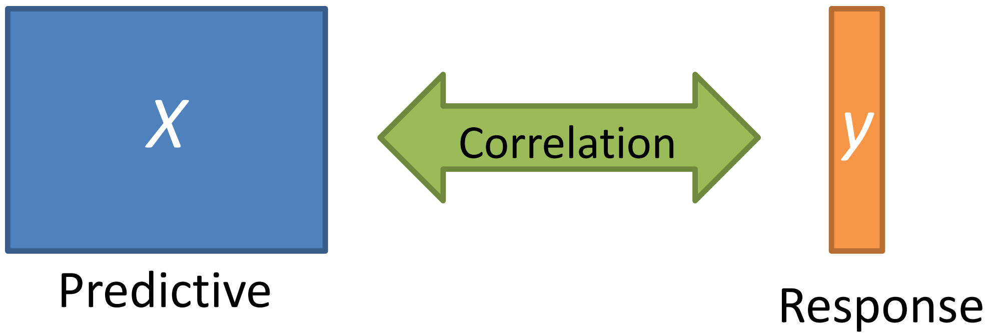
Multivariate regression

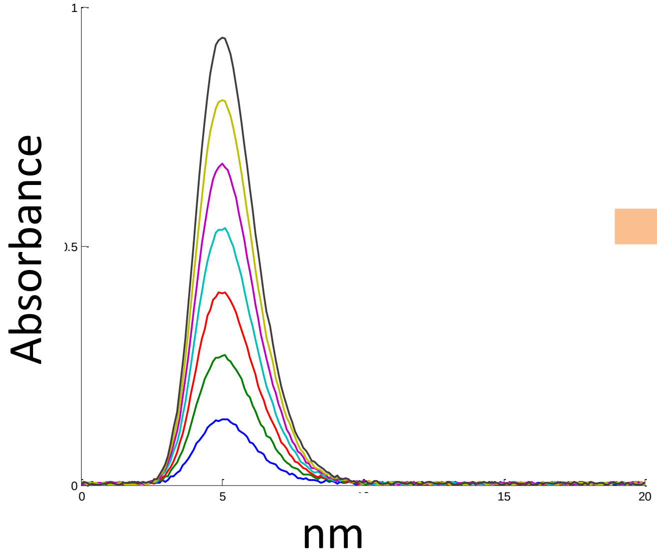
- Regression model investigates “**correlation**” between predictive and response parameters



- The information of “**correlation**” is used to establish calibration model such as MLR, PCR, PLS and ANN.

ผศ.ดร. ศิลา กิตติวัชนะ และคณะนักศึกษา
ภาควิชาเคมี คณะวิทยาศาสตร์ มหาวิทยาลัยเชียงใหม่
E-mail: silacmu@gmail.com
Tel: 087-9166692

Univariate linear regression

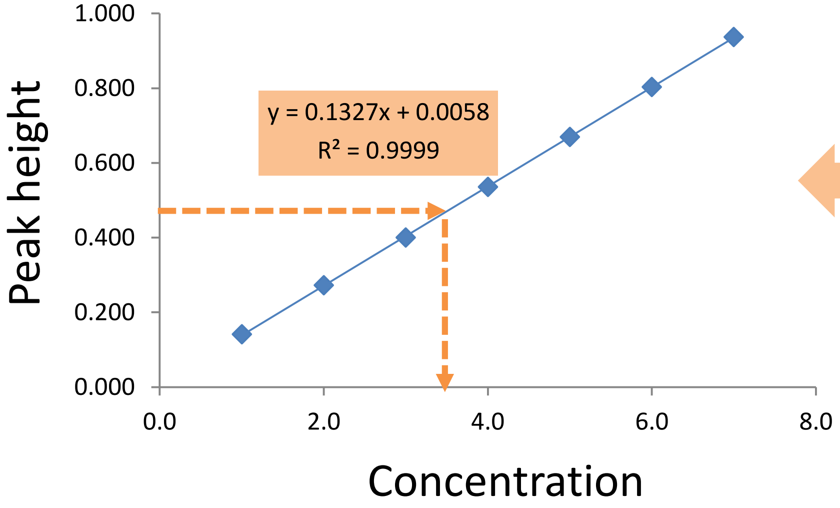


Concentration (***x***)	Peak height (***y***)
1.0	0.141
2.0	0.272
3.0	0.400
4.0	0.536
5.0	0.669
6.0	0.803
7.0	0.936

Linear equation

y = ax + c

$y = 0.1327x + 0.0058$

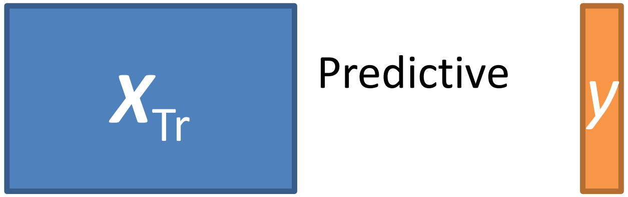


Multivariate regression

- MLR (multivariate linear regression)
- PCR (principal component regression)
- PLS (partial least squares regression)

Datasets

1) Training set is used for building modes.

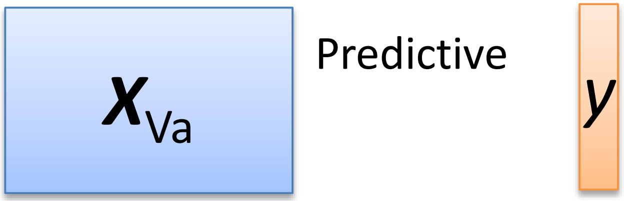


Predictive

Response ($\boldsymbol{y}_{Tr}$)

R^2, RMSEC, Bias, SEC and RPD

2) Validation set is used for investigating the model performance.

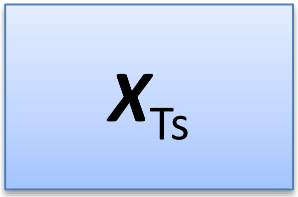


Predictive

Response ($\boldsymbol{y}_{Va}$)

Q^2, RMSEP, Bias, SEC and RPD

3) Test set is unknown samples.



Predictive

Response of the test samples ($\boldsymbol{y}_{Ts}$ or ^) is predicted from $\boldsymbol{X}_{Ts}$ using the regression models.

Q^2, RMSEP, Bias, SEC and RPD

Model statistics

- Coefficient of determination (R^2 and Q^2)

$$\boxed{R^2 = \frac{\sum_{i=1}^{I}(y_i - \bar{y})^2}{\sum_{i=1}^{I}(\hat{y}_i - \bar{y})^2}}$$

R^2 → training set (**auto prediction**)
Q^2 → test set (**prediction**)

- Root mean square error (RMSE)

$$\boxed{RMSE = \sqrt{\frac{\sum_{i=1}^{I}(y_i - \hat{y}_i)^2}{I}}}$$

RMSE of calibration (RMSEC) → training set
RMSE of prediction (RMSEP) → test set

$\hat{y}_i$ = predicted value
$\bar{y}$ = average value of y
y_i = actual value
I = number of samples

MLR

- The simplest algorithm
- Based on a simple linear regression of raw data

However,

- The number of parameters is limited.
- Usually the model suffers from multicollinear problem.

MLR

Modeling: ($\boldsymbol{y} = \boldsymbol{Xb} + \boldsymbol{E}$)

$$\boldsymbol{X}_{\mathrm{Tr}}\boldsymbol{b} = \boldsymbol{y}$$

$$\boldsymbol{X}_{\mathrm{Tr}}{}'\boldsymbol{X}_{\mathrm{Tr}}\boldsymbol{b} = \mathbf{X}_{\mathrm{Tr}}{}'\boldsymbol{y}$$

$$[\boldsymbol{X}_{\mathrm{Tr}}{}'\boldsymbol{X}_{\mathrm{Tr}}]^{-1}\,\boldsymbol{X}_{\mathrm{Tr}}{}'\boldsymbol{X}_{\mathrm{Tr}}\boldsymbol{b} = [\boldsymbol{X}_{\mathrm{Tr}}{}'\boldsymbol{X}_{\mathrm{Tr}}]^{-1}\boldsymbol{X}_{\mathrm{Tr}}{}'\boldsymbol{y}$$

$$\boldsymbol{Ib} = [\boldsymbol{X}_{\mathrm{Tr}}{}'\boldsymbol{X}_{\mathrm{Tr}}]^{-1}\boldsymbol{X}_{\mathrm{Tr}}{}'\boldsymbol{y}$$

$$\boldsymbol{b} = [\boldsymbol{X}'\boldsymbol{X}]^{-1}\boldsymbol{X}'\boldsymbol{y}$$

MLR

Given ***X***➔[*N*×*M*] and *y*➔ [*N*×1]

$$\boldsymbol{b} = [\boldsymbol{X'X}]^{-1}\boldsymbol{X'y}$$

$$[M\times 1] = [M\times N][N\times M][M\times N][N\times 1]$$

$$[M\times 1] = [M\times 1]$$

MLR

Prediction:

$$\boldsymbol{X}_{\mathrm{Ts}}\boldsymbol{b} = \boldsymbol{y}_{\mathrm{Ts}}$$

$$[N_{\mathrm{Ts}} \times M][M \times 1] = [N_{\mathrm{Ts}} \times 1]$$

Simulation data

Ideally generated data

Even	Square	Square root	Divide by 10	Inverse	Noise		Response
2	4	1.41	0.2	40	0		1
4	16	2.00	0.4	38	0		2
6	36	2.45	0.6	36	0		3
8	64	2.83	0.8	34	0		4
10	100	3.16	1	32	0		5
12	144	3.46	1.2	30	0		6
14	196	3.74	1.4	28	0		7
16	256	4.00	1.6	26	0		8
18	324	4.24	1.8	24	0		9
20	400	4.47	2	22	0		10
22	484	4.69	2.2	20	0		11
24	576	5.20	2.4	18	0		12
26	676	5.51	2.6	16	0		13
28	784	5.83	2.8	14	0		14
30	900	6.14	3	12	0		15
32	1024	6.46	3.2	10	0		16
34	1156	6.77	3.4	8	0		17
36	1296	7.08	3.6	6	0		18
38	1444	7.40	3.8	4	0		19
40	1600	7.71	4	2	0		20

Noise	Noise	Noise	Noise	Noise	Noise
4.0	0.6	1.1	0.8	0.3	3.2
3.7	6.2	3.2	2.2	8.8	7.8
1.9	1.2	3.6	0.6	2.0	2.9
7.0	8.3	2.8	0.0	6.7	5.0
0.5	0.4	0.3	2.7	0.8	2.5
10.5	2.1	2.8	3.7	8.7	8.0
2.0	2.0	1.8	4.1	0.1	1.0
10.5	3.9	1.8	8.1	7.5	3.0
1.4	1.2	0.0	4.3	4.4	3.5
6.8	2.2	11.1	7.2	1.4	0.1
1.8	1.4	1.0	1.0	2.0	1.0
3.8	3.7	10.8	6.3	5.8	6.2
3.8	1.9	0.0	2.2	4.4	1.7
9.9	4.7	9.5	5.1	4.0	3.2
1.8	0.5	0.9	1.7	3.5	4.1
5.1	6.3	8.4	3.3	1.1	10.9
1.8	2.3	0.7	2.4	0.0	3.9
10.5	0.6	2.0	5.8	1.7	10.5
1.6	3.2	1.2	1.2	3.1	1.8
0.4	6.4	4.2	8.8	2.3	4.5

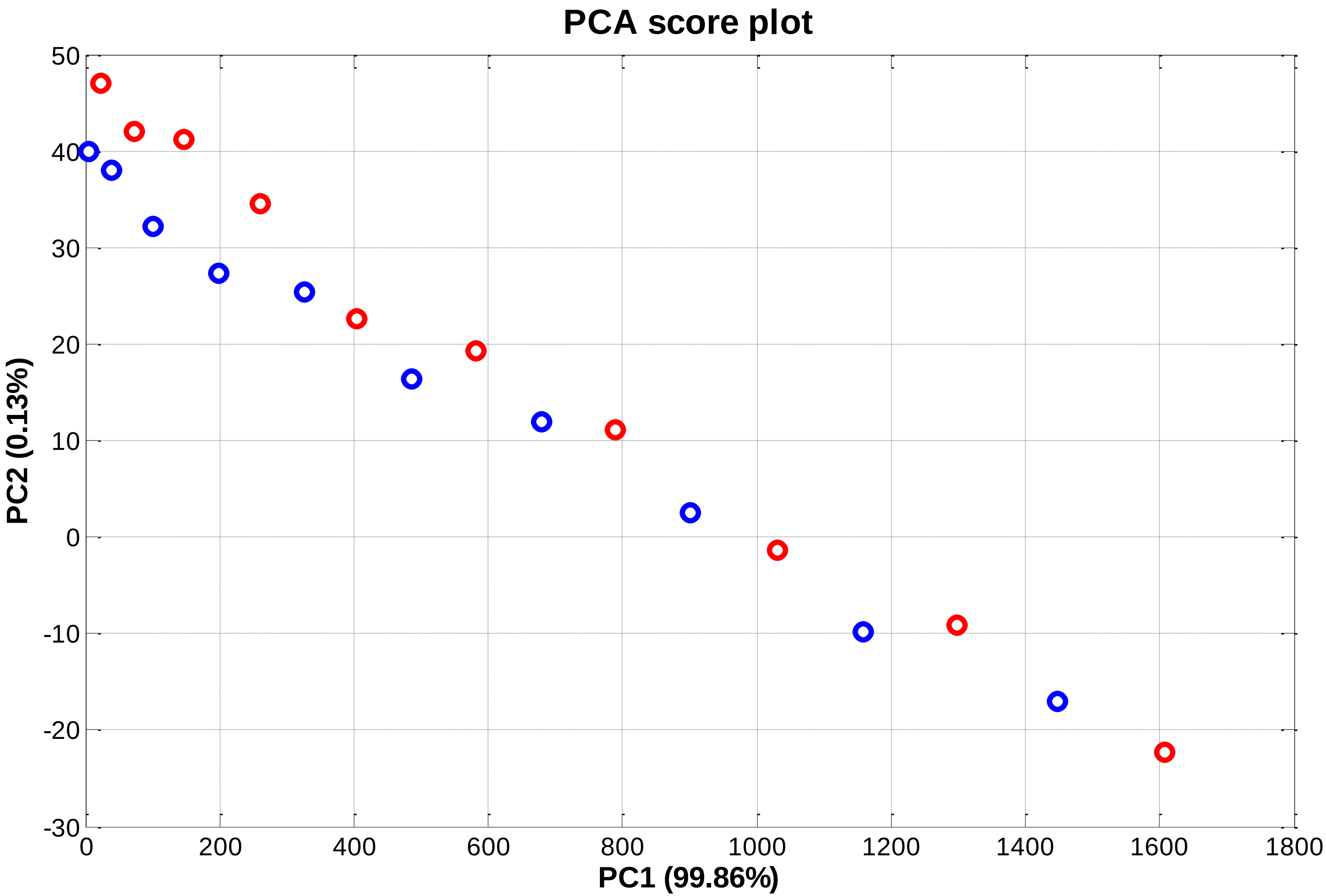
Noise

Ideally generated data + Noise

Even	Square	Square root	Divide by 10	Inverse	Noise		Response
5.96	4.64	2.52	1.00	40.31	3.23		1
7.68	22.16	5.18	2.58	46.82	7.78		2
7.86	37.19	6.04	1.23	37.99	2.95		3
15.04	72.29	5.62	0.81	40.66	4.99		4
10.51	100.40	3.50	3.74	32.79	2.52		5
22.53	146.07	6.27	4.86	38.65	8.03		6
16.03	197.96	5.54	5.52	28.12	1.00		7
26.52	259.89	5.78	9.69	33.47	2.97		8
19.37	325.18	4.26	6.09	28.36	3.53		9
26.84	402.17	15.54	9.23	23.43	0.11		10
23.83	485.36	5.70	3.21	21.97	1.04		11
27.85	579.72	16.03	8.68	23.84	6.16		12
29.81	677.94	5.52	4.81	20.39	1.69		13
37.87	788.67	15.30	7.88	18.02	3.23		14
31.84	900.54	7.00	4.72	15.48	4.07		15
37.12	1030.35	14.89	6.53	11.07	10.95		16
35.78	1158.26	7.42	5.79	8.03	3.91		17
46.46	1296.56	9.11	9.40	7.71	10.49		18
39.65	1447.23	8.63	5.01	7.11	1.84		19
40.38	1606.40	11.88	12.81	4.29	4.54		20

Training samples

Test samples


PCA score plot
PC2 (0.13%)
PC1 (99.86%)
50
40
30
20
10
0
-10
-20
-30
0
200
400
600
800
1000
1200
1400
1600
1800

$$\begin{bmatrix} 5.96 & 4.64 & 2.52 & 0.99 & 40.3 & 3.23 \\ 7.86 & 37.9 & 6.04 & 1.23 & 37.9 & 2.95 \\ \vdots & \vdots & X_{Tr} & \vdots & \vdots & \vdots \\ 35.8 & 1158 & 7.42 & 5.79 & 8.03 & 3.91 \\ 39.6 & 1447 & 8.63 & 5.01 & 7.11 & 1.84 \end{bmatrix} \cdot \begin{bmatrix} b \end{bmatrix} = \begin{bmatrix} 1 \\ 3 \\ \vdots \\ 17 \\ 19 \end{bmatrix}$$

$$[b] = [0.40 \quad 0.003 \quad 0.17 \quad 0.19 \quad -0.047 \quad 0.06]$$

$$\begin{bmatrix} 7.68 & 22.1 & 5.18 & 2.58 & 46.8 & 7.77 \\ 15.0 & 72.3 & 5.62 & 0.81 & 40.6 & 4.99 \\ \vdots & \vdots & X_{Ts} & \vdots & \vdots & \vdots \\ 46.5 & 1296 & 9.11 & 9.40 & 7.71 & 10.5 \\ 40.4 & 1606 & 11.9 & 12.8 & 4.29 & 4.54 \end{bmatrix} \cdot \begin{bmatrix} b \end{bmatrix} = \begin{bmatrix} ? \\ ? \\ \vdots \\ ? \\ ? \end{bmatrix}$$

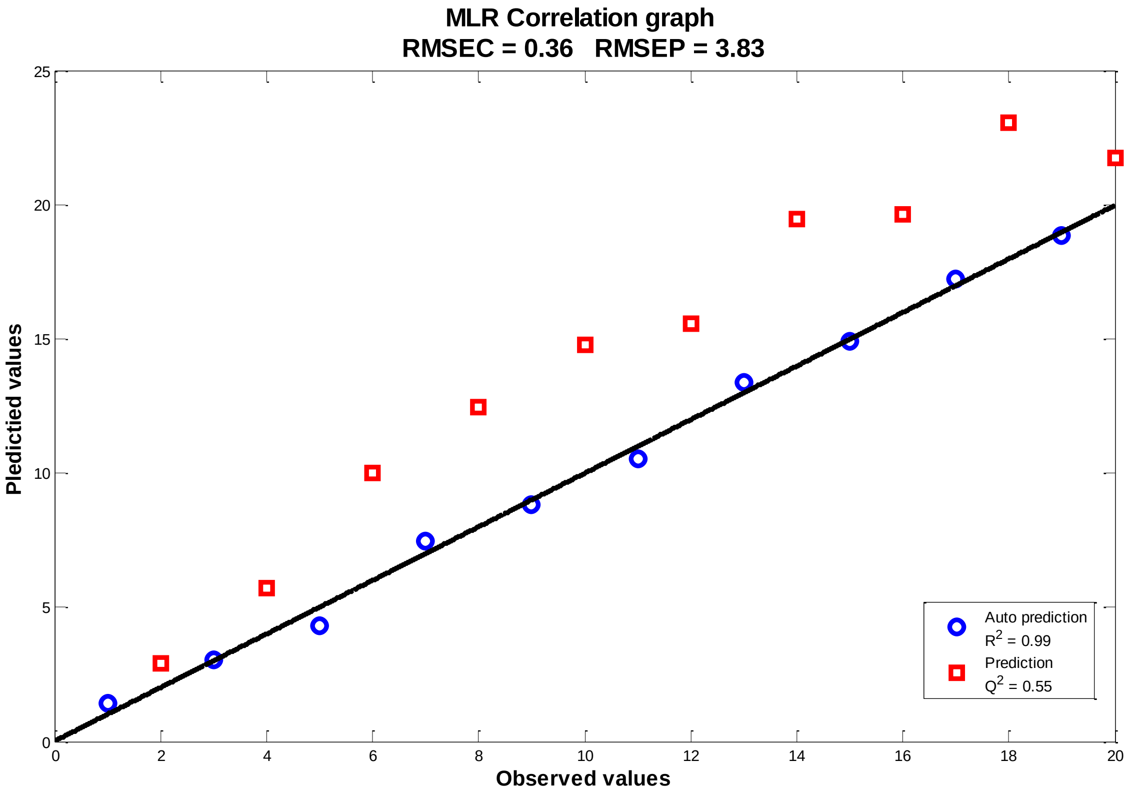
Prediction of the simulation data using MLR

Training samples

Expected	Predicted	Error	[Error]^2
1	1.40	0.40	0.162
3	3.02	0.02	0.000
5	4.32	-0.68	0.463
7	7.46	0.46	0.210
9	8.84	-0.16	0.027
11	10.53	-0.47	0.224
13	13.38	0.38	0.144
15	14.92	-0.08	0.006
17	17.23	0.23	0.051
19	18.87	-0.13	0.018
	Sum	-0.0391	1.3053
		RMSEC	0.36

Test samples

Expected	Predicted	Error	[Error]^2
2	2.92	0.92	0.854
4	5.73	1.73	3.009
6	9.99	3.99	15.890
8	12.44	4.44	19.719
10	14.79	4.79	22.984
12	15.59	3.59	12.856
14	19.45	5.45	29.678
16	19.66	3.66	13.380
18	23.03	5.03	25.305
20	21.76	1.76	3.081
	Sum	35.3567	146.7566
		RMSEP	3.83



PCR

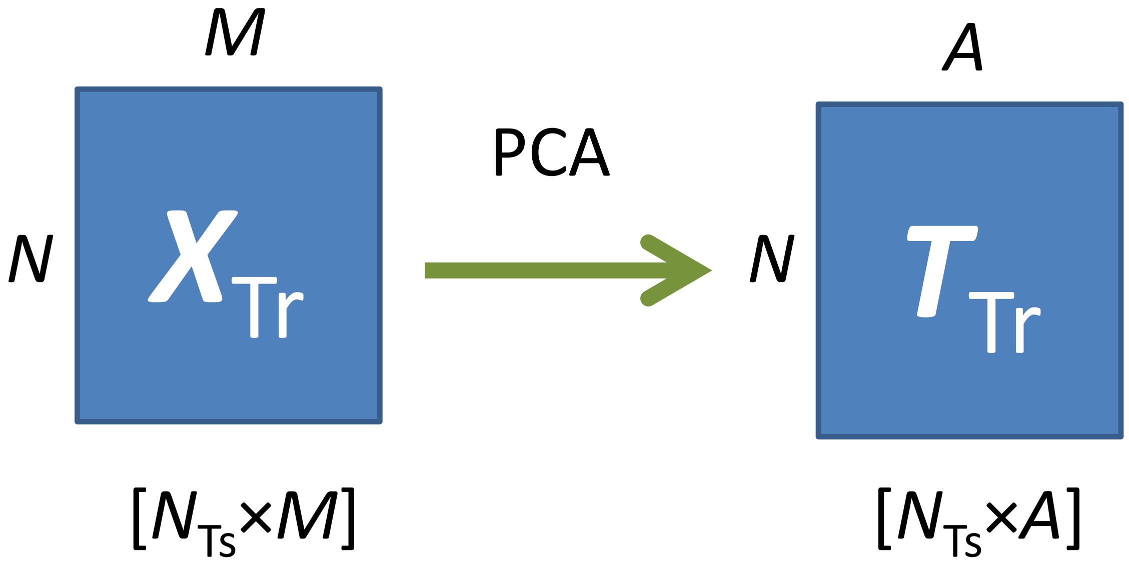
- A little bit more complicate algorithm
- Based on a simple linear regression of the score (***T***) of the raw data

So,

- The number of parameters is not limited but the number of PCs used is still limited and should be carefully defined.

PCR

Modeling:



M

N $\boldsymbol{X}_{\text{Tr}}$

$\xrightarrow{\text{PCA}}$

A

N $\boldsymbol{T}_{\text{Tr}}$

$[N_{\text{Ts}} \times M]$

$[N_{\text{Ts}} \times A]$

PCR

Modeling:

$$\boldsymbol{T}_{\mathrm{Tr}}\boldsymbol{b} = \boldsymbol{y}$$

$$\boldsymbol{T}_{\mathrm{Tr}}'\boldsymbol{T}_{\mathrm{Tr}}\boldsymbol{b} = \mathrm{T}_{\mathrm{Tr}}'\boldsymbol{y}$$

$$[\boldsymbol{T}_{\mathrm{Tr}}'\boldsymbol{T}_{\mathrm{Tr}}]^{-1}\,\boldsymbol{T}_{\mathrm{Tr}}'\boldsymbol{T}_{\mathrm{Tr}}\boldsymbol{b} = [\boldsymbol{T}_{\mathrm{Tr}}'\boldsymbol{T}_{\mathrm{Tr}}]^{-1}\boldsymbol{T}_{\mathrm{Tr}}'\boldsymbol{y}$$

$$\boldsymbol{I}\boldsymbol{b} = [\boldsymbol{T}_{\mathrm{Tr}}'\boldsymbol{T}_{\mathrm{Tr}}]^{-1}\boldsymbol{T}_{\mathrm{Tr}}'\boldsymbol{y}$$

$$\boldsymbol{I}\boldsymbol{b} = [\boldsymbol{T}'\boldsymbol{T}]^{-1}\boldsymbol{T}'\boldsymbol{y}$$

PCR

Prediction:

Step 1: Estimate the scores of the test data ($\boldsymbol{X} = \boldsymbol{TP}'$).

Step 2: Predict the ***y*** vector using the established coefficient vector (***b***)

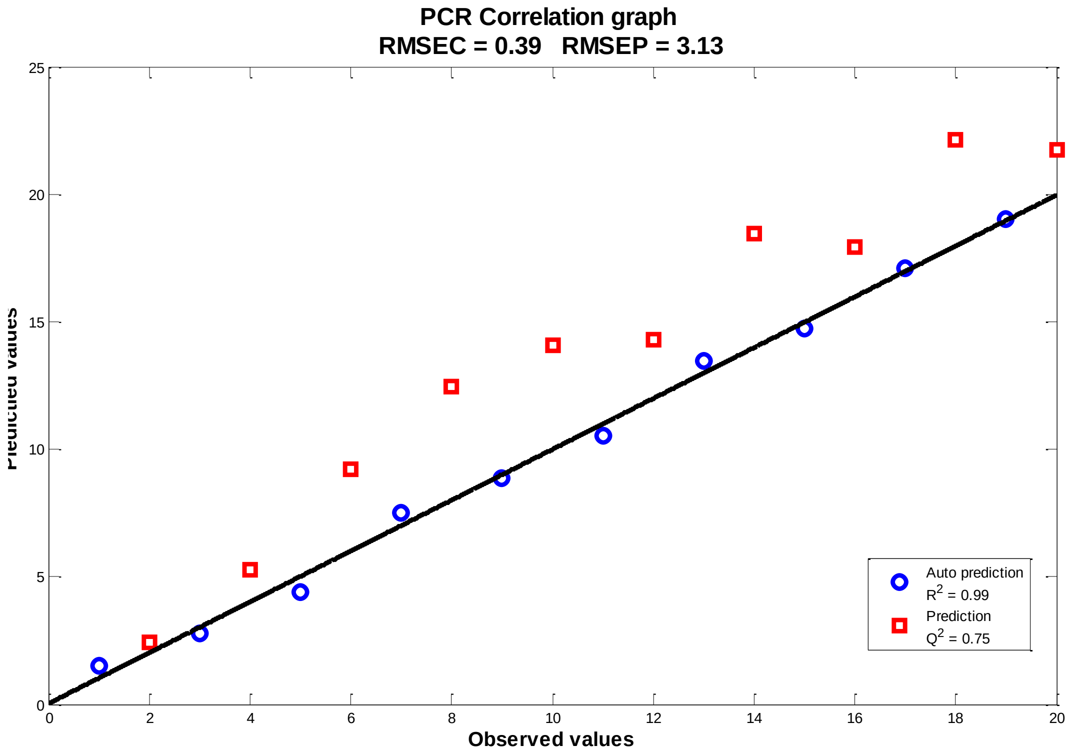
Prediction of the simulation data using PCR

Training samples

Expected	Predicted	Error	[Error]^2
1	1.51	0.51	0.263
3	2.78	-0.22	0.049
5	4.42	-0.58	0.339
7	7.53	0.53	0.276
9	8.86	-0.14	0.018
11	10.53	-0.47	0.220
13	13.48	0.48	0.229
15	14.71	-0.29	0.083
17	17.10	0.10	0.010
19	19.03	0.03	0.001
	Sum	-0.0442	1.4903
		RMSEC	0.39

Test samples

Expected	Predicted	Error	[Error]^2
2	2.45	0.45	0.203
4	5.27	1.27	1.603
6	9.22	3.22	10.388
8	12.46	4.46	19.881
10	14.07	4.07	16.564
12	14.30	2.30	5.277
14	18.47	4.47	19.987
16	17.94	1.94	3.773
18	22.15	4.15	17.198
20	21.74	1.74	3.043
	Sum	28.0705	97.9177
		RMSEP	3.13

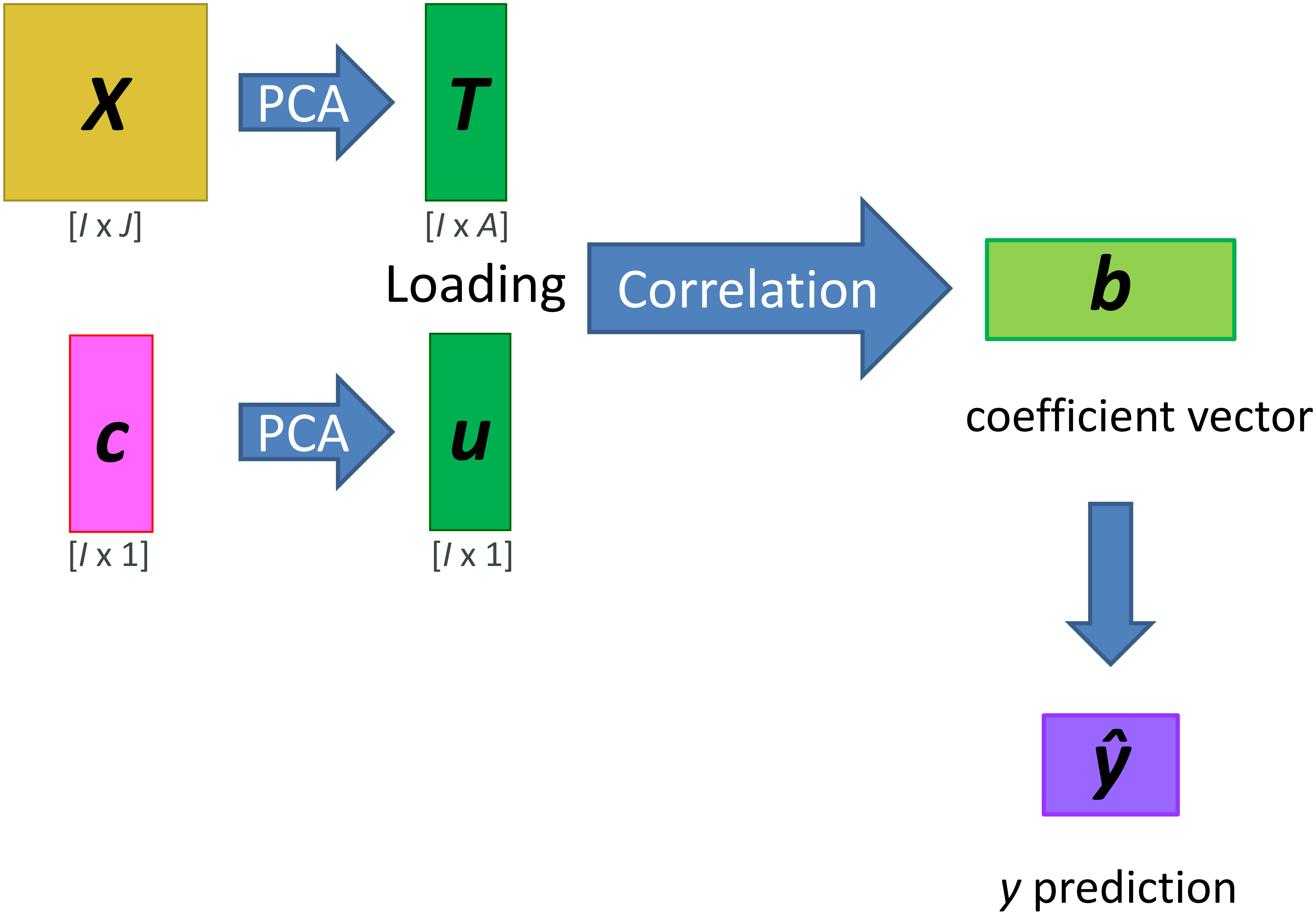


PLS

Prediction:

Step 1: Estimate the scores of the test data (***X*** = ***TP***’).

Step 2: Predict the ***y*** vector using the established coefficient vector (***b***)



X

[*I* x *J*]

PCA

T

[*I* x *A*]

Loading

Correlation

b

coefficient vector

c

[*I* x 1]

PCA

u

[*I* x 1]

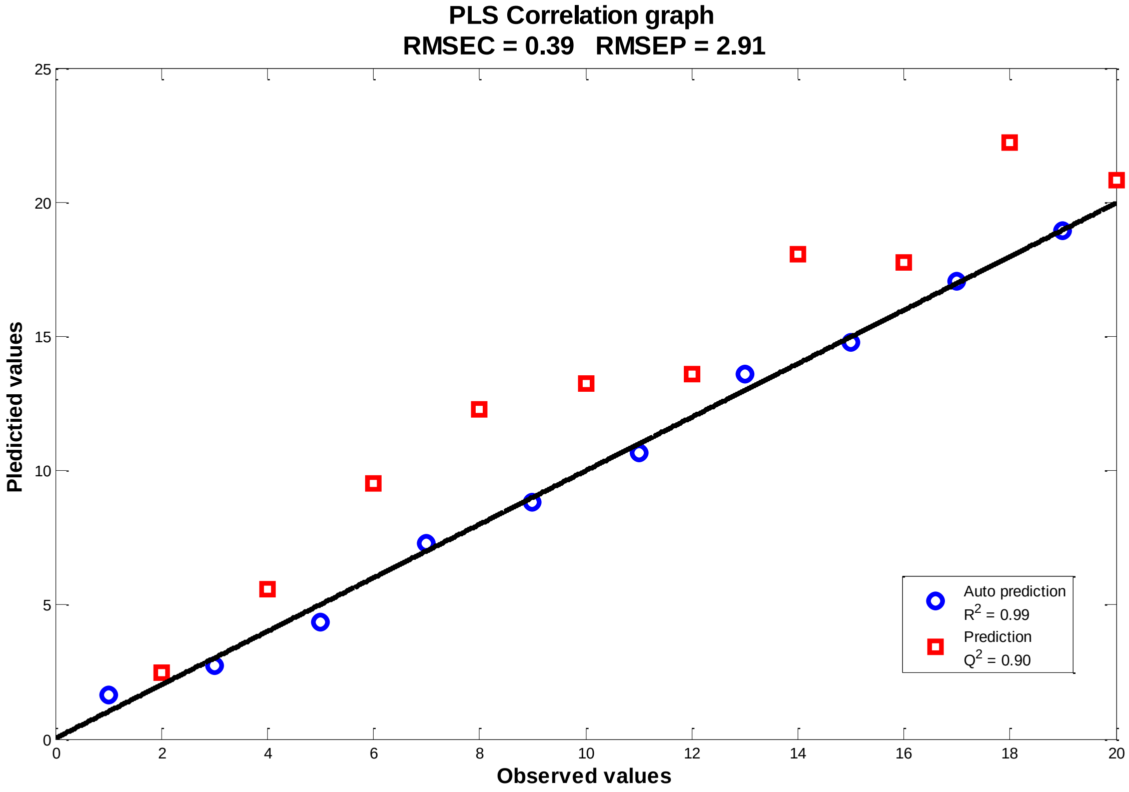
$\hat{y}$

y prediction

Prediction of the simulation data using PLS

Training samples

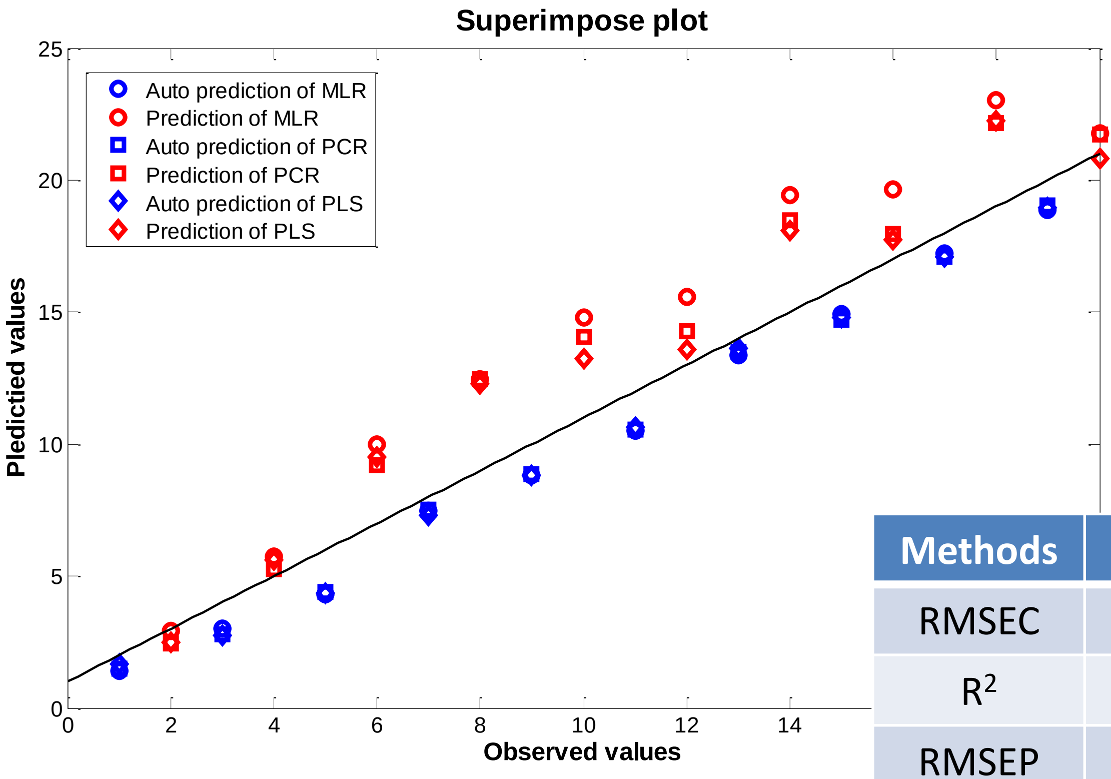
Expected	Predicted	Error	[Error]^2
1	1.65	0.65	0.419
3	2.74	-0.26	0.070
5	4.36	-0.64	0.412
7	7.31	0.31	0.096
9	8.81	-0.19	0.037
11	10.65	-0.35	0.124
13	13.62	0.62	0.380
15	14.80	-0.20	0.042
17	17.07	0.07	0.006
19	18.95	-0.05	0.003
	Sum	-0.0604	1.5875
		RMSEC	0.398



Test samples

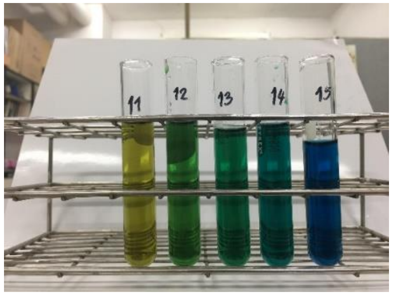
Expected	Predicted	Error	[Error]^2
2	2.49	0.49	0.239
4	5.60	1.60	2.566
6	9.51	3.51	12.295
8	12.30	4.30	18.508
10	13.24	3.24	10.498
12	13.60	1.60	2.573
14	18.08	4.08	16.610
16	17.74	1.74	3.036
18	22.24	4.24	17.945
20	20.82	0.82	0.673
	Sum	25.6180	84.9425
		RMSEP	2.91

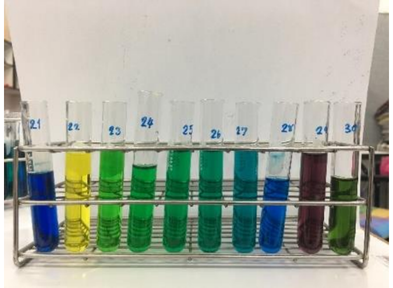
Comparison of the prediction performance using MLR PCR and PLS

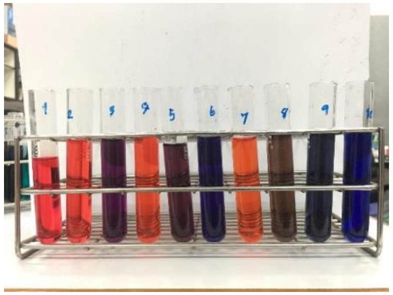


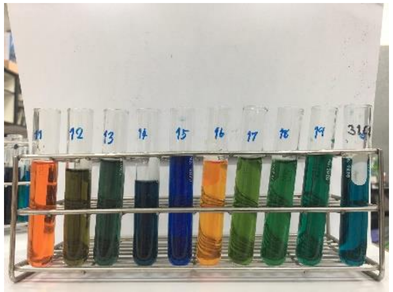
Methods	MLR	PCR	PLS
RMSEC	0.09	0.09	0.09
R^2	0.99	0.99	0.99
RMSEP	3.70	2.67	0.31
Q^2	0.55	0.75	0.90

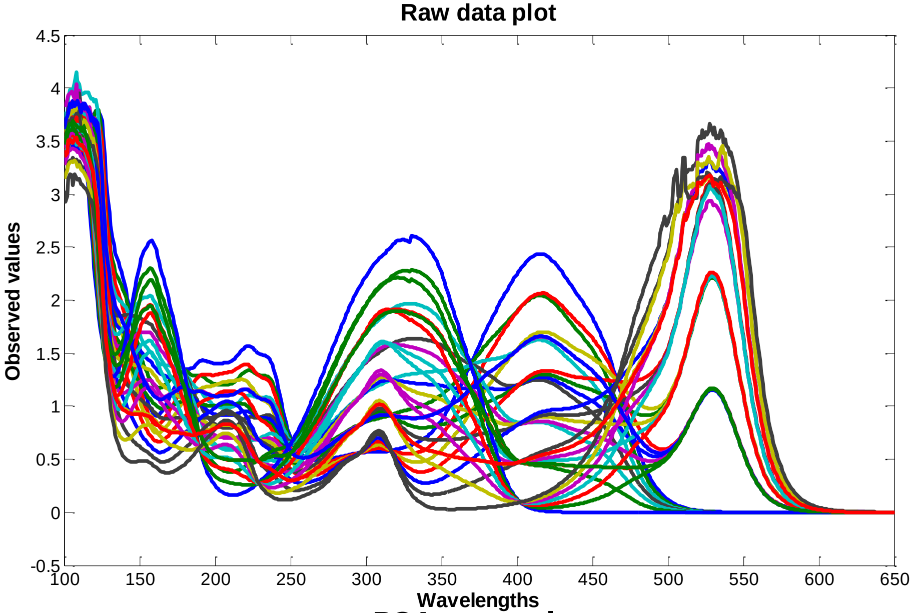
Food colorant samples

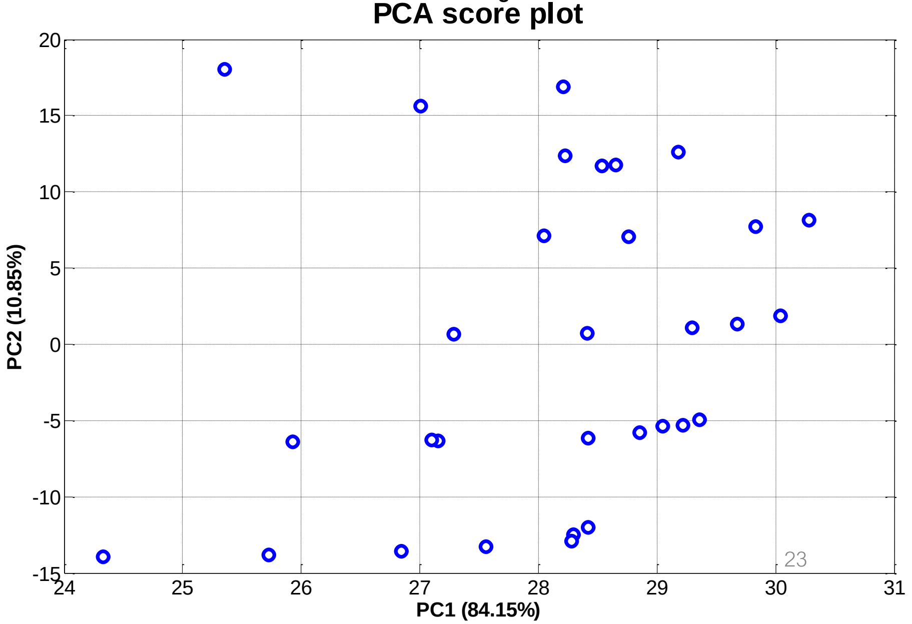




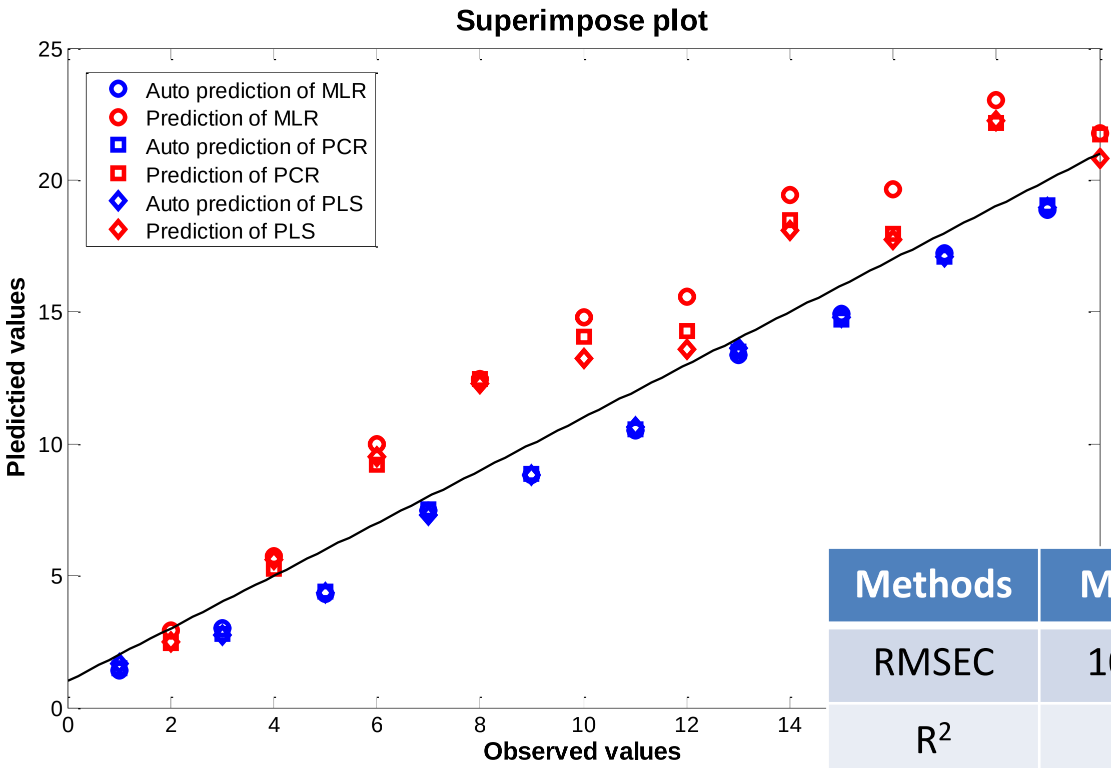








Prediction performance of the food colorant data using MLR PCR and PLS



Methods	MLR	PCR	PLS
RMSEC	108	0.74	0.28
R^2	-	0.77	0.92
RMSEP	122	0.86	0.36
Q^2	-	0.75	0.76