CNH Eigenvector Instructions (General notes)

This step-by-step instruction was created by Evan Casey with advice from Dr. Li An in March 2016. The intent is to create a set of eigenvectors that can be used to filter out possible spatial autocorrelation effects in the regression conducted by Abigail Sullivan. The regression is multilevel regression model aiming to predict local people's willingness (0 for not willing and 1 for willing) to participating in Mikania (an invasive species) control in Chitwan Valley, Nepal based on a survey of around 1,235 households. The input data are an Excel spreadsheet with household ID and latitude and longitude data (i.e., CNH_HH_Data_Edit_4132015.xlsx).

[The blue lines below are the ways the datasets are stored/structured or some comments to clarify some lines. You can skip them]

PATH = C:\Users\[User Name]\Documents\CNH\

```
CNH\ Directory Contents:CNH.gdb\
        Data\
                Eigen\
                        [Contents created in step 20]
                Neighbors\
                        [Contents created in step 8 and 15]
                Shapefiles\
                        [Contents created in step 3]
                Tables\
                        CNH_HH_Data_Edit_4132015.xlsx
                        [The original data file that is used to create the eigenvector spatial filters.

                       Also including additional contents created in step 7]
        Meta\
                CNH_Eigvec_Instructions.docx  [THIS meta-data file]
        Scripts\
                CNH.r           [R code that is used]
                CNH.sas         [SAS code that is used]
```

Phase I (Steps 1 to 4): Create an ArcGIS shapefile with X/Y data.

1) Launch ArcMap
2) File > Add Data > Add XY Data…
    a) File: PATH\Data\Tables\CNH_HH_Data_Edit_4132015.xlsx
    b) X Field: G_1_2 [The variable for longitude]
    c) Y Field: G_1_1 [The variable for latitude]
    d) Z Field: <None>

e) Coordinate System of Input Coordinates: [Edit]
i) Geographic Coordinate Systems > World > WGS 1984
ii) [OK]
f) [OK]

3) In Table Of Contents, right-click CNH_HH_Data_Edit_4132015.xlsx Events
a) Data > Export Data…
i) Output feature class: PATH\Data\Shapefiles\CNH_HH_Data_Edit_4132015.shp
ii) [OK]

4) Close ArcMap

Phase II (Steps 5 to 10): Create spatial neighborhoods (1/0 binary data) in GeoDa.

5) Launch GeoDa

6) File > New Project From > ESRI Shapefile (*.shp)
a) File name: PATH\Data\Shapefiles\CNH_HH_Data_Edit_4132015.shp
b) [Open]

7) File > Export…
a) Export file name: PATH\Data\Tables\CNH_HH_Data_Edit_4132015.csv
b) Create a project file?: [un-check]
c) [Export]

8) Tools > Weights > Create
a) Weights File ID Variable: OBJECTID
b) k-Nearest Neighbors: [Check]
c) Number of neighbors: 10
d) [Create]
i) File name: PATH\Data\Neighbors\CNH_HH_Data_Edit_41322015_knn10.gwt
ii) Save as type: GWT files (*.gwt)
iii) [Save]
e) Repeat steps c) and d), replacing 10 (k-nearest neighbors) with 20, 30, 40 and finally 50
f) [Close]

9) File > Close Project

10) Close GeoDa

Phase III (Steps 11 to 16): Convert the 1/0 binary data (in GeoDa format) to CSV format.

11) Launch R

12) File > Open script…
a) File name: PATH\Scripts\CNH.r
b) [Open]

13) If spdep and maptool libraries not installed execute the following commands:

```
install.packages(“spdep”)
install.packages(“maptools”)
```

14) Change setwd to PATH/Data
15) Edit > Run all (Creates *.csv neighbor matrixes at PATH\Data\Neighbors\)
16) Close R

Phase IV (Steps 17 to 21): Create Eigenvectors in SAS using the output files in Phase III.

17) Launch SAS
18) File > Open Program…
    a) File name: PATH\Scripts\CNH.sas
    b) [Open]
19) Alter following variables to match correct path, keeping the file name (e.g. &data_set..csv) intact:

input_1	PATH\Data\Tables\
input_2	PATH\ Data\Neighbors\
output_1	PATH\Data\Eigen\

20) Run > Submit (Creates *.csv tables with eigenvector data at PATH\Data\Eigen\)
21) Close SAS

Below is the content in CNH.r (the R code)

```
setwd("C:/Users/edc/Documents/CNH/Data/")

library(spdep)

library(maptools)

shapeList <- c("CNH_HH_Data_Edit_4132015")

gtwList <- c("10knn", "20knn", "30knn", "40knn", "50knn")

for (i in 1:length(shapeList)) {

        gh <- data.frame(readShapePoints(paste("Shapefiles/", shapeList[[i]], ".shp", sep="")))

        attach(gh)

        for (j in 1:length(gtwList)) {

                gh.nb <- read.gwt2nb(paste("Neighbors/", shapeList[[i]], "_", gtwList[[j]], ".gwt",
sep=""), region.id=OBJECTID)

                gh.mat <- nb2mat(gh.nb, style="B", zero.policy=TRUE)
```

```
                        colnames(gh.mat) <- paste("v", 1:ncol(gh.mat), sep="")

                        write.table(gh.mat, file=paste("Neighbors/", shapeList[[i]], "_", gtwList[[j]], ".csv",
sep=""), sep=",", row.names=FALSE, col.names=TRUE)

            }

}
```

<u>Below is the content in CNH.SAS (the SAS code)</u>

```
%LET data_sets=CNH_HH_Data_Edit_4132015;
%LET csv_neighbors=10knn 20knn 30knn 40knn 50knn;

%MACRO loop_data_sets (data_sets_loop);
      %DO i=1 %TO %SYSFUNC(COUNTW(&data_sets_loop));
            %LET data_set=%SCAN(&data_sets_loop,&i);

            FILENAME input_1
"C:/Users/edc/Documents/CNH/Data/Tables/&data_set..csv" TERMSTR=CRLF;

            PROC IMPORT DATAFILE=input_1
                              OUT=cnh_data
                              DBMS=CSV
                              REPLACE;
            RUN;

            PROC SORT DATA=cnh_data;
                  BY OBJECTID;
            RUN;

            %loop_neighbors(data_set_loop=&data_set);

      %END;
%MEND loop_data_sets;

%MACRO loop_neighbors (data_set_loop);
      %DO j=1 %TO %SYSFUNC(COUNTW(&csv_neighbors));
            %LET csv_neighbor=%SCAN(&csv_neighbors,&j);

            FILENAME input_2
"C:/Users/edc/Documents/CNH/Data/Neighbors/&data_set_loop._&csv_neighbor..csv
" TERMSTR=CRLF;

            PROC IMPORT DATAFILE=input_2
                            OUT=cnh_neighbor
                            DBMS=CSV
                            REPLACE;
            RUN;

            %LET dsid=%SYSFUNC(OPEN(cnh_neighbor));
            %LET n=%SYSFUNC(ATTRN(&dsid,NLOBS));
            %LET rc=%SYSFUNC(CLOSE(&dsid));
            PROC IML;
                  USE cnh_neighbor;
                  READ ALL VAR _ALL_ INTO C;
                  ones=J(&n,1,1);
                  ident=I(&n);
                  b=ones*ones`/&n;
                  M=ident-b;
                  MCM=M*C*M;
                  CALL EIGEN(e_val,e_vec,MCM) VECL="l_vec";

                  CREATE eig_val FROM e_val [COLNAME="eigen"];
                  APPEND FROM e_val;
                        CLOSE eig_val;

                  CREATE eig_vec FROM e_vec;
                  APPEND FROM e_vec;
                        CLOSE eig_vec;

                  sumC=SUM(C);
                  PRINT sumC[LABEL="sumC (&data_set_loop csv_neighbor)"];
                  CALL SYMPUT("sum_cmv", CHAR(sumC));
            QUIT;

            DATA _NULL_;
                  SET eig_val;
                  IF _N_=1 THEN
                        CALL SYMPUT("max_eig_val", eigen);
                  ELSE STOP;
            RUN;

            DATA moran_i;
                  SET eig_val;
                  MI=eigen*&n/&sum_cmv;
                  adj_eigen=eigen/&max_eig_val;
            RUN;

            DATA eigen_regressors;
                  SET eig_vec;
                  KEEP COL1-COL100;
            RUN;

            DATA eigen_regressors_2;
                  SET eigen_regressors;
                  OBJECTID=_n_;
            RUN;

            DATA cnh_eigen_data;
                  MERGE cnh_data eigen_regressors_2;
                  BY OBJECTID;
            RUN;

            FILENAME output_1
"C:/Users/edc/Documents/CNH/Data/Eigen/&data_set_loop._&csv_neighbor._Eigvec.
csv" TERMSTR=CRLF;

            PROC EXPORT DATA=cnh_eigen_data
```

```
                            OUTFILE=output_1
                            DBMS=CSV
                            REPLACE;

      %END;
%MEND loop_neighbors;

%loop_data_sets(data_sets_loop=&data_sets);
```