Display RGB/CMF

Reading the Dataset

```
warning('off','all');
lms_dataset = readtable("HW_Opponency_Data.xlsx",Sheet="LMS");
rgb_spd_dataset = readtable("HW_DisplaySPD_Data.xlsx",Sheet="DisplaySPD");
cie_dataset = readtable("HW_DisplaySPD_Data.xlsx",Sheet="CIE 1931");
```

Question 1: Compute the LMS values of the display's three primaries

1. Create LMS relative sensitivity matrix
2. Create Spectral radiance matrix
3. Find the change in wavelength
4. Calculate LMS of the display using the equation

$$\begin{pmatrix} L_R & L_G & L_B \\ M_R & M_G & M_B \\ S_R & S_G & S_B \end{pmatrix} = \begin{pmatrix} l_\lambda & \dots & l_\lambda \\ m_\lambda & \dots & m_\lambda \\ s_\lambda & \dots & s_\lambda \end{pmatrix} \begin{pmatrix} L_{\lambda,r} & \dots & L_{\lambda,r} \\ L_{\lambda,g} & \dots & L_{\lambda,g} \\ L_{\lambda,b} & \dots & L_{\lambda,b} \end{pmatrix}'$$

```
lms_matrix = transpose(lms_dataset{:,[2:4]});

rgb_spd = transpose(rgb_spd_dataset{:,[2:4]});

wavelength = lms_dataset{:,1};
d_lambda = mean(diff(wavelength));

LMS_RGB = calc_LMS_source(lms_matrix,transpose(rgb_spd),d_lambda);
```

Generating Table of LMS for the Display (Only for Table Visualisation)

```
Table_LMS = array2table(LMS_RGB);
Table_LMS.Properties.VariableNames(1:3) = {'L','M','S'};
```

Question 1: Answer

```
Table_LMS
```

Table_LMS = 3×3 table

	L	M	S
1	0.1526	0.3782	0.0352
2	0.0484	0.3861	0.0522
3	0.0054	0.0357	0.3071

Question 2a: Compute the color matching functions (CMFs, also known as r-bar, g-bar, b-bar) of the display's primaries

$$\begin{pmatrix} r_\lambda & \cdots & r_\lambda \\ g_\lambda & \cdots & g_\lambda \\ b_\lambda & \cdots & b_\lambda \end{pmatrix} = \begin{pmatrix} L_R & L_G & L_B \\ M_R & M_G & M_B \\ S_R & S_G & S_B \end{pmatrix}^{-1} \begin{pmatrix} l_\lambda & \cdots & l_\lambda \\ m_\lambda & \cdots & m_\lambda \\ s_\lambda & \cdots & s_\lambda \end{pmatrix}$$

```
CMF_RGB = LMS_RGB\lms_matrix;
CMF_RGB_norm = custom_normalization(CMF_RGB);
```

Generating Table of CMF for the Display (Only for Table Visualisation)

```
CMF_RGB_norm_transpose = transpose(CMF_RGB_norm);
Table_CMF = array2table(CMF_RGB_norm_transpose);
Table_CMF.Properties.VariableNames(1:3) = {'R','G','B'};
```

Question 2a: Answer

```
Table_CMF
```

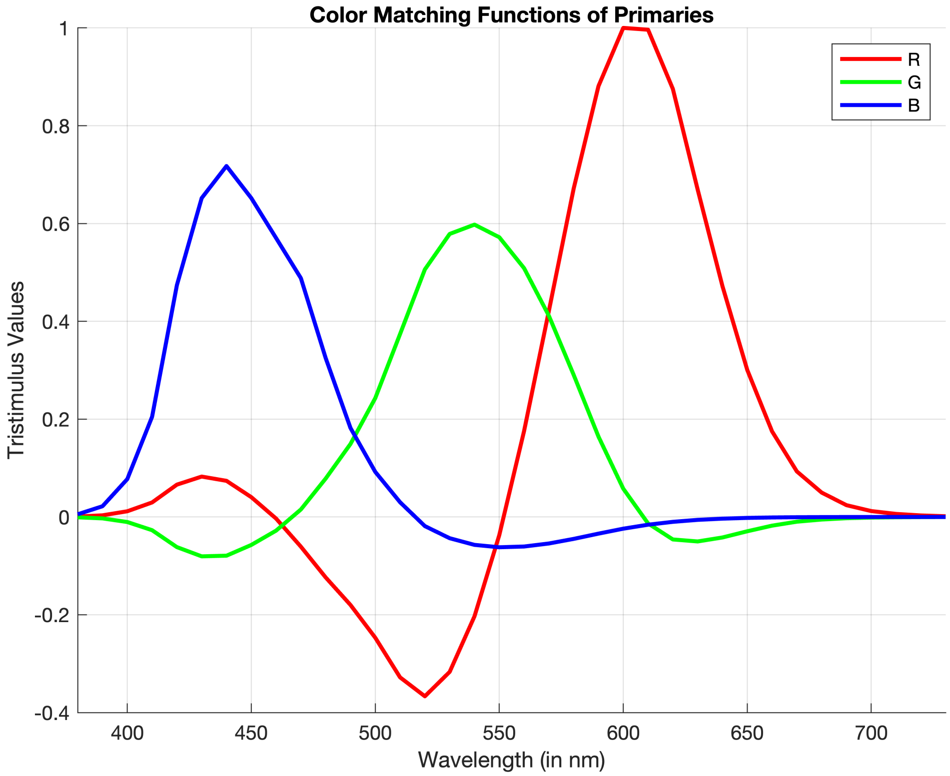
Table_CMF = 36×3 table

	R	G	B
1	0.0008	-0.0007	0.0055
2	0.0034	-0.0029	0.0219
3	0.0116	-0.0103	0.0774
4	0.0297	-0.0270	0.2048
5	0.0661	-0.0616	0.4739
6	0.0826	-0.0805	0.6520
7	0.0739	-0.0792	0.7176
8	0.0405	-0.0572	0.6520
9	-0.0033	-0.0280	0.5703
10	-0.0607	0.0153	0.4883
11	-0.1236	0.0785	0.3249
12	-0.1799	0.1493	0.1820
13	-0.2469	0.2430	0.0922
14	-0.3280	0.3740	0.0301
15	-0.3667	0.5061	-0.0187
16	-0.3168	0.5787	-0.0435
17	-0.2032	0.5978	-0.0571
18	-0.0372	0.5720	-0.0620

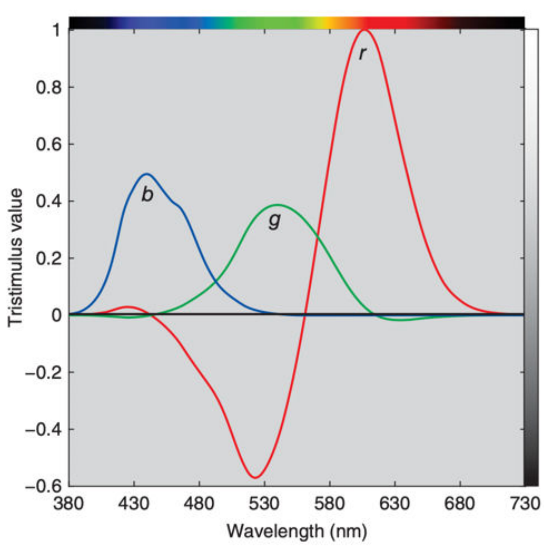
	R	G	B
19	0.1756	0.5087	-0.0604
20	0.4212	0.4114	-0.0542
21	0.6714	0.2906	-0.0448
22	0.8809	0.1644	-0.0341
23	1	0.0578	-0.0240
24	0.9963	-0.0132	-0.0159
25	0.8752	-0.0460	-0.0100
26	0.6695	-0.0501	-0.0059
27	0.4720	-0.0419	-0.0035
28	0.3006	-0.0293	-0.0019
29	0.1750	-0.0178	-0.0010
30	0.0931	-0.0097	-0.0005
31	0.0499	-0.0053	-0.0003
32	0.0241	-0.0026	-0.0001
33	0.0121	-0.0013	-0.0001
34	0.0062	-0.0007	-0
35	0.0031	-0.0003	-0
36	0.0015	-0.0001	-0

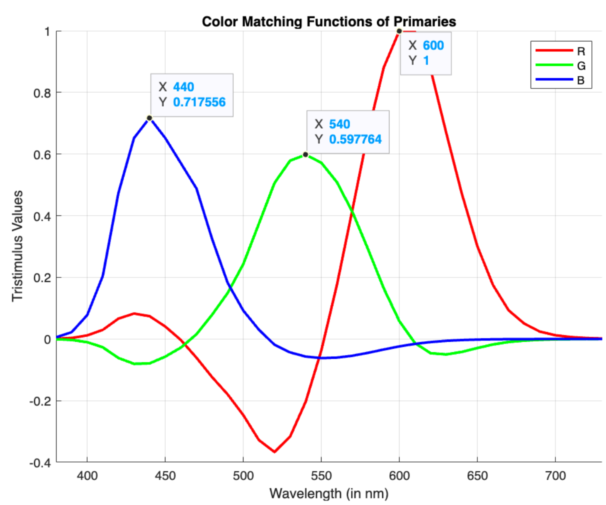
Question 2b: Plotting CMFs versus wavelength

```
figure(1)
colors=['r','g','b'];
labels = ['Red','Green','Blue'];
for i = 1:3
    hold on
    plot(lms_dataset{:,1},CMF_RGB_norm_transpose(:,i), ...
        "LineWidth",2,"Color",colors(i))
end
grid on
hold off
xlabel('Wavelength (in nm)')
ylabel('Tristimulus Values')
xlim([380 730])
legend('R','G','B');
title('Color Matching Functions of Primaries')
```



Question 3: Comparing the book's Fig 4.14 (p 56) to our plot (right side)





Color-matching functions of the Display primaries might result from using 440nm (blue), 540nm (green) and 600m (red). In the book's example, more amount of red is added in the fixed field for color matching and blue is almost not added in the fixed field. Whereas, in our computed plot for the display, more amount of blue light is added in the fixed field for color matching and is represented as negative value in the graph. This might be because of it's increased energy nature. Emitting more blue light enhances the visual quality of the display through increased brightness..

Question 4: 3x3 matrix to linearly transform the computedCMFs to approximate the CIE 1931 standard colorimetric observer

1. *r*, *g*, and *b*, were normalized to unit area
2. Deriving *V* from y-bar value of CIE 1931 Data source (HW_DisplaySPD_Data.xlsx)
3. Computing 3x3 matrix (M_function) using the following equation Eq 4.12

$$(0.176\,90\ \ 0.812\,40\ \ 0.010\,63) = \begin{pmatrix} V_\lambda & \ldots & V_\lambda \end{pmatrix} \begin{pmatrix} \bar{r}_\lambda & \ldots & \bar{r}_\lambda \\ \bar{g}_\lambda & \ldots & \bar{g}_\lambda \\ \bar{b}_\lambda & \ldots & \bar{b}_\lambda \end{pmatrix}^{+} \tag{4.12}$$

```
CIE_source = cie_dataset{:,2:4};

r_bar_norm = normalize(CMF_RGB(1,:),'norm',1);
g_bar_norm = normalize(CMF_RGB(2,:),'norm',1);
b_bar_norm = normalize(CMF_RGB(3,:),'norm',1);
rgb_bar_norm = [r_bar_norm;g_bar_norm;b_bar_norm];

V_lamda = transpose(CIE_source(:,2));

M_function = transpose(CIE_source)*pinv(rgb_bar_norm);
```

Question 4: Answer

```
M_function
```

```
M_function = 3×3
    9.1919    4.4922    2.2744
    4.7800    9.1800    0.8955
    0.3086    1.4809   11.8684
```

Question 5: Plot the computed approximation together on the same axes with the actual CIE 1931 functions

1. Calculating the approximation using the equation 4.15

$$\begin{pmatrix} \bar{x}_\lambda & \ldots & \bar{x}_\lambda \\ \bar{y}_\lambda & \ldots & \bar{y}_\lambda \\ \bar{z}_\lambda & \ldots & \bar{z}_\lambda \end{pmatrix} = \begin{pmatrix} 0.49 & 0.31 & 0.20 \\ 0.176\,97 & 0.812\,40 & 0.010\,63 \\ 0.00 & 0.01 & 0.99 \end{pmatrix} \cdot \begin{pmatrix} \bar{r}_\lambda & \ldots & \bar{r}_\lambda \\ \bar{g}_\lambda & \ldots & \bar{g}_\lambda \\ \bar{b}_\lambda & \ldots & \bar{b}_\lambda \end{pmatrix} \begin{pmatrix} n_\lambda & & \\ & \ddots & \\ & & n_\lambda \end{pmatrix} \tag{4.14}$$

2. To fix the residual error, we are multiplying with *n* and *n* is calculated using equation 4.16

$$n_\lambda = V_\lambda / (0.176\,90\bar{r}_\lambda + 0.812\,40\bar{g}_\lambda + 0.010\,63\bar{b}_\lambda) \quad (4.15)$$

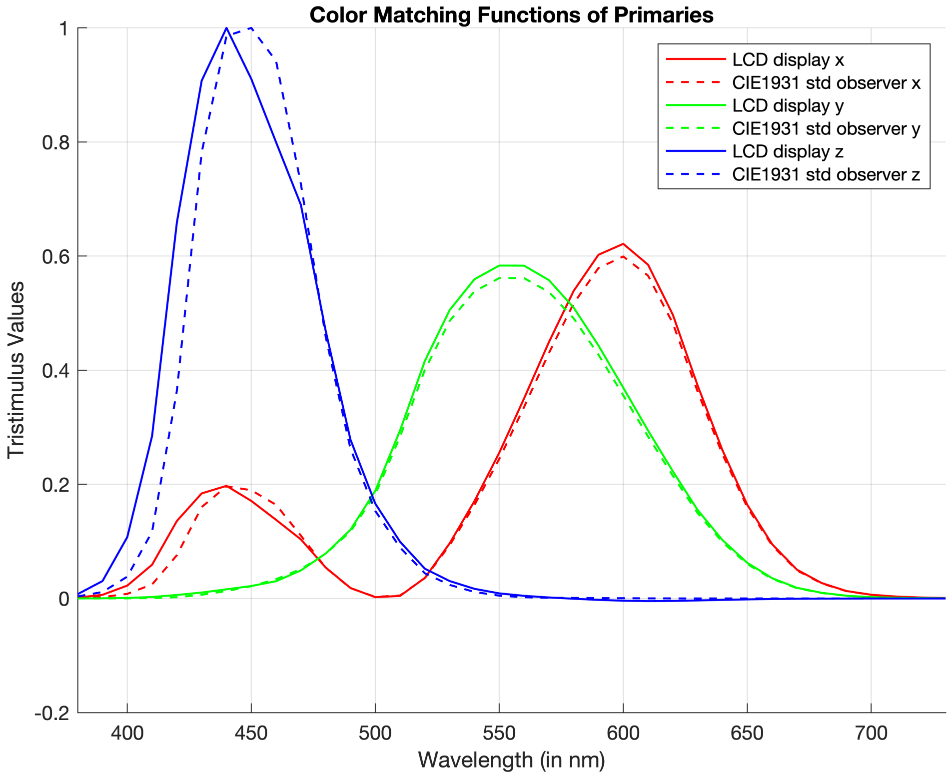
```
% Calculating n_lamda to fix the residual error
n_denominator = M_function(2,1)*(rgb_bar_norm(1,:)) + ...
    M_function(2,2)*(rgb_bar_norm(2,:)) + M_function(2,3)*(rgb_bar_norm(3,:));
n_lamda = V_lamda/n_denominator;
n_lamda_diagonal = diag(ones(36,1)*n_lamda);

% Calculate xyz-bar for the displays
xyz_bar = calc_CIE(M_function,rgb_bar_norm,n_lamda_diagonal);
xyz_bar_norm = (custom_normalization(xyz_bar));
```

Plotting the graph for computed LCD primaries against CIE 1931 standard colorimetric observer.

```
CIE_source_norm = custom_normalization(CIE_source);
xyz_bar_norm_transpose = transpose(xyz_bar_norm);

figure(2)
colors=['r','g','b'];
for i = 1:3
    hold on
    plot(lms_dataset{:,1},xyz_bar_norm_transpose(:,i),"LineWidth",1, ...
        "Color",colors(i));
    plot(lms_dataset{:,1},CIE_source_norm(:,i),"LineWidth",1,"Color", ...
        colors(i),"LineStyle","--");
end
grid on
hold off
xlabel('Wavelength (in nm)');
ylabel('Tristimulus Values');
xlim([lms_dataset{1,1} lms_dataset{end,1}]);
title('Color Matching Functions of Primaries');
legend('LCD display x','CIE1931 std observer x','LCD display y', ...
    'CIE1931 std observer y','LCD display z','CIE1931 std observer z');
```



On comparing the computed LCD primaries against CIE 1931 standard colorimetric observer, as the wavelength increase, especially after 500nm, all the primaries of LCD appears to have higher tristimulus value. So, for shorter wavelength, the the weighing used from *V* for calculating might be an issue. Similar case was found with Gibson and Tyndall (mentioned in page 62)

Function used:

Function to normalize a matrix with its peak value

```
function n = custom_normalization(x)
    max_value = max(x, [], 'all');
    n = x/max_value;
end
```

Function to calculate LMS of light source

```
function T = calc_LMS_source(t,s,d_lambda)
    T = t*s*d_lambda;
end
```

Function to calculate the approximation of Primaries with respect to CIE 1931 Standard colorimetric observer

```
function CIE = calc_CIE(M,rgb_bar,n_lamda)
    CIE = M*(rgb_bar*n_lamda);
end
```