```
%B11102112 李家睿
```

```
% Clear workspace and close all figures
clear, close all
% Define the variable names expected in each .mat file
VarName={'name','age','gender','height','weight','predata','postdata'};
% Define the directory containing the data files
dirname=[pwd filesep 'data'];
% Get a list of all files in the directory
dirinfo=dir(dirname);
% Remove the first three entries ('.', '..', and possibly hidden system files)
 dirinfo(1:3)=[];
 % Loop through each directory (assumed to contain a .mat file)
 for i=1:length(dirinfo)
    % Load the .mat file from the corresponding directory
     load([dirname filesep dirinfo(i).name filesep dirinfo(i).name '.mat'])
    % Assign the loaded data to a struct array NC
    for j=1:length(VarName)
        eval(['NC(i).' VarName{j} '=' VarName{j} ';']);
    end
 end
 % Save the NC struct to a .mat file for later use
 save('NC.mat','NC')


% Separate the data based on gender (1 = male, 0 = female)
male_data = NC([NC.gender]==1);
female_data = NC([NC.gender]==0);
count_gender_1 = sum([NC.gender]==1);
count_gender_0 = sum([NC.gender]==0);


% Initialize a structure to store statistical results
stats = struct();
% Define the attributes to analyze (age, height, weight)
info = {'age', 'height','weight'};

% Define gender categories for iteration
genders = {'male', 'female'};
% Compute mean and standard deviation for each gender and attribute
for g = 1:length(genders);
    % Dynamically retrieve the corresponding gender dataset
    data = eval([genders{g} '_data']);
    % Loop through each attribute
    for i = 1:length(info);
        info_name = info{i};
         % Compute mean and standard deviation for the attribute
        stats.(genders{g}).(info_name).mean = mean([data.(info_name)]);
        stats.(genders{g}).(info_name).std = std([data.(info_name)]);
```

```
    end
end
% Display the NC struct (all loaded data)
NC
```

NC = 1×30 struct

Fields	name	age	gender	height	weight	predata	postdata
1	'NC01'	67	1	168.9951	79.9885	8×450 double	8×450 double
2	'NC02'	59	1	170.9523	63.6980	8×450 double	8×450 double
3	'NC03'	57	1	161.8978	52.8100	8×450 double	8×450 double
4	'NC04'	63	1	168.9596	57.9199	8×450 double	8×450 double
5	'NC05'	68	0	167.8267	62.2405	8×450 double	8×450 double
6	'NC06'	67	1	169.2874	68.6386	8×450 double	8×450 double
7	'NC07'	57	1	173.4562	59.3461	8×450 double	8×450 double
8	'NC08'	65	1	161.4962	51.5521	8×450 double	8×450 double
9	'NC09'	63	1	163.6110	51.5550	8×450 double	8×450 double
10	'NC10'	63	1	167.6989	73.5794	8×450 double	8×450 double
11	'NC11'	61	0	160.4540	65.5484	8×450 double	8×450 double
12	'NC12'	60	1	162.0846	68.6749	8×450 double	8×450 double
13	'NC13'	59	0	165.8496	68.3210	8×450 double	8×450 double
14	'NC14'	50	0	164.8730	53.5264	8×450 double	8×450 double
15	'NC15'	52	1	172.6919	57.2291	8×450 double	8×450 double
16	'NC16'	53	0	163.8809	59.1720	8×450 double	8×450 double
17	'NC17'	64	1	168.6986	73.7783	8×450 double	8×450 double
18	'NC18'	62	1	164.8790	70.9246	8×450 double	8×450 double
19	'NC19'	52	0	171.6970	68.7181	8×450 double	8×450 double
20	'NC20'	56	0	164.9102	53.6330	8×450 double	8×450 double
21	'NC21'	55	1	166.5719	63.4037	8×450 double	8×450 double
22	'NC22'	59	1	171.1880	67.7463	8×450 double	8×450 double
23	'NC23'	66	0	160.9877	70.1898	8×450 double	8×450 double
24	'NC24'	55	0	170.2819	73.0139	8×450 double	8×450 double
25	'NC25'	60	1	170.8547	76.9674	8×450 double	8×450 double
26	'NC26'	62	0	166.0561	51.9179	8×450 double	[ ]
27	'NC27'	51	0	162.1981	78.4609	8×450 double	8×450 double
28	'NC28'	56	1	168.7833	53.3143	8×450 double	8×450 double
29	'NC29'	55	1	166.1138	56.4616	[ ]	8×450 double

Fields	name	age	gender	height	weight	predata	postdata
30	'NC30'	63	0	171.1190	71.4411	8×450 double	8×450 double

```
% Print statistical results for males
fprintf('Male_Avg_Age: %.2f, Std Div: %.2f\n', stats.male.age.mean,
stats.male.age.std);
```

```
Male_Avg_Age: 60.22, Std Div: 4.33
```

```
fprintf('Male_Avg_Height: %.2f, Std Div: %.2f\n', stats.male.height.mean,
stats.male.height.std );
```

```
Male_Avg_Height: 167.68, Std Div: 3.69
```

```
fprintf('Male_Avg_Weight: %.2f, Std Div: %.2f\n', stats.male.weight.mean,
stats.male.weight.std );
```

```
Male_Avg_Weight: 63.75, Std Div: 9.16
```

```
% Print statistical results for females
fprintf('Female_Avg_Age: %.2f, Std Div: %.2f\n', stats.female.age.mean,
stats.female.age.std);
```

```
Female_Avg_Age: 58.00, Std Div: 6.05
```

```
fprintf('Female_Avg_Height: %.2f, Std Div: %.2f\n', stats.female.height.mean,
stats.female.height.std );
```

```
Female_Avg_Height: 165.84, Std Div: 3.78
```

```
fprintf('Female_Avg_Weight: %.2f, Std Div: %.2f\n', stats.female.weight.mean,
stats.female.weight.std );
```

```
Female_Avg_Weight: 64.68, Std Div: 8.59
```

```
% Print gender counts
fprintf('male_count: %d\n', count_gender_1);
```

```
male_count: 18
```

```
fprintf('female_count: %d\n', count_gender_0);
```

```
female_count: 12
```

```
% Save male and female data to separate .mat files
save('Male.mat',"male_data");
save('Female.mat',"female_data");

%The lab this week is much more diffcult compare to last two labs, but
% I think that it is still within acceptable range.
```