Seznam příloh

1. Skript do MATLAB – Porovnání RRN a CDR s metodou Kriging
2. Skript do MATLAB – RRN GUI
3. Skript do MATLAB – TS GUI
4. Databáze Time Series (1992 – 2015) na microSD kartě
5. Elektronická verze diplomové práce na CD

1. Skript do MATLAB – Porovnání RRN a CDR s metodou Kriging (příklad FMI)

```
clc
clear

%READ IMAGES without correction
%Landsat 4
a=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index1Aa.tif');
b=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index2Aa.tif');
%Landsat 5
c=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index3Aa.tif');
d=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index4Aa.tif');
e=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index5Aa.tif');
f=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index6Aa.tif');
g=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index7Aa.tif');
h=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index8Aa.tif');
i=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index9Aa.tif');
j=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index10Aa.tif');
%Landsat 7
k=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index11Aa.tif');
l=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index12Aa.tif');
%Landsat 8
m=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index13Aa.tif');
n=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index14Aa.tif');
o=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index15Aa.tif');
p=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index16Aa.tif');
q=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIA\index17Aa.tif');

%READ IMAGES with RIDGE RRN
%Landsat 4
a2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index1Ya.tif');
b2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index2Ya.tif');
%Landsat 5
c2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index3Ya.tif');
d2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index4Ya.tif');
e2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index5Ya.tif');
f2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index6Ya.tif');
g2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index7Ya.tif');
h2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index8Ya.tif');
i2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index9Ya.tif');
j2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index10Ya.tif');
%Landsat 7
k2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index11Ya.tif');
l2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index12Ya.tif');
%Landsat 8
m2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index13Ya.tif');
n2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index14Ya.tif');
o2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index15Ya.tif');
p2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index16Ya.tif');
q2=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIY\index17Ya.tif');

%READ IMAGES with PIF RRN
%Landsat 4
a3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index1Xa.tif');
b3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index2Xa.tif');
%Landsat 5
c3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index3Xa.tif');
d3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index4Xa.tif');
e3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index5Xa.tif');
f3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index6Xa.tif');
g3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index7Xa.tif');
h3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index8Xa.tif');
i3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index9Xa.tif');
j3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index10Xa.tif');
%Landsat 7
k3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index11Xa.tif');
```

```
l3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index12Xa.tif');
%Landsat 8
m3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index13Xa.tif');
n3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index14Xa.tif');
o3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index15Xa.tif');
p3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index16Xa.tif');
q3=imread('Vegetation_Indices_CDR_RRNCDR\FMI\FMIX\index17Xa.tif');

%CHOOSE PIXELS
%Map
%index0=imread('Data_GUI\index0.tif');
%Sattelite Image
%index0=imread('Data_GUI\index0b.tif');
%[xi,yi,Y]=impixel(index0);
%[xi,yi,Y]=impixel(mat2gray(index0))
%Location 1
xi=2519;
yi=1240;
%Location 2
%xi = 3099;
%yi = 909;
%location 3
%xi = 2211;
%yi = 1184;
%Location 4
%xi=2066;
%yi=1039;
%impixelinfo
%impixelregion
%imshow(index0)
%colorbar
%axis on
%grid on
%hold on
%plot(xi,yi,'ro')
%title('PIXEL LOCATION FOR TS')
figure
Y1=[impixel(a,xi,yi) impixel(b,xi,yi)];
Y1b=[impixel(a2,xi,yi) impixel(b2,xi,yi)];
Y1c=[impixel(a3,xi,yi) impixel(b3,xi,yi)];
Y1=[Y1(1,1,1) Y1(1,4,1)];
Y1b=[Y1b(1,1,1) Y1b(1,4,1)];
Y1c=[Y1c(1,1,1) Y1c(1,4,1)];
%Landsat 5
Y2=[impixel(c,xi,yi) impixel(d,xi,yi) impixel(e,xi,yi) impixel(f,xi,yi) impixel(g,xi,yi) impixel(h,xi,yi) impixel(i,xi,yi)
impixel(j,xi,yi)];
Y2b=[impixel(c2,xi,yi) impixel(d2,xi,yi) impixel(e2,xi,yi) impixel(f2,xi,yi) impixel(g2,xi,yi) impixel(h2,xi,yi)
impixel(i2,xi,yi) impixel(j2,xi,yi)];
Y2c=[impixel(c3,xi,yi) impixel(d3,xi,yi) impixel(e3,xi,yi) impixel(f3,xi,yi) impixel(g3,xi,yi) impixel(h3,xi,yi)
impixel(i3,xi,yi) impixel(j3,xi,yi)];
Y2=[Y2(1,1,1) Y2(1,4,1) Y2(1,7,1) Y2(1,10,1) Y2(1,13,1) Y2(1,16,1) Y2(1,19,1) Y2(1,22,1)];
Y2b=[Y2b(1,1,1) Y2b(1,4,1) Y2b(1,7,1) Y2b(1,10,1) Y2b(1,13,1) Y2b(1,16,1) Y2b(1,19,1) Y2b(1,22,1)];
Y2c=[Y2c(1,1,1) Y2c(1,4,1) Y2c(1,7,1) Y2c(1,10,1) Y2c(1,13,1) Y2c(1,16,1) Y2c(1,19,1) Y2c(1,22,1)];
%Landsat 7
Y3=[impixel(k,xi,yi) impixel(l,xi,yi)];
Y3b=[impixel(k2,xi,yi) impixel(l2,xi,yi)];
Y3c=[impixel(k3,xi,yi) impixel(l3,xi,yi)];
Y3=[Y3(1,1,1) Y3(1,4,1)];
Y3b=[Y3b(1,1,1) Y3b(1,4,1)];
Y3c=[Y3c(1,1,1) Y3c(1,4,1)];
%Landsat 8
Y4=[impixel(m,xi,yi) impixel(n,xi,yi) impixel(o,xi,yi) impixel(p,xi,yi) impixel(q,xi,yi)];
Y4b=[impixel(m2,xi,yi) impixel(n2,xi,yi) impixel(o2,xi,yi) impixel(p2,xi,yi) impixel(q2,xi,yi)];
Y4c=[impixel(m3,xi,yi) impixel(n3,xi,yi) impixel(o3,xi,yi) impixel(p3,xi,yi) impixel(q3,xi,yi)];
Y4=[Y4(1,1,1) Y4(1,4,1) Y4(1,7,1) Y4(1,10,1) Y4(1,13,1)];
Y4b=[Y4b(1,1,1) Y4b(1,4,1) Y4b(1,7,1) Y4b(1,10,1) Y4b(1,13,1)];
Y4c=[Y4c(1,1,1) Y4c(1,4,1) Y4c(1,7,1) Y4c(1,10,1) Y4c(1,13,1)];
```

```
%WRITE DATES
%Landsat 4
X1=[datenum('07-04-1992','mm-dd-yyyy') datenum('07-20-1992','mm-dd-yyyy')];
%Landsat 5
X2=[datenum('07-02-1994','mm-dd-yyyy') datenum('09-02-2005','mm-dd-yyyy') datenum('07-19-2006','mm-dd-yyyy') datenum('07-22-2007','mm-dd-yyyy') datenum('08-23-2007','mm-dd-yyyy') datenum('08-28-2009','mm-dd-yyyy') datenum('07-17-2011','mm-dd-yyyy') datenum('09-03-2011','mm-dd-yyyy')];
%Landsat 7
X3=[datenum('08-09-1999','mm-dd-yyyy') datenum('08-30-2001','mm-dd-yyyy')];
%Landsat 8
X4=[datenum('08-07-2013','mm-dd-yyyy') datenum('09-08-2013','mm-dd-yyyy') datenum('07-12-2015','mm-dd-yyyy') datenum('08-13-2015','mm-dd-yyyy') datenum('08-29-2015','mm-dd-yyyy')];

%JOIN X Y
X=[datenum('07-04-1992','mm-dd-yyyy'), datenum('07-20-1992','mm-dd-yyyy'), datenum('07-02-1994','mm-dd-yyyy'), datenum('08-09-1999','mm-dd-yyyy'), datenum('08-30-2001','mm-dd-yyyy'), datenum('09-02-2005','mm-dd-yyyy'), datenum('07-19-2006','mm-dd-yyyy'), datenum('07-22-2007','mm-dd-yyyy'), datenum('08-23-2007','mm-dd-yyyy'), datenum('08-28-2009','mm-dd-yyyy'), datenum('07-17-2011','mm-dd-yyyy'), datenum('09-03-2011','mm-dd-yyyy'), datenum('08-07-2013','mm-dd-yyyy'), datenum('09-08-2013','mm-dd-yyyy'), datenum('07-12-2015','mm-dd-yyyy'), datenum('08-13-2015','mm-dd-yyyy'), datenum('08-29-2015','mm-dd-yyyy')];
Y=[Y1, Y2(1,1), Y3, Y2(1,2), Y2(1,3), Y2(1,4), Y2(1,5), Y2(1,6), Y2(1,7), Y2(1,8), Y4];
Yb=[Y1b, Y2b(1,1), Y3b, Y2b(1,2), Y2b(1,3), Y2b(1,4), Y2b(1,5), Y2b(1,6), Y2b(1,7), Y2b(1,8), Y4b];
Yc=[Y1c, Y2c(1,1), Y3c, Y2c(1,2), Y2c(1,3), Y2c(1,4), Y2c(1,5), Y2c(1,6), Y2c(1,7), Y2c(1,8), Y4c];

%TS PLOT with Kriging
figure
%CDR
xy=[X(:) Y(:)];
[rnan,cnan]=find(isnan(xy));
xy(rnan,:)=[];
xy=xy.';
xd=xy(1,:);
zd=xy(2,:);
fk=xd(1,1);
dlk=length(xd);
fl=xd(1,dlk);
fx=[fk:1:fl];
plot(xd,zd,'ko')
hold on
nd=length(xd);
fn=length(fx);

for fi=1:nd
   for fj=1:nd
      fh(fi,fj)=abs(xd(fi)-xd(fj));
   end
end
% linear variogram
G=2*fh;
% spherical variogram
%G=3*fh/2-(fh.^3)/2;
%circular variogram
%G=1-(2/pi*cos(fh))+sqrt(1-(fh.^2));

for fk=1:fn
   for fi=1:nd
      hb(fi)=abs(xd(fi)-fx(fk));
   end
   % linear variogram
   Gb=2*hb;
   % spherical variogram
   %Gb=3*hb/2-(hb.^3)/2;
   %circular variogram
```

```
    %Gb=1-(2/pi*cos(hb))+sqrt(1-(hb.^2));
    J=ones(nd,1);
    L=[G  J;
       J' 0];
    P=[Gb 1]';
    fX=inv(L)*P;
    lam=fX(1:nd)';
    z(fk)=lam*zd';
    plot(fx(fk),z(fk),'k-')
end

%Ridge
xy2=[X(:) Yb(:)];
[rnan,cnan]=find(isnan(xy2));
xy2(rnan,:)=[];
xy2=xy2.';
xd2=xy2(1,:);
zd2=xy2(2,:);
fk2=xd2(1,1);
dlk2=length(xd2);
fl2=xd2(1,dlk2);
fx2=[fk2:1:fl2];
plot(xd2,zd2,'mo')
hold on
nd2=length(xd2);
fn2=length(fx2);

for fi2=1:nd2
    for fj2=1:nd2
        fh2(fi2,fj2)=abs(xd2(fi2)-xd2(fj2));
    end
end
% linear variogram
G2=2*fh2;
% spherical variogram
%G2=3*fh2/2-(fh2.^3)/2;
%circular variogram
%G2=1-(2/pi*cos(fh2))+sqrt(1-(fh2.^2));

for fk2=1:fn2
    for fi2=1:nd2
        hb2(fi2)=abs(xd2(fi2)-fx2(fk2));
    end
    % linear variogram
    Gb2=2*hb2;
    % spherical variogram
    %Gb2=3*hb2/2-(hb2.^3)/2;
    %circular variogram
    %Gb2=1-(2/pi*cos(hb2))+sqrt(1-(hb2.^2));
    J2=ones(nd2,1);
    L2=[G2  J2;
        J2' 0];
    P2=[Gb2 1]';
    fX2=inv(L2)*P2;
    lam2=fX2(1:nd2)';
    z2(fk2)=lam2*zd2';
    plot(fx2(fk2),z2(fk2),'m-')
end

%PIF
xy3 = [X(:) Yc(:)];
[rnan,cnan] = find(isnan(xy3));
xy3(rnan,:) = [];
xy3=xy3.';
xd3 = xy3(1,:);
zd3 = xy3(2,:);
```

```
fk3=xd3(1,1);
dlk3=length(xd3);
fl3=xd3(1,dlk3);
fx3=[fk3:1:fl3];
plot(xd3,zd3,'ro')
hold on
nd3=length(xd3);
fn3=length(fx3);

for fi3=1:nd3
   for fj3=1:nd3
      fh3(fi3,fj3)=abs(xd3(fi3)-xd3(fj3));
   end
end
% linear variogram
G3=2*fh3;
% spherical variogram
%G3=3*fh3/2-(fh3.^3)/2;
%circular variogram
%G3=1-(2/pi*cos(fh3))+sqrt(1-(fh3.^2));

for fk3=1:fn3
   for fi3=1:nd3
      hb3(fi3)=abs(xd3(fi3)-fx3(fk3));
   end
   % linear variogram
   Gb3=2*hb3;
   % spherical variogram
   %Gb3=3*hb3/2-(hb3.^3)/2;
   %circular variogram
   %Gb3=1-(2/pi*cos(hb3))+sqrt(1-(hb3.^2));
   J3=ones(nd3,1);
   L3=[G3  J3;
      J3' 0];
   P3=[Gb3 1]';
   fX3=inv(L3)*P3;
   lam3=fX3(1:nd3)';
   z3(fk3)=lam3*zd3';
   plot(fx3(fk3),z3(fk3),'r-')
end

xlabel('Time [years]') % x-axis label
ylabel('Index Value') % y-axis label
datetick('x','yyyy','keepticks');
title('FMI TIME SERIES (1992-2015) WITH KRIGING')
%COMPARE LANDSAT CDR x LANDSAT RIDGE x LANDSAT PIF
%minimum
min1=min(zd)
min2=min(zd2)
min3=min(zd3)
%maximum
max1=max(zd)
max2=max(zd2)
max3=max(zd3)
%mean
mean1=mean(zd)
mean2=mean(zd2)
mean3=mean(zd3)
%max-min
dif1=max1-min1
dif2=max2-min2
dif3=max3-min3
%standard deviation
so1=std(zd)
so2=std(zd2)
so3=std(zd3)
```

2. Skript do MATLAB – RRN GUI

```
function varargout = GUI_RRN_LINEARREGRESSION(varargin)
% GUI_RRN_LINEARREGRESSION MATLAB code for GUI_RRN_LINEARREGRESSION.fig
%      GUI_RRN_LINEARREGRESSION, by itself, creates a new GUI_RRN_LINEARREGRESSION or raises the
existing
%      singleton*.
%
%      H = GUI_RRN_LINEARREGRESSION returns the handle to a new GUI_RRN_LINEARREGRESSION or
the handle to
%      the existing singleton*.
%
%      GUI_RRN_LINEARREGRESSION('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in GUI_RRN_LINEARREGRESSION.M with the given input arguments.
%
%      GUI_RRN_LINEARREGRESSION('Property','Value',...) creates a new GUI_RRN_LINEARREGRESSION or
raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before GUI_RRN_LINEARREGRESSION_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to GUI_RRN_LINEARREGRESSION_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help GUI_RRN_LINEARREGRESSION

% Last Modified by GUIDE v2.5 18-Mar-2016 09:27:50

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @GUI_RRN_LINEARREGRESSION_OpeningFcn, ...
                   'gui_OutputFcn',  @GUI_RRN_LINEARREGRESSION_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
% End initialization code - DO NOT EDIT

% --- Executes just before GUI_RRN_LINEARREGRESSION is made visible.
function GUI_RRN_LINEARREGRESSION_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to GUI_RRN_LINEARREGRESSION (see VARARGIN)

% Choose default command line output for GUI_RRN_LINEARREGRESSION
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
```

```
% UIWAIT makes GUI_RRN_LINEARREGRESSION wait for user response (see UIRESUME)
% uiwait(handles.figure1);
axes(handles.axes1);
logo=imread('Data_GUI\logo.jpg');
imshow(logo)

% --- Outputs from this function are returned to the command line.
function varargout = GUI_RRN_LINEARREGRESSION_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in exit.
function exit_Callback(hObject, eventdata, handles)
% hObject    handle to exit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
fig = handles.figure1;
close(fig)

% --- Executes on button press in loader.
function loader_Callback(hObject, eventdata, handles)
% hObject    handle to loader (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

%READ DATA
axes(handles.axes2);
x=imread('Data_GUI\X.tif');
imshow(mat2gray(x))
%imshow(x)
axes(handles.axes3);
y=imread('Data_GUI\Y.tif');
imshow(mat2gray(y))
%imshow(y)
x=double(x);
y=double(y);

%NAN
x(find(x==0)) = NaN;
y(find(y==0)) = NaN;

xy = [x(:) y(:)];
[rnan,cnan] = find(isnan(xy));
xy(rnan,:) = [];
xy=xy.';
xx = xy(1,:).';
yy = xy(2,:).';

%LINEAR REGRESSION
format long
p = polyfit(xx,yy,1)
axes(handles.axes5);
scatter(xx,yy);
title('Before RRN')
xlabel('Master - [SR]') % x-axis label
ylabel('Slave - [SR]') % y-axis label
refline
slope=p(1,1)
```

```
intercept=p(1,2)

%DO GUI_RRN_LINEARREGRESSION
y2=(yy-intercept)/slope;
axes(handles.axes6);
scatter(xx,y2);
title('After RRN')
xlabel('Master - [SR]') % x-axis label
ylabel('Slave - [SR]') % y-axis label
refline
q = polyfit(xx,y2,1);
slope2=q(1,1)
intercept2=q(1,2)

yii=num2str(slope);
set(handles.text2,'String',slope)
yii=num2str(intercept);
set(handles.text4,'String',intercept)
yii=num2str(slope2);
set(handles.text9,'String',slope2)
yii=num2str(intercept2);
set(handles.text13,'String',intercept2)

% --- Executes on button press in loader2.
function loader2_Callback(hObject, eventdata, handles)
% hObject    handle to loader2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes on button press in dorrn.
function dorrn_Callback(hObject, eventdata, handles)
% hObject    handle to dorrn (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

3. Skript do MATLAB – TS GUI (příklad NDVI Ridge)

```
function varargout = GUI_TS_NDVI_Ridge(varargin)
% GUI_TS_NDVI_Ridge MATLAB code for GUI_TS_NDVI_Ridge.fig
%      GUI_TS_NDVI_Ridge, by itself, creates a new GUI_TS_NDVI_Ridge or raises the existing
%      singleton*.
%
%      H = GUI_TS_NDVI_Ridge returns the handle to a new GUI_TS_NDVI_Ridge or the handle to
%      the existing singleton*.
%
%      GUI_TS_NDVI_Ridge('CALLBACK',hObject,eventData,handles,...) calls the local
%      function named CALLBACK in GUI_TS_NDVI_Ridge.M with the given input arguments.
%
%      GUI_TS_NDVI_Ridge('Property','Value',...) creates a new GUI_TS_NDVI_Ridge or raises the
%      existing singleton*.  Starting from the left, property value pairs are
%      applied to the GUI before GUI_TS_NDVI_Ridge_OpeningFcn gets called.  An
%      unrecognized property name or invalid value makes property application
%      stop.  All inputs are passed to GUI_TS_NDVI_Ridge_OpeningFcn via varargin.
%
%      *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%      instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
% Edit the above text to modify the response to help GUI_TS_NDVI_Ridge
% Last Modified by GUIDE v2.5 24-Mar-2016 17:44:27
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @GUI_TS_NDVI_Ridge_OpeningFcn, ...
                   'gui_OutputFcn',  @GUI_TS_NDVI_Ridge_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
% End initialization code - DO NOT EDIT
% --- Executes just before GUI_TS_NDVI_Ridge is made visible.
function GUI_TS_NDVI_Ridge_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to GUI_TS_NDVI_Ridge (see VARARGIN)
% Choose default command line output for GUI_TS_NDVI_Ridge
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes GUI_TS_NDVI_Ridge wait for user response (see UIRESUME)
% uiwait(handles.figure1);
%LOAD LOGO
axes(handles.logo);
logo=imread('Data_GUI\logo.jpg');
imshow(logo)
%colorbar
%XI AND YI FOR LISTBOX
xi=0;
yi=0;
```

```
% --- Outputs from this function are returned to the command line.
function varargout = GUI_TS_NDVI_Ridge_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;
% --- Executes on selection change in listbox1.
function listbox1_Callback(hObject, eventdata, handles)
% hObject    handle to listbox1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns listbox1 contents as cell array
%        contents{get(hObject,'Value')} returns selected item from listbox1



% --- Executes during object creation, after setting all properties.
function listbox1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to listbox1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');

end
% --- Executes on button press in loader.
function loader_Callback(hObject, eventdata, handles)
% hObject    handle to loader (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of loader
axes(handles.axes1);

%READ IMAGES
%Landsat 4
a=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index1Yd.tif');
b=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index2Yd.tif');
%Landsat 5
c=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index3Yd.tif');
d=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index4Yd.tif');
e=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index5Yd.tif');
f=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index6Yd.tif');
g=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index7Yd.tif');
h=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index8Yd.tif');
i=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index9Yd.tif');
j=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index10Yd.tif');
%Landsat 7
k=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index11Yd.tif');
l=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index12Yd.tif');

%Landsat 8
m=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index13Yd.tif');
n=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index14Yd.tif');
o=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index15Yd.tif');
p=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index16Yd.tif');
q=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index17Yd.tif');
```

```
%CHOOSE PIXELS
%Map
index0=imread('Data_GUI\index0.tif');
%Sattelite Image
%index0=imread('Data_GUI\index0b.tif');
[xi,yi,Y]=impixel(index0);
%impixelinfo
%impixelregion
imshow(index0)
hold on
plot(xi,yi,'ro')
title('PIXEL LOCATION FOR TIME SERIES')
%Landsat 4
Y1=[impixel(a,xi,yi) impixel(b,xi,yi)];
Y1=[Y1(1,1,1) Y1(1,4,1)];
%Landsat 5
Y2=[impixel(c,xi,yi) impixel(d,xi,yi) impixel(e,xi,yi) impixel(f,xi,yi) impixel(g,xi,yi) impixel(h,xi,yi) impixel(i,xi,yi)
impixel(j,xi,yi)];
Y2=[Y2(1,1,1) Y2(1,4,1) Y2(1,7,1) Y2(1,10,1) Y2(1,13,1) Y2(1,16,1) Y2(1,19,1) Y2(1,22,1)];
%Landsat 7
Y3=[impixel(k,xi,yi) impixel(l,xi,yi)];
Y3=[Y3(1,1,1) Y3(1,4,1)];
%Landsat 8
Y4=[impixel(m,xi,yi) impixel(n,xi,yi) impixel(o,xi,yi) impixel(p,xi,yi) impixel(q,xi,yi)];
Y4=[Y4(1,1,1) Y4(1,4,1) Y4(1,7,1) Y4(1,10,1) Y4(1,13,1)];
axes(handles.axes23);
hold on
plot(xi,yi,'ro')

%WRITE DATES
%Landsat 4
X1=[datenum('07-04-1992','mm-dd-yyyy') datenum('07-20-1992','mm-dd-yyyy')];
%Landsat 5
X2=[datenum('07-02-1994','mm-dd-yyyy') datenum('09-02-2005','mm-dd-yyyy') datenum('07-19-2006','mm-
dd-yyyy') datenum('07-22-2007','mm-dd-yyyy') datenum('08-23-2007','mm-dd-yyyy') datenum('08-28-
2009','mm-dd-yyyy') datenum('07-17-2011','mm-dd-yyyy') datenum('09-03-2011','mm-dd-yyyy')];
%Landsat 7
X3=[datenum('08-09-1999','mm-dd-yyyy') datenum('08-30-2001','mm-dd-yyyy')];
%Landsat 8
X4=[datenum('08-07-2013','mm-dd-yyyy') datenum('09-08-2013','mm-dd-yyyy') datenum('07-12-2015','mm-
dd-yyyy') datenum('08-13-2015','mm-dd-yyyy') datenum('08-29-2015','mm-dd-yyyy')];

%SAVE TXT
%dlmwrite('DNvaluesForPlot.txt',[Y1 Y2 Y3 Y4],'delimiter','\t','precision',3);
%dlmwrite('DNvaluesForPlot1.txt',[Y1],'delimiter','\t','precision',3);
%dlmwrite('DNvaluesForPlot2.txt',[Y2],'delimiter','\t','precision',3);
%dlmwrite('DNvaluesForPlot3.txt',[Y3],'delimiter','\t','precision',3);
%dlmwrite('DNvaluesForPlot4.txt',[Y4],'delimiter','\t','precision',3);
%dlmwrite('PositionOfXY.txt',[xi yi]);

%GUI_TS_NDVI_Ridge PLOT
X=[datenum('07-04-1992','mm-dd-yyyy'), datenum('07-20-1992','mm-dd-yyyy'), datenum('07-02-1994','mm-
dd-yyyy'), datenum('08-09-1999','mm-dd-yyyy'), datenum('08-30-2001','mm-dd-yyyy'), datenum('09-02-
2005','mm-dd-yyyy'), datenum('07-19-2006','mm-dd-yyyy'), datenum('07-22-2007','mm-dd-yyyy'),
datenum('08-23-2007','mm-dd-yyyy'), datenum('08-28-2009','mm-dd-yyyy'), datenum('07-17-2011','mm-dd-
yyyy'), datenum('09-03-2011','mm-dd-yyyy'), datenum('08-07-2013','mm-dd-yyyy'), datenum('09-08-
2013','mm-dd-yyyy'), datenum('07-12-2015','mm-dd-yyyy'), datenum('08-13-2015','mm-dd-yyyy'),
datenum('08-29-2015','mm-dd-yyyy')];
Y=[Y1, Y2(1,1), Y3, Y2(1,2), Y2(1,3), Y2(1,4), Y2(1,5), Y2(1,6), Y2(1,7), Y2(1,8), Y4];

axes(handles.axes2);
%fig=figure;
xy = [X(:) Y(:)];
[rnan,cnan] = find(isnan(xy));
```

```
xy(rnan,:) = [];
xy=xy.';
xd = xy(1,:);
zd = xy(2,:);
plot(xd,zd,'ko-','LineWidth', 1)
title('NDVI LANDSAT CDR RRN TIME SERIES (1992-2015)')
xlabel('Time [years]') % x-axis label
ylabel('Index Value') % y-axis label
legend('NDVI Landsat CDR with Ridge RRN','Location','northeast')
hold on
datetick('x','yyyy','keepticks');
hold off

%SAVE GUI_TS_NDVI_Ridge PLOT
%print(fig,'TimeSeriesPlotNDVI','-dpng')
%print('TimeSeriesPlotNDVI','-dpng', '-painters')

%ZOOMED AREAS
xii=xi-50;
yii=yi+50;
axes(handles.axes3);
I=imcrop(a,[xii yii 100 100]);
imshow(I)
axes(handles.axes4);
I=imcrop(b,[xii yii 100 100]);
imshow(I)
axes(handles.axes5);
I=imcrop(c,[xii yii 100 100]);
imshow(I)
axes(handles.axes6);
I=imcrop(d,[xii yii 100 100]);
imshow(I)
axes(handles.axes7);
I=imcrop(e,[xii yii 100 100]);
imshow(I)
axes(handles.axes8);
I=imcrop(f,[xii yii 100 100]);
imshow(I)
axes(handles.axes9);
I=imcrop(g,[xii yii 100 100]);
imshow(I)
axes(handles.axes10);
I=imcrop(h,[xii yii 100 100]);
imshow(I)
axes(handles.axes11);
I=imcrop(i,[xii yii 100 100]);
imshow(I)
axes(handles.axes12);
I=imcrop(j,[xii yii 100 100]);
imshow(I)
axes(handles.axes13);
I=imcrop(k,[xii yii 100 100]);
imshow(I)
axes(handles.axes14);
I=imcrop(l,[xii yii 100 100]);
imshow(I)
axes(handles.axes15);
I=imcrop(m,[xii yii 100 100]);
imshow(I)
axes(handles.axes16);
I=imcrop(n,[xii yii 100 100]);
imshow(I)
axes(handles.axes17);
I=imcrop(o,[xii yii 100 100]);
imshow(I)
axes(handles.axes18);
```

```
I=imcrop(p,[xii yii 100 100]);
imshow(I)
axes(handles.axes19);
I=imcrop(q,[xii yii 100 100]);
imshow(I)

xxxx='X location:';
set(handles.text35,'String',xxxx)
yii=num2str(xi);
set(handles.text36,'String',xii)
yyyy='Y location:';
set(handles.text37,'String',yyyy)
yii=num2str(yi);
set(handles.text38,'String',yii)

% --- Executes on button press in exit.
function exit_Callback(hObject, eventdata, handles)
% hObject    handle to exit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
fig = handles.figure1;
close(fig)

% --- Executes on button press in imagechange.
function imagechange_Callback(hObject, eventdata, handles)
% hObject    handle to imagechange (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
%LOAD IMAGE IN LEFT FROM LISTBOX
contents = get(handles.listbox1,'Value');
axes(handles.axes23)
    if (contents == 1);
        a=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index1Yd.tif');
        imshow(a);
        colorbar
        axis on
        grid on
        title('LANDSAT NDVI')
    elseif (contents == 2);
        b=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index2Yd.tif');
        imshow(b);
        colorbar
        axis on
        grid on
        title('LANDSAT NDVI')
    elseif (contents == 3);
        c=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index3Yd.tif');
        imshow(c);
        colorbar
        axis on
        grid on
        title('LANDSAT NDVI')
    elseif (contents == 4);
        d=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index4Yd.tif');
        imshow(d);
        colorbar
        axis on
        grid on
        title('LANDSAT NDVI')
    elseif (contents == 5);
        e=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index5Yd.tif');
        imshow(e);
        colorbar
        axis on
        grid on
        title('LANDSAT NDVI')
```

```
elseif (contents == 6);
    f=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index6Yd.tif');
    imshow(f);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 7);
    g=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index7Yd.tif');
    imshow(g);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 8);
    h=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index8Yd.tif');
    imshow(h);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 9);
    i=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index9Yd.tif');
    imshow(i);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 10);
    j=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index10Yd.tif');
    imshow(j);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 11);
    k=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index11Yd.tif');
    imshow(k);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 12);
    l=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index12Yd.tif');
    imshow(l);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 13);
    m=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index13Yd.tif');
    imshow(m);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 14);
    n=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index14Yd.tif');
    imshow(n);
    colorbar
    axis on
    grid on
    title('LANDSAT NDVI')
elseif (contents == 15);
    o=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index15Yd.tif');
    imshow(o);
    colorbar
```

```
        axis on
        grid on
        title('LANDSAT NDVI')
    elseif (contents == 16);
        p=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index16Yd.tif');
        imshow(p);
        colorbar
        axis on
        grid on
        title('LANDSAT NDVI')
    elseif (contents == 17);
        q=imread('Vegetation_Indices_CDR_RRNCDR\NDVI\NDVIY\index17Yd.tif');
        imshow(q);
        colorbar
        axis on
        grid on
        title('LANDSAT NDVI')
    end

% --- Executes during object creation, after setting all properties.
function text35_CreateFcn(hObject, eventdata, handles)
% hObject    handle to text35 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
```

4. Databáze Time Series (1992 – 2015) na microSD kartě:

   Kompletní příloha v elektronické podobě na externím disku a zmenšená verze pro odevzdání na microSD kartě.

   a. Landsat 4-8 družicové snímky s Fmask a RRN
   b. CDR vegetační indexy pro Nízké Tatry a okolí
   c. RRN vegetační indexy pro Nízké Tatry a okolí
   d. Skripty do Matlab pro potřebné analýzy
   e. Soubor s metadaty (přehledem užitých snímků, možných exportů, užitých formulací pro tvorbu vegetačních indexů a stromová struktura uložení)

5. Elektronická verze diplomové práce na CD