

```

%Tracking of PW values
%Copyright: Matthew R. Igel 2019 University of California
%Requires the Image Processing Toolbox

%clear, clean, and time
clear
clc
tic

%Change directory to location of data
cd('F:\TrackPW')

%Set's the minimum PW threshold to define a "region"
%I tried 40:5:65
PW_thresh=55;%55;

%Read in PW from netCDF
PW=ncread('MERRA_tqv_0.67x0.5deg_1997-2015_dailyavg.nc','tqv');

%Determine length of PW (in time)
[~,~,il]=size(PW);

%Create a small subdomain
%Size depends on region of interest
PWsmallout=ones(84+2,81+2,il+2);

%Set binary marked based on PW_thresh on subdomain
for i=1:il
    PWsmall=squeeze(PW(338:421,41:121,i));
    PWsmall(PWsmall<PW_thresh)=0;
    PWsmall(PWsmall~=0)=1;
    PWsmallout(2:end-1,2:end-1,i+1)=PWsmall;
end

%Call the image processing toolbox
%Returns a struct of connected regions in time and space
Objects=bwconncomp(PWsmallout);

clear PWsmall PWsmallout PW

%For each region, identify the bounding box
BB=regionprops(Objects,'BoundingBox');

%Extracts geometric lengths for convenience
TimeLength=NaN(1,Objects.NumObjects);
Area=NaN(1,Objects.NumObjects);
for i=2:Objects.NumObjects
    TimeLength(i)=BB(i).BoundingBox(6);
    Area(i)=(BB(i).BoundingBox(4)*2/3)*(BB(i).BoundingBox(5)*0.5);
end

histogram(TimeLength(2:end))

toc

%%

count=0;
%Determine the properties of long-lived PW regions in the same way we did
%above for all regions
for i=2:Objects.NumObjects

```

```

%Make sure objects last at least 5 days
if TimeLength(i)>4
    count=count+1;
    PW=zeros(84+2,81+2,i1+2);
    PW(Objects.PixelIdxList{i})=1;
    %Record long-lived PW regions longevity
    LongObjects{count}.Times=BB(i).BoundingBox(3)+.5:BB(i).BoundingBox(3)+BB(i).BoundingBox(6)+.5;
    for k=LongObjects{count}.Times(1):LongObjects{count}.Times(end)
        %Record the geometric properties of potentially serval regions
        %at any one time that are part of the long-lived region. We're
        %taking 2D snapshots and recording centroids.
        tempObjects=bwconncomp(squeeze(PW(:,:,k+1)));
        tempCent=regionprops(tempObjects,'Centroid');
        for j=1:tempObjects.NumObjects
            LongObjects{count}.Centroids(k-
LongObjects{count}.Times(1)+1,j,1:2)=tempCent(j).Centroid;
        end
    end
end
end
end

clear tempObjects tempCent

toc

%%

PWRegion=NaN(1,4);

%Constructs the useful output vector, 'PWRegion'. The vector is composed
%of columns: 1) LongObject number 2) Gregorian day 3) latitude 4) longitude
%5) year 6) doy 7) month 8) day of month.
%Example: If the first 'LongObject' has 3 2D regions at some point in its
% lifetime, there will be 3 centroid locations recorded for each day the
% 'LongObject' exists. Some locations are NaNs. This implies fewer than
% the maximum number of regions occur on that day. So, day 1 might have
% 3 centroid (i.e. 3 2D regions), day 2 might have 2 centroids and a NaN
% centroid (2 of the origional regions merged), day 3 might have 1
% centroid and 2 NaNs (the 2 remaining regions merged), day 4 might have
% 2 centroids and a NaN (the region split).

%Insert centroids to PWRegion
for i=1:length(LongObjects)
    [Times,Pieces,~]=size(LongObjects{i}.Centroids);
    Rows=Times*Pieces;
    count=0;
    for j=1:Times
        for k=1:Pieces
            count=count+1;
            A(count,1:4)=[i 2450449+LongObjects{i}.Times(j),___
                LongObjects{i}.Centroids(j,k,1),___
                LongObjects{i}.Centroids(j,k,2)];
        end
    end
end

if i==1
    PWRegion(1:Rows,1:4)=A;
else
    PWRegion=[PWRegion; A];
end
end

```

```

clear A
end

%Replace rogue zeros with NaN
PWRegion(PWRegion==0)=NaN;

%Convert centroids to lat/long
lat=ncread('MERRA_tqv_0.67x0.5deg_1997-2015_dailyavg.nc','latitude');
long=ncread('MERRA_tqv_0.67x0.5deg_1997-2015_dailyavg.nc','longitude');
l=~isnan(PWRegion(:,3));
PWRegion(l,3)=lat(round(PWRegion(l,3))+40); %Test Switch
PWRegion(l,4)=long(round(PWRegion(l,4))+337);%Test Switch

%rows of PWRegion
[i12,~]=size(PWRegion);

%For each PWRegion find year and convert to day of year
%This should probably be written more sensibly if this code is to be used
%for files other than the one I used (Matt)
for i=1:i12
    JD=PWRegion(i,2);
    if JD<2450815
        yr=1997;
        doy=JD-2450450+1;
    elseif JD<2451180
        yr=1998;
        doy=JD-2450815+1;
    elseif JD<2451545
        yr=1999;
        doy=JD-2451180+1;
    elseif JD<2451911
        yr=2000;
        doy=JD-2451545+1;
    elseif JD<2452276%5
        yr=2001;
        doy=JD-2451911+1;
    elseif JD<2452641
        yr=2002;
        doy=JD-2452276+1;%5
    elseif JD<2453006
        yr=2003;
        doy=JD-2452641+1;
    elseif JD<2453372
        yr=2004;
        doy=JD-2453006+1;
    elseif JD<2453737
        yr=2005;
        doy=JD-2453372+1;
    elseif JD<2454102
        yr=2006;
        doy=JD-2453737+1;
    elseif JD<2454467
        yr=2007;
        doy=JD-2454102+1;
    elseif JD<2454833
        yr=2008;
        doy=JD-2454467+1;
    elseif JD<2455198
        yr=2009;
        doy=JD-2454833+1;
    elseif JD<2455563

```

```

        yr=2010;
        doy=JD-2455198+1;
    elseif JD<2455928
        yr=2011;
        doy=JD-2455563+1;
    elseif JD<2456294
        yr=2012;
        doy=JD-2455928+1;
    elseif JD<2456659
        yr=2013;
        doy=JD-2456294+1;
    elseif JD<2457024
        yr=2014;
        doy=JD-2456659+1;
    elseif JD>=2457024
        yr=2015;
        doy=JD-2457024+1;
    end

    %Fill time stamps for PWRegions
    PWRegion(i,5)=yr;
    PWRegion(i,6)=doy;
    [~,PWRegion(i,7),PWRegion(i,8)]=julian2greg(JD);

end

%Make the 'short version' of PWRegion. (On RAMADDA server as case list)
PWRegions_Short=NaN(length(LongObjects),3);
for i=1:length(LongObjects)
    Ind=find(PWRegion(:,1)==i,1,'first');
    PWRegions_Short(i,:)=PWRegion(Ind,5) PWRegion(Ind,7) PWRegion(Ind,8)];
end

toc

```