0.247	0.280	0.381	0.270	0.623	0.181
Aha	0.674	0.667	0.692	0.700	0.656	0.156
Ale	0.398	0.402	0.512	0.641	0.486	0.124
Ali	0.373	0.386	0.503	0.385	0.609	0.165
Anq	0.590	0.579	0.765	0.497	0.584	0.102
Arf	0.765	0.757	0.855	0.943	0.740	0.167
Bba	0.848	0.838	0.966	0.884	0.853	0.183
Bca	0.792	0.801	0.929	0.850	0.781	0.169
Bha	0.360	0.383	0.516	0.291	0.676	0.122
Bua	1.000	1.000	1.000	0.708	1.000	0.172
Can	0.368	0.402	0.588	0.501	0.598	0.498
Chr	0.829	0.813	0.880	0.949	0.791	0.170
Dan	0.682	0.670	0.766	0.872	0.644	0.183
Dma	0.895	0.882	0.914	0.758	0.879	0.199
Emo	0.634	0.660	0.681	0.700	0.942	0.240
Eta	0.834	0.829	0.866	0.672	0.835	0.172
Flk	0.173	0.202	0.109	0.266	0.828	0.872
Fra A	0.412	0.477	0.500	0.701	0.572	0.201
Fra B	0.412	0.486	0.506	0.727	0.611	0.201
Hma	0.294	0.302	0.401	0.278	0.394	0.083
Hys	0.216	0.252	0.359	0.177	0.633	0.108
Ker	0.166	0.190	0.109	0.348	0.779	1.000
Moh	0.371	0.389	0.501	0.297	0.626	0.086
Oma	0.814	0.801	0.738	0.736	0.826	0.189
Phy	0.934	0.910	0.971	1.000	0.909	0.222
Pur	0.487	0.508	0.619	0.591	0.757	0.234
Qlz	0.482	0.495	0.500	0.772	0.705	0.236
Qza	0.753	0.741	0.833	0.929	0.724	0.194
Rba	0.013	0.078	0.075	0.121	0.000	0.000
Rgl	0.000	0.000	0.000	0.000	0.608	0.241
Rhe	0.430	0.458	0.632	0.914	0.639	0.567
Tec	0.716	0.707	0.776	0.615	0.718	0.154
Xpu	0.475	0.498	0.681	0.480	0.808	0.188

Formatted

"***Orthogonality" MatLab script***

```
% Clearing the workspace
clear all;
close all;
clc;

input_file_name  = 'input.xls';
output_file_name = 'out.xls';

% Loading data from file 'input_file_name'
% Data format:
% - Row 1 contains names of systems
% - Column A contains names of chemicals
% - Columns B..n contain series of measured retention times of each chemical
%   for each system.
[Data, Labels] = xlsread (input_file_name);

NumericData = Data;

rows    = size(NumericData,1);
cols   = size(NumericData,2);

% % Normalization between the smallest and greatest values for each serie
% %  - Substract the smallest element of given column from all elemets in this
column:
NumericData = NumericData - (min(NumericData)' * ones(1,rows))';
% %  - Divide each column element by maximum element in this collumn
NumericData = NumericData .* ((1./max(NumericData))' * ones(1,rows))';

dist_table = [];
idx = 0;
for A=1:(cols-1)
    for B = (A+1):cols

        idx = idx + 1;

        disp(sprintf('Mixing columns: ''%d'' and ''%d''',A, B));

        % Extracting 2 collumns
        data = [NumericData(:,A) , NumericData(:,B)];

        % Getting distances
        distances = linkage(pdist(data));
        distances = distances(:,3); % Taking only interesting data
        % Number of all distances
        N_d = size(distances,1);

        % Finding nonzero distances
        distances_nonzero = distances(logical(distances));
        %Number of nonzero distances
        N_nzd = size(distances_nonzero,1);
```

```
        % Arithmetic mean
        ArithmeticAverage = sum(distances_nonzero) / N_nzd;

        % Harmonic mean
        HarmonicAverage = N_nzd / sum(1./distances_nonzero);

        % Printing results
        fh = figure;
        plot(data(:,1), data(:,2), 'ro');
        title(sprintf('2D surface based on: ''%s'' and ''%s'' series.', ...
            Labels{1,A+1}, Labels{1,B+1}));
        xlabel(Labels{1,A+1});
        ylabel(Labels{1,B+1});
        print(fh, '-djpeg', sprintf('%d-%d.jpg',A,B));

        % Save distances
        dist_table = [ dist_table, distances];
        outcome{idx,1} = sprintf('%s-%s', Labels{1,A+1}, Labels{1,B+1});
        outcome{idx,2} = N_d;
        outcome{idx,3} = N_nzd;
        outcome{idx,4} = ArithmeticAverage;
        outcome{idx,5} = HarmonicAverage;
    end
end

% Distances table
dist_table = [[1:N_d]', dist_table];
dist_table_header = [{'di'}, {outcome{:,1}}];

% Sort values (by nonzero distances, then by Harmonic average)
outcome = sortrows(outcome,[-3 -5]);
% Insert header
outcome = [{'System', 'Total number of di', 'Number of nonzero di', ...
    'Arithmetic average', 'Harmonic average'};
    outcome];

% Print data to xls file
xls_row = 1;
xlswrite(output_file_name, dist_table_header, sprintf('A%d:%c%d', ...
    xls_row, 'A'+ idx, xls_row));
xls_row = xls_row + 1;
xlswrite(output_file_name, dist_table, sprintf('A%d:%c%d', ...
    xls_row, 'A'+ idx, xls_row + N_d -1 ));
xls_row = xls_row + N_d + 1 ;
xlswrite(output_file_name, outcome, sprintf('A%d:%c%d', ...
    xls_row, 'A'+ size(outcome,2) -1 , xls_row + idx));

disp 'Done!';
```