

1ST

MATLAB - Lectures 1ST SEMESTER

LAB 1 IMPORTING AND EXPORTING DATA

Topics Covered:

1. The **load** command.
2. The **save** command.
3. The **xlsread** command.
4. The **xlwrite** command.

This is **useful** to know **how** to use **save** and **load** to transfer variables between the Workspace and a disk file.

1

1ST

IMPORTING AND EXPORTING DATA

IMPORTANCE DATA IN A MODEL

- What is a **model** ?
- The Atmospheric Science community includes a large and energetic group of researchers who devise and carry out measurements in the Atmosphere. This work involves instrument development, algorithm development, **data** collection, **data** reduction, and **data** analysis.
- The **data** by themselves are just numbers. In order to make physical sense of the **data**, some sort of **model** is needed. This might be a qualitative conceptual **model**, or it might be an analytical theory, or it might take the form of a **computer program**.

2

1ST

IMPORTING AND EXPORTING DATA INPUT/OUTPUT (I/O) DATA

- ❑ **Importing data** is the process of retrieving (Input) data from sources external an ASCII text file.
- ❑ **Exporting data** is the process of extracting (Output) data from an instance of output data into some user-specified format .
- ❑ **load** and **save** are used to **import** and **export** data.

3

1ST

IMPORTING AND EXPORTING DATA

Other import/export functions

- ❑ with differing degrees of flexibility and ease of use, include **csvread**, **csvwrite**, **dlmread**, **dlmwrite**, **fgets**, **fprintf** (which has an optional argument to specify a file), **fscanf**, **textread**, **xlsread** , **xlswrite** You know where to look for the details!
- ❑ '-mat' Keyword that indicates that the specified file is a MAT-file.
- ❑ '-ascii' Keyword that indicates that the specified file is an ASCII file.

4

IMPORTING AND EXPORTING DATA

1ST

1. The **save** command :-

If you want to store the data into an ASCII dat-file (in the **current directory**), make the filename the same as the name of the data and type **'/ascii'** at the end of the **save** statement.

Syntax

save (filename)

Save all workspace variables to (...-.mat)

save(filename, variables)

stores only the specified variables

save (filename, ... ,format)

saves in the specified format '-mat' or '-ascii'

5

1ST

Ex:- Save data to an ASCII file ,and view the contents of the file with the type function.

```
>> p = rand(1, 10);
>> q = ones(10);
>> save('pqfile.txt','p','q','-ascii')
>> save('pqfile.mat','p','q','-mat')
or
>> save pqfile.txt p q -ascii
>> save pqfile.mat p q -mat
```

➤ The extension -.mat is the default—you can specify a different extension.

6

IMPORTING AND EXPORTING DATA

1ST

2. The load command :-

The **load** command is the reverse of save. It is used to Importing text (ASCII) data or load data from MAT-file into workspace .

Syntax

S=load (filename)

S=load (filename, variables)

S=load (filename, format ,variables) load in the specified format '-mat' or '-ascii'

load (filename,___) loads without combining MAT-file variables into a structure array .

7

IMPORTING AND EXPORTING DATA

1ST

```
>> p = rand(1, 10);
```

```
>> q = ones(10);
```

```
>> save pqfile p q
```

```
>> clear all
```

```
>> p
```

Undefined function or variable 'p'.

```
>> load pqfile p q
```

```
>> p
```

```
>> q
```

8

1ST

IMPORTING AND EXPORTING DATA

EXAMPLE 1: THE PRECIPITATION

This example show you how to load a simple data set and plot it.

The PDXprecip.dat file contains two columns of numbers. The first is the number of the month, and the second is the mean precipitation recorded at the Portland International Airport between 1961 and 1990. Here are the MATLAB commands to create a symbol plot with the data from PDXprecip.dat.

9

1ST

IMPORTING AND EXPORTING DATA

EXAMPLE 1: THE PRECIPITATION

PDXprecip.dat

1	5.35
2	3.68
3	3.54
4	2.39
5	2.06
6	1.48
7	0.63
8	1.09
9	1.75
10	2.66
11	5.34
12	6.13

✓ Documents → MATLAB → R.Click :New → Text Document
write the table above and save it as PDXprecip.dat

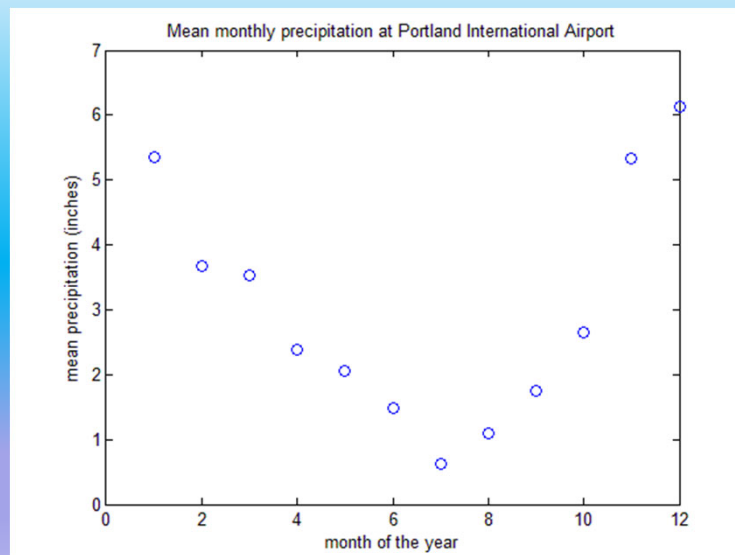
10

script-file [precipPlot.m](#) :

```
% Filename: precipPlot.m
% Load data from PDXprecip.dat and plot it with symbols
% read data into PDXprecip matrix
clc,clear all
load PDXprecip.dat;
% copy first column of PDXprecip into month and second
%column into precip
month = PDXprecip(:,1);
precip = PDXprecip(:,2);
plot(month , precip,'o');    % plot precip vs. month with circles
xlabel('month of the year'); % add axis labels and plot title
ylabel('mean precipitation (inches)');
title('Mean monthly precipitation at Portland International Airport');
```

1ST

IMPORTING AND EXPORTING DATA



1ST

12

1ST

IMPORTING AND EXPORTING DATA

3. The xlsread command :- Read Microsoft Excel spreadsheet file (.xlsx),retrieving or Importing data .

Syntax

```
num = xlsread(filename)
num = xlsread(filename , sheet)
num = xlsread(filename , xlRange)
num = xlsread(filename , sheet , xlRange)
num = xlsread(filename , sheet, xlRange ,'basic')
[num , txt , raw] = xlsread(____)
____ = xlsread(filename,-1)
```

13

1ST

IMPORTING AND EXPORTING DATA

num= xlsread(filename)

- Read and returns numeric data in **double** array **num** from the first sheet in the Microsoft Excel spreadsheet file named **filename**.

num= xlsread(filename,-1)

- Opens the file **filename** in an Excel window, enabling you to interactively select the worksheet to be read and the range of data on that worksheet to import.

num= xlsread(filename, sheet)

- Reads the specified worksheet, where **sheet** is either a positive, double scalar value or a quoted string containing the sheet name.

14

IMPORTING AND EXPORTING DATA

1ST

num= **xlsread**(filename, xlRange)

- Reads data from a specific rectangular region of the default worksheet (**Sheet1**). Specify **range** using the syntax '**C1:C2**', where **C1** and **C2** are two opposing corners that define the region to be read.

num=**xlsread**(filename , sheet, xlRange)

- Reads data from a specific rectangular region (**range**) of the worksheet specified by **sheet**.

num= **xlsread**(filename, sheet , xlRange, 'basic')

- Imports data from the spreadsheet in basic import mode. This is the mode used on UNIX platforms as well as on Windows when Excel is not available as a COM **server**.

15

IMPORTING AND EXPORTING DATA

1ST

[num,text,row]= **xlsread**(filename, ...)

- Returns numeric and text data in **num** and **txt**, and unprocessed cell content in cell array **raw**, which contains both numeric and text data.

Example :- Read Data from First Worksheet into Numeric Array .

- Create an Excel file named myExample.xlsx

```
>> values = {1, 2, 3 ; 4, 5, 'x' ; 7, 8, 9};
% to define a subset of the array. Enclose indices in curly
brackets, { }, to refer to the text, numbers, or other data
within individual cells.
>> headers = {'First', 'Second', 'Third'};
>> xlswrite('myExample.xlsx', [headers; values]);
```

16

IMPORTING AND EXPORTING DATA

1ST

- Read data from the first worksheet

```
>> filename = 'myExample.xlsx';
>> A = xlsread(filename)
or
>> A = xlsread('myExample.xlsx')
A =
     1     2     3
     4     5 NaN
     7     8     9
>> [num text raw]=xlsread('myExample.xlsx')
```

- ❖ xlsread returns the numeric data in array A .

17

1ST

```
>> [num text raw]=xlsread('myExample.xlsx')
num =
     1     2     3
     4     5 NaN
     7     8     9
text =
'First' 'Second' 'Third'
"      "      "
"      "      'x'
raw =
'First' 'Second' 'Third'
 [ 1] [ 2] [ 3]
 [ 4] [ 5] 'x'
 [ 7] [ 8] [ 9]
```

18

1ST

IMPORTING AND EXPORTING DATA

EXAMPLE 2: READ A SPECIFIC RANGE OF DATA FROM THE EXCEL FILE IN THE PREVIOUS EXAMPLE.

```

>> filename = 'myExample.xlsx';
>> sheet = 1;
>> xlRange = 'B2:C3';
>> subsetA = xlsread(filename, sheet, xlRange)
subsetA =
     2     3
     5    NaN
    
```

19

1ST

IMPORTING AND EXPORTING DATA

EXAMPLE 3: Import data of the outdoor temperature, Sunlight radiation, pressure and the relative humidity from nivada.xlsx
 Do four plots of outdoor temperature, sunlight radiation, pressure, and relative humidity versus time (hours) for Nivada dataset 2007 for four months (Jan. to Apr.)

1	Year	Day of Year	Month	Day	Hour	Solar Altitude	Solar Energy(W/m ²)	Air Temp(C)	Wind Speed(m/s)	Wind Dir	Pressure(h pa)	RH %
2	2007	1	1	1	0	-73	0	-6.9	0.9	193	907	72
3	2007	1	1	1	1	-64	0	-7	1	147	907	73
4	2007	1	1	1	2	-52	0	-8.4	0.9	188	908	77
5	2007	1	1	1	3	-40	0	-7.8	1	194	908	78
6	2007	1	1	1	4	-29	0	-9.4	0.9	174	908	81
7	2007	1	1	1	5	-17	0	-8.9	1.1	193	909	82
8	2007	1	1	1	6	-6	0	-7.5	1.3	180	909	78
9	2007	1	1	1	7	5	58	-6.5	1	162	910	78
10	2007	1	1	1	8	14	228	2.5	1.8	336	910	51
11	2007	1	1	1	9	22	373	7.7	2.3	55	910	38
12	2007	1	1	1	10	27	476	10.4	4.3	40	910	32
13	2007	1	1	1	11	30	524	11.8	6.2	30	909	29
14	2007	1	1	1	12	29	511	12.5	6.2	49	908	28
15	2007	1	1	1	13	25	439	12.9	5.8	10	908	27
16	2007	1	1	1	14	19	317	12.9	5.5	342	908	27
17	2007	1	1	1	15	10	163	12	5	288	908	28
18	2007	1	1	1	16	0	21	9.3	3.7	34	908	33
19	2007	1	1	1	17	-10	0	5.9	1.6	110	909	39
20	2007	1	1	1	18	-22	0	6	2.8	30	910	41
21	2007	1	1	1	19	-34	0	3.8	1.1	166	911	45
22	2007	1	1	1	20	-46	0	4.3	3.1	21	911	46
23	2007	1	1	1	21	-57	0	4.8	2.5	24	912	44
24	2007	1	1	1	22	-68	0	5.1	3.3	30	912	43
25	2007	1	1	1	23	-75	0	3.7	3.2	33	912	48

20

script-file myNivada.m :

```
% Filename: myNivada.m
% by Thaer O. Roomi (2014)
% This program plots the values of temperature, sunlight radiation,
% Average pressure, and Relative Humidity for hourly averages
% versus the time (hours) for four months (1,2,3 and 4) 2007.
clear all
[num txt]=xlsread('nivada.xlsx');
year=num(:,1) ; day_of_year=num(:,2) ; hour=num(:,5);
DateNumber = datenum(year,month,day,hour,0,0);
temperature=num(:,8);
sunlight=num(:,7);
pressure=num(:,11);
relhum=num(:,12);
```

```
% divided figure into 4-plots
subplot(2,2,1)
plot(DateNumber, temperature)
xlabel('Time [hours]')
ylabel('Temperature [degrees C]')
title('Outdoor temperature in Nivada 2007')
%datetick('x','mmm','kepticks')
%datetick('x','mmm-dd (ddd)','kepticks')
%datetick('x','HHPM','kepticks')
datetick('x','yyyy-mmm','kepticks')
```

22

1ST

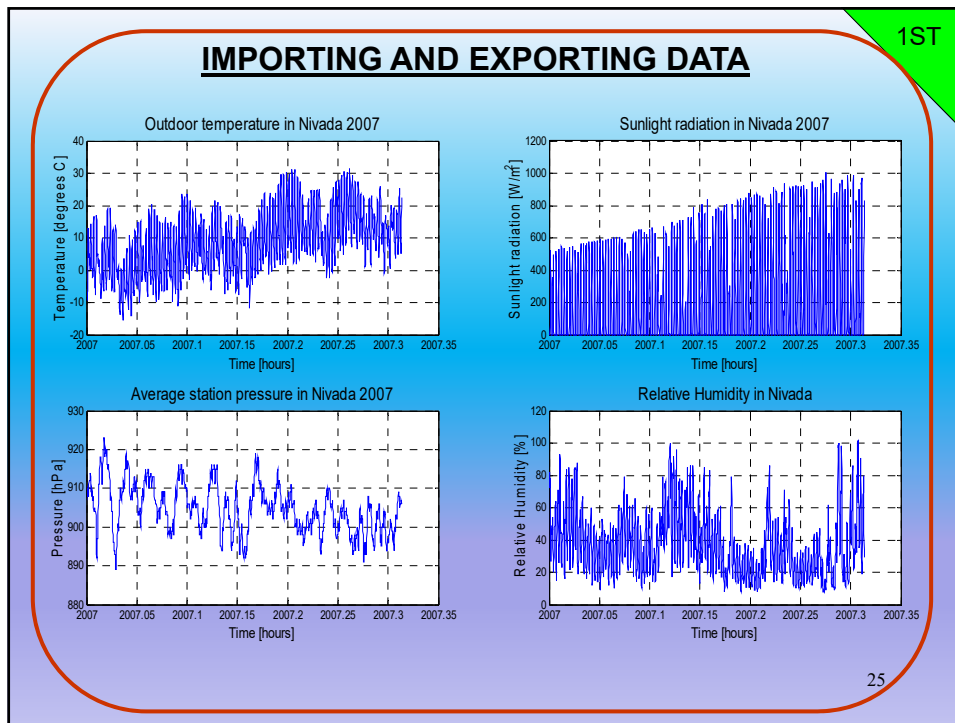
```
subplot(2,2,2)
plot(DateNumber,sunlight)
xlabel('Time [hours]')
ylabel('Sunlight radiation [W/m^2]')
title('Sunlight radiation in Nivada 2007')
datetick('x','yyyy-mmm','kepticks')
subplot(2,2,3)
plot(DateNumber,pressure)
xlabel('Time [hours]')
ylabel('Pressure [hPa]')
title('Average station pressure in Nivada 2007')
datetick('x','yyyy-mmm','kepticks')
```

23

1ST

```
grid on
subplot(2,2,4)
plot(DateNumber,relhum)
xlabel('Time [hours]')
ylabel('Relative Humidity [%]')
title('Relative Humidity in Nivada')
datetick('x','yyyy-mmm','kepticks')
```

24



IMPORTING AND EXPORTING DATA

1ST

4. The xlswrite command :- Write Microsoft Excel spreadsheet file (.xlsx) , extracting or Exporting data .

Syntax

`xlswrite(filename ,A)`
`xlswrite(filename ,A, sheet)`
`xlswrite(filename ,A, xlRange)`
`xlswrite(filename ,A, sheet , xlRange)`

26

IMPORTING AND EXPORTING DATA

1ST

xlswrite(filename ,A)

- Writes array **A** to the first worksheet in Excel file, **filename** , starting at cell **A1**.

xlswrite(filename ,A, sheet)

- Writes to the specified worksheet.

xlswrite(filename ,A, xlRange)

- Writes to the rectangular region specified by **xlRange** in the first worksheet of the file.

xlswrite(filename ,A, sheet, xlRange)

- Writes to the specified **sheet** and range, **xlRange**.

27

IMPORTING AND EXPORTING DATA

1ST

Example :- Write Data to a Spreadsheet First. Write a 7-element vector to an Excel file, testdata.xlsx.

```
>> filename = 'testdata.xlsx';
>> A = [12.7, 5.02, -98, 63.9, 0, -.2, 56];
>> xlswrite(filename , A)
```

Example :- Write Data to a Specific Sheet and Range in a Spreadsheet. Write mixed text and numeric data to an Excel file, testdata.xlsx, Starting at cell E1 of Sheet2.

```
>> filename = 'testdata.xlsx';
>> A = {'Time', 'Temperature'; 12,98; 13,99; 14,97};
>> sheet = 2;
>> xlRange = 'E1';
>> xlswrite(filename, A, sheet , xlRange)
```

8

1ST

IMPORTING AND EXPORTING DATA

Homework

Write a MATLAB program to read the numerical and analytical solutions in table (1) from excel sheet file to calculate the mean of the numerical values, "Quarter, Hourly, Simidiurnal, Diurnal"

Note: use an Excel spreadsheet to reading and writing values. Then draw each of numerical values versus the time steps.

20

1ST

IMPORTING AND EXPORTING DATA

	A	B	C	D	E
1	LONGWAVE SOLAR RADIATION				
2		Time (hour)	Wm ⁻²		
3	1	10/21/2014 12:00:0	366.946		
4	1	10/21/2014 12:15:0	375.059		
5	1	10/21/2014 12:30:0	377.508		
6	1	10/21/2014 12:45:0	379.093		
7	1	10/21/2014 1:00:0	371.259		
8	1	10/21/2014 1:15:0	371.807		
9	1	10/21/2014 1:30:0	360.323		
10	1	10/21/2014 1:45:0	334.417		
11	1	10/21/2014 2:00:0	335.704		
12	1	10/21/2014 2:15:0	352.121		
13	1	10/21/2014 2:30:0	340.177		
14	1	10/21/2014 2:45:0	343.073		
15	1	10/21/2014 3:00:0	347.619		
16	1	10/21/2014 3:15:0	342.998		
17	1	10/21/2014 3:30:0	341.575		
18	1	10/21/2014 3:45:0	337.019		

30

IMPORTING AND EXPORTING DATA

1ST

```

clear all , clc
[A date]=xlsread('long_w.xlsx');
C=A(:,3);           % Read data
h=0;               % new index for the new values
n=numel(C);       % number of elements in the excel
sheet
for i=1:4:n-3      % read i =1,5,9,...n-3
    h=h+1; % to add 1 to index h=1,2,3,4,...
    C_mean_4(h)=mean(C([i:i+3])); % find mean value for
each 4 elements of the array
end
if mod(n,4)~=0;
C_mean_4(h+1)= mean(C(n-mod(n,4):end));
end

```

IMPORTING AND EXPORTING DATA

1ST

```

h=numel(C_mean_4)
C_M_4=C_mean_4';
k=0;
for i=1:12:h-11
    k=k+1;
    C_mean_12(k)=mean(C_M_4([i:i+11]));
end
if mod(h,12)~=0;
C_mean_12(k+1)= mean(C_M_4(n-mod(n,12):end));
end
k=numel(C_mean_12);
C_M_12=C_mean_12';

```


IMPORTING AND EXPORTING DATA

1ST

```
kk=0;
for i=1:24:h-23
    kk=kk+1;
    C_mean_24(kk)=mean(C_M_4([i:i+23]));
end
if mod(h,24)~=0;
C_mean_24(kk+1)= mean(C_M_4(n-mod(n,24):end));
end
kk=numel(C_mean_24);
C_M_24=C_mean_24';
```

IMPORTING AND EXPORTING DATA

1ST

```
filename='long_w.xlsx';
sheet = 2;
DD={'Time in
Year','Time','Quarter','Time','HOURLY','Time','SIMIDIURNAL','Ti
me','DIURNAL'};
date1=date([3:end],2);
t1=date1(1:4:end);
t2=date1(1:4*12:end);
t3=date1(1:4*24:end);sheet = 1;
```

IMPORTING AND EXPORTING DATA

1ST

```

xlswrite(filename,DD,sheet,'A1')
xlswrite(filename,A(:,1),sheet,'A2')
xlswrite(filename,date1,sheet,'B2')
xlswrite(filename,C,sheet,'C2')
xlswrite(filename,t1,sheet,'D2')
xlswrite(filename,C_M_4,sheet,'E2')
xlswrite(filename,t2,sheet,'F2')
xlswrite(filename,C_M_12,sheet,'G2')
xlswrite(filename,t3,sheet,'H2')
xlswrite(filename,C_M_24,sheet,'I2')

```

IMPORTING AND EXPORTING DATA

1ST

```

d=datenum(date1);
% datenum :Convert date and time to serial date number
figure(1)
hold on
plot(d,C,':B')
plot(d(1:4:end),C_M_4,':Y')
plot(d(1:12*4:end),C_M_12,':G')
plot(d(1:24*4:end),C_M_24,'R')
datetick('x','mmm','kepticks')
xlabel('Time(hr)')
ylabel('Long Wave Solar Radiation (W/m^2)')
title('Long Wave Solar Radiation')
legend('QUARTER','HOURLY','SIMIDIURNAL','DIURNAL')

```

1ST

IMPORTING AND EXPORTING DATA

Homework2

Write a MATLAB program to read data for Aerosol Optical Depth from Excel spreadsheet file AOD.xlsx to calculate the mean of monthly, seasonal and yearly .

37

1ST

IMPORTING AND EXPORTING DATA

	A	B	C	D	E
1	time	AOD			
2	2/24/2000	-9999			
3	2/25/2000	-9999			
4	2/26/2000	0.319			
5	2/27/2000	-9999			
6	2/28/2000	0.154			
7	2/29/2000	-9999			
8	3/1/2000	-9999			
9	3/2/2000	0.672			
10	3/3/2000	0.13			
11	3/4/2000	0.648			
12	3/5/2000	-9999			
13	3/6/2000	0.414			
14	3/7/2000	0.465			

38

IMPORTING AND EXPORTING DATA

1ST

```

% This is a program to calculate mean of monthly, seasonal
and yearly data.
clear all,clc
[num txt]=xlsread('AOD.xlsx');
date=txt(2:end,1);
formatIn = 'mm/dd/yyyy';
t=datenum(date,formatIn);
c1=find(num==-9999);
t(c1)=[];num(c1)=[];
y=year(t);m=month(t);d=day(t);
AOD=num;
%%%%%%%%%% MEAN OF MONTHLY DATA %%%%%%%%%%%
h=0; % new index for the new values

```

IMPORTING AND EXPORTING DATA

1ST

```

for yy=2000:2016 % for yy=y(1):y(end)
    h=h+1;
    for mo=1:12
        in=find(y==yy);
        in1=find(m(in)==mo);
        monthly(h,mo)=nanmean(AOD(in1));
    end
end
filename='AOD.xlsx'; sheet=2;
dd={'Jan','Feb','Mar','Apr','May','Jun','Jul','Aug','Sep','Oct','Nov','Dec'};
xlswrite(filename,dd,sheet,'b1')
xlswrite(filename,{'YEAR'},sheet,'A1')
xlswrite(filename,[2000:2016],sheet,'a2')
xlswrite(filename,monthly,sheet,'B2')

```

IMPORTING AND EXPORTING DATA

1ST

```

%%%%%%%%%% MEAN OF SEASONAL DATA %%%%%%%%%%%
winter=nanmean(monthly(:,[12,1,2]));
winter=winter';
spring=nanmean(monthly(:,[3,4,5]));
spring=spring';
summer=nanmean(monthly(:,[6,7,8]));
summer=summer';
autumn=nanmean(monthly(:,[9,10,11]));
autumn=autumn';

```

IMPORTING AND EXPORTING DATA

1ST

```

sheet=3;
ddd={'YEAR','Winter','Spring','Summer','Autumn'};
aa=[winter,spring,summer,autumn];
xlswrite(filename,ddd,sheet,'A1')
xlswrite(filename,[2000:2016]',sheet,'a2')
xlswrite(filename,aa,sheet,'b2')
%%%%%%%%%% MEAN OF YEARLY DATA %%%%%%%%%%%
h=0;
for yy=2000:2016
    h=h+1;
    in=find(y==yy);
    yearly_mean(h)=nanmean(AOD(in));
end

```

1ST

IMPORTING AND EXPORTING DATA

```
sheet=4;
xlswrite(filename,{'YEAR'},sheet,'A1')
xlswrite(filename,[2000:2016],sheet,'a2')
xlswrite(filename,{'AOD'},sheet,'B1')
xlswrite(filename,yearly_mean',sheet,'B2')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```