

---

```
function varargout = untitled2(varargin)

gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',   gui_Singleton, ...
                  'gui_OpeningFcn',  @untitled2_OpeningFcn, ...
                  'gui_OutputFcn',   @untitled2_OutputFcn, ...
                  'gui_LayoutFcn',   [], ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function untitled2_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;
set(handles.axes6,'xTick',[]);
set(handles.axes6,'ytick',[]);
set(handles.axes6,'box','on');

guidata(hObject, handles);

function varargout = untitled2_OutputFcn(hObject, eventdata, handles)

varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)

[FileName PatName]=uigetfile({'*.xlsx';*.xls'},'Choose a File');
str=[PatName FileName];
set(handles.edit1,'string',str);
hwaitbar=waitbar(0,'请等待');
[a,b]=xlsread(str,1,'A1:AB9000');    %读取 EXCEL 内容，数据和字符分别记录在 a 和 b 内
waitbar(0.2,hwaitbar)
for i=2:length(b(:,1))                %该循环作用是将字符和数据合并
    for j=6:length(b(1,:))
        if isempty(b{i,j})
            if isnan(a(i-1,j-5))
```

---

```

        b{i,j}="";
    else
        b{i,j}=num2str(a(i-1,j-5));
    end
end
end
end
waitbar(0.6,hwaitbar)
set(handles.table1,'columnname',b(1,[1:end]));    %设置表格名称
set(handles.table1,'data',b(2:end,:));          %表格赋值
set(handles.listbox1,'string',b(1,[1:end]));      %列表赋值
waitbar(0.8,hwaitbar)
plot(handles.axes4,a(:,20),a(:,22),'.');        %画图
xlabel(handles.axes4,'平均辐射温度/°C');
ylabel(handles.axes4,'平均温度/°C');
waitbar(1,hwaitbar)
delete(hwaitbar);
set(handles.listbox1,'string',b(2:end,1));
handles.b=b;
handles.xls = str;
guidata(hObject,handles);

function edit1_Callback(hObject, eventdata, handles)

function edit1_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function listbox1_Callback(hObject, eventdata, handles)

b=handles.b;
set(handles.listbox2,'string',b(1,2:end));
i=get(handles.listbox1,'value');
j=get(handles.listbox2,'value');
set(handles.listbox7,'string',b{i+1,j+1});

function listbox1_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```
function edit2_Callback(hObject, eventdata, handles)
```

```
function edit2_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function listbox2_Callback(hObject, eventdata, handles)
```

```
b=handles.b;  
i=get(handles.listbox1,'value');  
j=get(handles.listbox2,'value');  
set(handles.listbox7,'string',b{i+1,j+1});
```

```
function listbox2_CreateFcn(hObject, eventdata, handles)
```

```
function listbox3_Callback(hObject, eventdata, handles)
```

```
function listbox3_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function pushbutton3_Callback(hObject, eventdata, handles)
```

```
clc  
clear all  
close(gcf);
```

```
function popupmenu1_Callback(hObject, eventdata, handles)
```

```
data=xlsread('数据');  
x=data(:,1);  
y=data(:,4);  
plot(x,y) ;  
axes(handles.axes2)
```

```
function popupmenu1_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pushbutton4_Callback(hObject, eventdata, handles)

function pushbutton5_Callback(hObject, eventdata, handles)

function axes1_CreateFcn(hObject, eventdata, handles)

ha=axes('units','normalized','pos',[0 0 1 1]);
uistack(ha,'down');
ii=imread('background.jpg');
image(ii);
colormap gray
set(ha,'handlevisibility','off','visible','on');

% --- Executes on key release with focus on axes1 and none of its controls.
function axes1_KeyReleaseFcn(hObject, eventdata, handles)

function edit3_Callback(hObject, eventdata, handles)

function edit3_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit4_Callback(hObject, eventdata, handles)

function edit4_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```

```
function edit5_Callback(hObject, eventdata, handles)
```

```
function edit5_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function pushbutton7_Callback(hObject, eventdata, handles)
```

```
function pushbutton8_Callback(hObject, eventdata, handles)
```

```
function popupmenu2_Callback(hObject, eventdata, handles)
```

```
function popupmenu2_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function popupmenu3_Callback(hObject, eventdata, handles)
```

```
function popupmenu3_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function listbox7_Callback(hObject, eventdata, handles)
```

```
function listbox7_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function edit7_Callback(hObject, eventdata, handles)
```

```
function edit7_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))  
    set(hObject,'BackgroundColor','white');  
end
```

```
function pushbutton10_Callback(hObject, eventdata, handles)

xlswrite('导出结果.xlsx',handles.b,1,'A1');

function table1_KeyPressFcn(hObject, eventdata, handles)

function table1_DeleteFcn(hObject, eventdata, handles)

function table1_CreateFcn(hObject, eventdata, handles)

function text3_CreateFcn(hObject, eventdata, handles)

function table1_CellEditCallback(hObject, eventdata, handles)

function axes2_CreateFcn(hObject, eventdata, handles)

function axes2_DeleteFcn(hObject, eventdata, handles)

function edit8_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit8_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit9_Callback(hObject, eventdata, handles)

function edit9_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function pushbutton11_Callback(hObject, eventdata, handles)

ab=str2num(get(handles.edit8,'string'));
cd=str2num(get(handles.edit9,'string'));

databasedatabase=xlsread(handles.xls);
```

---

```

database(:,1)=databasedatabase(:,25); % 25-->30
database(:,2)=databasedatabase(:,24);% 24-->29
database(:,3)=databasedatabase(:,6);% 6-->11
Z=size(database,1);
n=0;
for j=1:1:Z
    testdata = database(j,1:3);
    database1 = database(:,1:3);
    database1(j,:)=[];
    temp1 = repmat(testdata(:,2), [size(database1,1), 1]); %得到计算矩阵
    dist = (temp1 - database1(:,2)).^2; %得到其距离 (矩阵)
    dist0 = sqrt(dist); %计算距离 (数值)
    [sorted, sortedIndex] = sort(dist0); %对距离排序
    q=sum(sum(sorted<ab));
    for i=1:1:q
        traindata(i,1)=database1(sortedIndex(i),3); %k 个近邻点, 求训练集
    end
    mu=mean(traindata(1:q));
    Sigma=std(traindata(1:q));
    pp=normpdf(testdata(:,3),mu,Sigma);%求取联合概率密度;
    if pp<cd
        n=n+1;
        result(n,1)=testdata(:,1);
    end
end
a=0;
for m=1:1:Z
    temp2 = repmat(database(m,1), [size(result,1), 1]);
    Q=intersect(temp2,result(:,1));
    if isempty(Q)==1
        a=a+1;
        bestdata(a,1:2)=database(m,2:3);
    end
end
X=bestdata(:,1);
Y=bestdata(:,2);
Step=0.3;
Edges=floor(min(X)/Step)*Step:Step:ceil(max(X)/Step)*Step;%规定区间
[Num,Edges,Bin] = histcounts(X,Edges);%制定边界找出频次, Num 为落在区间的个数, Bin 为每个值落在哪个区间段
B=zeros(numel(Edges),2);%numel(Edges)统计个数
for NN=1:numel(Edges)
    B(NN,1)=mean(X(Bin==NN));%计算落在某个区间数的平均值, Bin 不仅统计了每个数落在哪个区间, 而且映射了实际值

```

```
B(NN,2)=mean(Y(Bin==NN));%MeanY
end
A=rmmissing(B);
P = polyfit(A(:,1),A(:,2), 1);
yfit = P(1)*A(:,1) + P(2);
R2 = norm(yfit - mean(A(:,2)))^2/norm(A(:,2) - mean(A(:,2)))^2;%拟合 R2
set(handles.listbox8,'string',result(n,1))

function listbox8_Callback(hObject, eventdata, handles)

function listbox8_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit10_Callback(hObject, eventdata, handles)

function edit10_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit11_Callback(hObject, eventdata, handles)

function edit11_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function axes6_CreateFcn(hObject, eventdata, handles)

function axes6_ButtonDownFcn(hObject, eventdata, handles)

function axes6_DeleteFcn(hObject, eventdata, handles)

function pushbutton12_Callback(hObject, eventdata, handles)
```



---

```

database=xlread(handles.xls);
database(:,1)=database(:,25); % 25-->30
database(:,2)=database(:,24); % 24-->29
database(:,3)=database(:,6); % 6-->11
Z=size(database,1);
ef=0.3;
r=0;
for d=0.1:0.2:2.5
    r=r+1;
    s=0;
    for p=0.005:0.005:0.1
        s=s+1;
        clearvars -except database Z r s d p eventdata handles;
        n=0;
        for j=1:1:Z
            testdata = database(j,1:3);
            database1 = database(:,1:3);
            database1(j,:)=[];
            temp1 = repmat(testdata(:,2), [size(database1,1), 1]); %得到计算矩阵
            dist = (temp1 - database1(:,2)).^2; %得到其距离 (矩阵)
            dist0 = sqrt(dist); %计算距离 (数值)
            [sorted, sortedIndex] = sort(dist0); %对距离排序
            q=sum(sum(sorted<d));
            for i=1:1:q
                traindata(i,1)=database1(sortedIndex(i),3); %k 个近邻点, 求训练集
            end
            mu=mean(traindata(1:q));
            Sigma=std(traindata(1:q));
            pp=normpdf(testdata(:,3),mu,Sigma);%求取联合概率密度;
            if pp<p
                n=n+1;
                result(n,1)=testdata(:,1);
            end
        end
        a=0;
        for m=1:1:Z
            temp2 = repmat(database(m,1), [size(result,1), 1]);
            Q=intersect(temp2,result(:,1));
            if isempty(Q)==1
                a=a+1;
                bestdata(a,1:2)=database(m,2:3);
            end
        end
        X=bestdata(:,1);

```

---

```

    Y=bestdata(:,2);
    Step=ef;
    Edges=floor(min(X)/Step)*Step:Step:ceil(max(X)/Step)*Step;%规定区间
    [Num,Edges,Bin]=histcounts(X,Edges);%制定边界找出频次, Num 为落在区间的个数, Bin 为每个
    值落在哪个区间段
    B=zeros(numel(Edges),2);%numel(Edges)统计个数
    for NN=1:numel(Edges)
        B(NN,1)=mean(X(Bin==NN));%计算落在某个区间数的平均值, Bin 不仅统计了每个数落在哪个
        区间, 而且映射了实际值
        B(NN,2)=mean(Y(Bin==NN));%MeanY
    end
    A=rmmissing(B);
    P = polyfit(A(:,1),A(:,2), 1);
    yfit = P(1)*A(:,1) + P(2);
    R2 = norm(yfit -mean(A(:,2)))^2/norm(A(:,2) - mean(A(:,2)))^2;%拟合 R2
    getdata1(s,r)=R2;
end
end
X=database(:,2);
Y=database(:,3);
Step=ef;
Edges=floor(min(X)/Step)*Step:Step:ceil(max(X)/Step)*Step;%规定区间
[Num,Edges,Bin]=histcounts(X,Edges);%制定边界找出频次, Num 为落在区间的个数, Bin 为每个值落在哪个
区间段
B=zeros(numel(Edges),2);%numel(Edges)统计个数
for NN=1:numel(Edges)
    B(NN,1)=mean(X(Bin==NN));%计算落在某个区间数的平均值, Bin 不仅统计了每个数落在哪个区间,
    而且映射了实际值
    B(NN,2)=mean(Y(Bin==NN));%MeanY
end
A=rmmissing(B);
P = polyfit(A(:,1),A(:,2), 1);
yfit = P(1)*A(:,1) + P(2);
R2 = norm(yfit -mean(A(:,2)))^2/norm(A(:,2) - mean(A(:,2)))^2;%拟合 R2
getdata2= repmat(R2,1,13);
getdata=cat(1,getdata2,getdata1);
m=0;
k=0;
for r=0.1:0.2:2.5
    m=m+1;
    n=0;
    for s=0:0.005:0.1
        k=k+1;
        n=n+1;
    end
end

```

```
        pictu(k,3)=getdata(n,m);
        pictu(k,1)=s;
        pictu(k,2)=r;
    end
end
axes(handles.axes6);
x=pictu(:,1);y=pictu(:,2);z=pictu(:,3);
[X,Y,Z]=griddata(x,y,z,linspace(min(x),max(x),20)',linspace(min(y),max(y),13),'v4');%插值
contourf(X,Y,Z,10,'linestyle','none');
colormap jet
colorbar;

function axes4_CreateFcn(hObject, eventdata, handles)

function edit13_Callback(hObject, eventdata, handles)

function edit13_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

function edit14_Callback(hObject, eventdata, handles)

function edit14_CreateFcn(hObject, eventdata, handles)

if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
```