Imputation of rents and values of offices

Jifei Ban and Richard Arnott

June 16th, 2011

Table of Contents

1 Impute against Costar Office data to Costar Office data.

a. Abstract of the imputation procedure:

We use geographically weighted regression to impute unknown rents and values of offices in L.A. metro area for the year 2000. For rents imputation, the responding variable is logarithm of annual rent per square footage for the first quarter of year 2008, the explanatory variables are distance to CBD, age of building, distance to the nearest subcenter, distance to the ocean, and distance to the nearest freeway. For values imputation, the responding variable is logarithm of value per square footage[1], the explanatory variables are sale year (categorical variable), distance to CBD, age of building, distance to the nearest subcenter, distance to the ocean, and distance to the nearest freeway. Distances are all in miles.

***b. Data**:*

Costar Office data: total.xls
Employment Subcenters coordinates; CBD coordinates
Coastline shape file
Freeway shape file
Model Zone shape file

c. The imputing procedure

Step 1

In the costar office data, not only many rents and values are missing, but also variables including YearBuilt, Latitude and Longitude, which are used to predict the explanatory variables, are missing. We use the mean of observed YearBuilt to impute the missing YearBuilt. For those records with addresses, we use addresses to find latitudes and longitudes. We use http://www.gpsvisualizer.com/geocoder/ to convert multiple addresses to GPS coordinates selecting Google as the source. We discard 4 records of which the addresses are missing. Costar office data includes a few records from counties outside the six counties of L.A. metro area, but we only use data from the six counties.

Besides, we load the coastline shape file, freeway shape file, and costar office shape file in ArcGIS. Using the Near function in Proximity in Analysis Tools, we calculate the distances of each office to the nearest coastline and the nearest freeway. We store these distances in “ocean.txt” and “fwy.txt” respectively.

Step 2

After the data is prepared, we can start imputing the missing rents and values.

R code

Section 1 parameters and functions

```
rm(list=ls())
CBD<-t(as.matrix(c(-118.446305110092,34.0615632358611)))
```

[1] The value per square footage is calculated as Last Sale Price divided by Rentable Building Area.

```
geodist<-function(center_coor,coordinates)
  {
   a<-57.2958
   dist_center<-3963*acos( outer(sin(center_coor[,2]/a),sin(coordinates[,2]/a))+
   outer(cos(center_coor[,2]/a),cos(coordinates[,2]/a))*cos(outer(-
center_coor[,1]/a,coordinates[,1]/a,"+")))
  return(dist_center)
  }
con<-1609.344
```

Section 1 discussion

CBD stores the coordinates of the TAZ which has the highest employment density. Function “geodist” calculates the distances in miles between two sets of coordinates. “con” means 1 mile=1609.344 meters.

Section 2 Load data

```
 coor<-read.csv(file="subcoord.csv")

subid<-as.numeric(as.matrix(as.data.frame(table(coor$subcenterid))$Var1))[-1]

subcoor<-matrix(0,nrow=51,ncol=3)
for (k in 1:length(subid))
{
   for(i in 1:dim(coor)[1])
     {
       if (coor$subcenterid[i]==subid[k])
         subcoor[k,]<-subcoor[k,]+
c(coor$EMP03[i]*coor$longitude[i],coor$EMP03[i]*coor$latitud[i],coor$EMP03[i])
        }
}

subcoor1<-cbind(subcoor[,1]/subcoor[,3],subcoor[,2]/subcoor[,3])

costar.obs<-read.csv("costar.csv")

subdist<-geodist(subcoor1,cbind(costar.obs$Longitude,costar.obs$Latitude))

costar.obs$fsub<- apply(subdist,2,min)

costar.obs$CBD<-as.vector(geodist(CBD,cbind(costar.obs$Longitude,costar.obs$Latitude)))

costar.obs$ocean<-as.vector(t(read.table("C:/Users/Ban/Desktop/landvalue/ocean.txt")))/con
costar.obs$fwy<-as.vector(t(read.table("C:/Users/Ban/Desktop/landvalue/fwy.txt")))/con
```

```
coobs<-data.frame(ID=costar.obs$ID,CBD=costar.obs$CBD,fsub=costar.obs$fsub,ocean=costar.obs$ocean,fwy=costar.obs$fwy)
write.csv(coobs,"C:/Users/Ban/Desktop/landvalue/costarobs.csv",row.names=F)

avarent<-read.csv("C:/Users/Ban/Desktop/landvalue/avarent.csv")
missrent<-read.csv("C:/Users/Ban/Desktop/landvalue/missrent.csv")
avarent$age<-2000-avarent$Year_Built
avarent$lgrent<-log(avarent$Average_Weighted_Rent)

missrent$age<-2000-missrent$Year_Built
missrent$lgrent<-missrent$Average_Weighted_Rent

avavalue<-read.csv("C:/Users/Ban/Desktop/landvalue/avavalue.csv")
avavalue$saleyr<-as.factor(avavalue$saleyr)
missvalue<-read.csv("C:/Users/Ban/Desktop/landvalue/value2000.csv")
missvalue$saleyr<-factor(2000,levels=1989:2008)

avavalue$age<-2000-avavalue$Year_Built
avavalue$lgvalue<-log(avavalue$valsq)

missvalue$age<-2000-missvalue$Year_Built
missvalue$lgvalue<-0
```

Section 2 discussion

"subcoord.csv" is a file containing columns of TAZ ID, coordinates of the TAZ, subcenter id, and total employment of 2003. “subcoor1” stores the coordinates of each subcenter weighted by the employment size of its components TAZ’s. "costar.csv" is the file we get from the original costar office data after ***step 1***. “costar.obs$fsub” is the vector of the distances of the offices to the nearest subcenter; “costar.obs$CBD” is the vector of the distances of the offices to the CBD; “costar.obs$ocean” is the vector of the distances of the offices to the ocean; “costar.obs$fwy” is the vector of the distances of the offices to the nearest freeway. We write these variables out to a file “costarobs.csv”. We combine “costarobs.csv” with “costar.csv” in SAS. Then from there, we generate the file “avarent.csv” [2]containing the records of which rents are available, and we generate the file “missren.csv” containing the records of which rents are missing. Similarly, “avavalue.csv” [3]contains those records with value per square footage and “missvalue.csv” contains the records of which value per square footage is unknown.

[2] This data set is a subset of observations of the costar office data, whose rent is available and attributes used as explanatory variables in the regression model (such as latitude, longitude, YearBuilt, etc) are not missing.

[3] This data set is a subset of observations of the costar office data, whose value per square footage is available and attributes used as explanatory variables in the regression model (such as latitude, longitude, YearBuilt, etc) are not missing.

Section 3 Imputation

```
library(spgwr)

####################Rent
ava.map<-
SpatialPointsDataFrame(data=avarent,coords=cbind(avarent$Longitude,avarent$Latitude))

miss.map<-
SpatialPointsDataFrame(data=missrent,coords=cbind(missrent$Longitude,missrent$Latitude))


gwr.sel(lgrent~CBD+age+fsub+ocean+fwy,data=avarent,adapt=T,coords=cbind(avarent$Longitude,avarent$Latitude),gweight=gwr.bisquare,longlat=T)

agwr<-
gwr(lgrent~CBD+age+fsub+ocean+fwy,data=avarent,coords=cbind(avarent$Longitude,avarent$Latitude),predictions=T
,adapt= 0.02595615,gweight=gwr.bisquare,longlat=T)

bgwr<-
gwr(lgrent~CBD+age+fsub+ocean+fwy,data=avarent,coords=cbind(avarent$Longitude,avarent$Latitude)
,adapt= 0.02595615, predictions=T,fit.points=miss.map,gweight=gwr.bisquare,longlat=T)

missrent$Average_Weighted_Rent<-exp(bgwr$SDF$pred)

write.csv(missrent," missrent1.csv",row.names=F)


###################Value

ava1.map<-
SpatialPointsDataFrame(data=avavalue,coords=cbind(avavalue$Longitude,avavalue$Latitude))

miss1.map<-
SpatialPointsDataFrame(data=missvalue,coords=cbind(missvalue$Longitude,missvalue$Latitude))


gwr.sel(lgvalue~saleyr+CBD+age+fsub+ocean+fwy,data=avavalue,adapt=T,coords=cbind(avavalue$Longitude,avavalue$Latitude),gweight=gwr.bisquare,longlat=T)
```

```
agwr1<-
gwr(lgvalue~saleyr+CBD+age+fsub+ocean+fwy,data=avavalue,coords=cbind(avavalue$Longitu
de,avavalue$Latitude),predictions=T
,adapt=0.51,gweight=gwr.bisquare,longlat=T)


bgwr1<-
gwr(lgvalue~saleyr+CBD+age+fsub+ocean+fwy,data=avavalue,coords=cbind(avavalue$Longitu
de,avavalue$Latitude)
,adapt=0.51,predictions=T, fit.points=miss1.map,gweight=gwr.bisquare,longlat=T)

missvalue$valsq<-exp(bgwr$SDF$pred)

write.csv(missvalue," missvalue1.csv",row.names=F)
```

Section 3 Discussion

The Geographically Weighted Regression function is in the library “spgwr”. First, we use “gwr.sel” to find the best window size. The weight function we choose is bisquare function, which is very similar to tricubic function. For the imputation of rents, the best window size is 0.02595615. “agwr” contains regression results based on the records of which rents are available; “bgwr” contains predicted logarithm rents (imputed logarithm rents) based on the records of which rents are available. For the imputation of values, the best window size is 0.51. “agwr1” contains regression results based on the records of which values are available; “bgwr1” contains predicted logarithm values of year 2000 (imputed logarithm values) based on the records of which values are available.

Regression Result from Step 2

Rents

Mean of coefficients

Intercept	CBD	age	fsub	ocean	fwy
5.605241	-0.1125505	-0.005535731	-0.0752174	-0.01547721	-0.007329758

Mean of local R-square 0.3347958

Global R-square 0.6505489

Values

Mean of coefficients

Intercept	Saleyr1990	Saleyr1991	Saleyr1992	Saleyr1993	Saleyr1994
5.262	0.1247	:-0.077694	-0.1540	-0.74709	-0.7400

```
Saleyr1995   Saleyr1996    Saleyr1997    Saleyr1998    Saleyr1999      Saleyr2000
-0.5522        -0.49040       -0.277246      -0.12674       0.11867         0.08800
Saleyr2001     Saleyr2002   Saleyr2003     Saleyr2004   Saleyr2005      Saleyr2006
0.09285         0.22017        0.39518         0.5341          0.7559           1.3228
Saleyr2007     Saleyr2008    CBD                age              fsub             ocean
1.7942           0.39667         -0.013794      -0.0074416   -0.016555   -0.012215
 fwy
-0.048755
```

The reference year is 1989.
Mean of local R-square   0.3221145
Global R-square   0.3354557

Remark: If we decide to add county dummies into the regression, I suggest we do regression for 6 counties separately. Because if we do not do it separately, when we impute for an office in Riverside, we will need at least one observation from LA (and other counties), which will cause a very large window size and over smooth the regression.

Step 3
The rent we imputed is for the first quarter of year 2008. Using the following table provided by Bill Wheaton, We can deflate the rent to the first quarter of year 2000, treating Ventura as Orange County, San Bernardino and Imperial as Riverside County.

TW Rent Index ($/SF)	Los Angeles	Orange County	Riverside
2000.1	$21.38	$25.76	$16.15
2008.1	$27.94	$29.54	$22.12

We update the original costar office data in ***step 1*** with imputed rents and imputed values of year 2000 and generate a shape file from it using the latitude and longitude of each office. Load the shape file and the model zone shape file in ArcGIS. Using the Spatial Join function in Overlay in Analysis Tools, we determine which office is within which model zone. Then we produce the square footage weighted average rents and values for each model zone. Notice that Rentable Building Area is missing for 8 records. We impute the missing values by the mean of Rentable Building Area so that we can calculate the square footage weighted rents and values.

2 Prepare Office Data in Parcel 2007 for imputation

Here what we want to do is draw all the office parcels from parcel 2007 database. Impute YearBuilt if YearBuilt is missing. Calculate accessibility variables (distances).

Step1
Draw the office data from parcel database, impute YearBuilt, and then combine 6 counties' office data.

SAS code

```
%macro readatt(data=);
proc import out=&data
             datafile="C:\Users\Ban\Downloads\&data..DBF"
                    DBMS=DBF replace;
                    getdel=no;
                    run;

                    proc sql number;
                    create table &data._off as
                    select *
                    from &data
                    where lu08 in (1210,1211,1212,1213);
                    quit;
%mend readatt;

%macro imputeyear(data=);
proc sql noprint;
alter table &data
add flagyearbuilt num format=1.;

update &data
set flagyearbuilt=
case
when yearbuilt>0 then 0
when yearbuilt=200 then 1
else 1 end;

update &data
set yearbuilt=
case when yearbuilt=200 then 2000
else yearbuilt
end;

select * from &data;
quit;
%mend imputeyear;


/*************** Riverside ******************/
%readatt(data=RIV_ATT)

%imputeyear(data=riv_off)

proc sql;
select mean(yearbuilt)
from riv_off
where yearbuilt>0 ;
quit;

proc sql noprint;
update riv_off as u
set yearbuilt=1982
where yearbuilt=0 or yearbuilt is missing;
select * from riv_off;
quit;

proc import out=riv_county
            datafile="I:parcel2007\riverside_county.dbf"
                    DBMS=DBF replace;
```

```
                   getdel=no;
                   run;


proc sql;
create table riv_off1 as
select *
from riv_county as u, riv_off as v
where u.scag_xyid=v.scagxyid;
quit;

/******************San Bernardino ************/
%readatt(data=SBN_ATT)

%imputeyear(data=sbn_off)

proc sql;
select mean(yearbuilt)
from sbn_off
where yearbuilt>0 ;
quit;

proc sql noprint;
update sbn_off as u
set yearbuilt=1977
where yearbuilt=0 or yearbuilt is missing;
select * from sbn_off;
quit;

proc import out=sbn_county
datafile="I:parcel2007\san_bernardino_county.dbf"
DBMS=DBF replace;
getdel=no;
run;

proc sql;
create table sbn_off1 as
select *
from sbn_county as u, sbn_off as v
where u.scag_xyid=v.scagxyid;
quit;

/******************Ventura ****************************/
%readatt(data=VEN_ATT)

%imputeyear(riv_off=ven_off)

proc sql;
select mean(yearbuilt)
from ven_off
where yearbuilt>0 ;
quit;

proc sql noprint;
update ven_off as u
```

```
set yearbuilt=1975
where yearbuilt=0 or yearbuilt is missing;
select * from ven_off;
quit;

proc import out=ven_county
            datafile="I:parcel2007\ventura_county.dbf"
                    DBMS=DBF replace;
                    getdel=no;
                    run;

proc sql;
create table ven_off1 as
select *
from ven_county as u, ven_off as v
where u.scag_xyid=v.scagxyid;
quit;

/******************Imperial ****************************/
%readatt(data=IMP_ATT)


%imputeyear(riv_off=imp_off)
proc sql;
select mean(yearbuilt)
from imp_off
where yearbuilt>0 ;
quit;

proc sql noprint;
update imp_off as u
set yearbuilt=2000
where yearbuilt=0 or yearbuilt is missing;
select * from imp_off;
quit;

proc import out=imp_county
            datafile="I:parcel2007\imperial_county.dbf"
                    DBMS=DBF replace;
                    getdel=no;
                    run;

proc sql;
create table imp_off1 as
select *
from imp_county as u, imp_off as v
where u.scag_xyid=v.scagxyid;
quit;


/********orange*****/
%readatt(data=ORA_ATT)

%imputeyear(riv_off=ora_off)
proc sql;
```

```
select mean(yearbuilt)
from ora_off
where yearbuilt>0 ;
quit;

proc sql noprint;
update ora_off as u
set yearbuilt=1968
where yearbuilt=0 or yearbuilt is missing;
select * from ora_off;
quit;

proc import out=ora_county
            datafile="I:parcel2007\orange_county.dbf"
                    DBMS=DBF replace;
                    getdel=no;
                    run;

proc sql;
create table ora_off1 as
select *
from ora_county as u, ora_off as v
where u.scag_xyid=v.scagxyid;
quit;


/*************los angels*****************/

%readatt(data=LAX_ATT)

%imputeyear(riv_off=lax_off)
proc sql;
select mean(yearbuilt)
from lax_off
where yearbuilt>0 ;
quit;

proc sql noprint;
update lax_off as u
set yearbuilt=1972
where yearbuilt=0 or yearbuilt is missing;
select * from lax_off;
quit;

proc import out=lax_county
            datafile="I:parcel2007\los_angeles_county.dbf"
                    DBMS=DBF replace;
                    getdel=no;
                    run;

proc sql;
create table lax_off1 as
select *
from lax_county as u, lax_off as v
where u.scag_xyid=v.scagxyid;
```

```
quit;


data parcel_off;
set

lax_off1 (keep= scagxyid x y lu08 shape_area lotsqft impsqft totvalue07
yearbuilt  officeclass  )
ora_off1 (keep= scagxyid x y lu08 shape_area lotsqft impsqft totvalue07
yearbuilt  officeclass )
riv_off1 (keep= scagxyid x y lu08 shape_area lotsqft impsqft totvalue07
yearbuilt  officeclass )
ven_off1  (keep= scagxyid x y lu08 shape_area lotsqft impsqft totvalue07
yearbuilt officeclass )
imp_off1 (keep= scagxyid x y lu08 shape_area lotsqft impsqft totvalue07
yearbuilt  officeclass )
sbn_off1  (keep= scagxyid x y lu08 shape_area lotsqft impsqft totvalue07
yearbuilt  officeclass )
;
run;

PROC EXPORT DATA= WORK.parcel_off
            OUTFILE= "I:\Imputation\parcel_off.dbf"
            DBMS=dbf REPLACE;
RUN;
```

Discussion of Step 1:

%readatt*(**data=**)** reads parcel attribute files and pick out office parcels. %imputeyear***(data=) add a new column flagyearbuilt to office parcles. If YearBuilt is missing or probably a typo (YearBuilt=200), flagyearbuilt=1; otherwise, 0. For each county, we impute missing YearBuilt of offices with the mean of YearBuilt of offices in that county, and then join the office data with each county's parcel GIS data by SCAGXYID. After that we combine the 6 counties' office data as one file "parcel_off.dbf".

Step 2

Read coordinates X&Y data from "parcel_off.dbf" in ArcGIS. Save the file as a shape file. We load the coastline shape file and freeway shape file in ArcGIS. Using the Near function in Proximity in Analysis Tools, we calculate the distances of each office to the nearest coastline and the nearest freeway, and add those two columns to parcel office data.

Step 3

Calculate distances of offices to CBD, and the nearest subcenter.

R code

```
library(rgdal)
library(foreign)
parceloff<-read.dbf("C:/Users/Ban/Desktop/landvalue/parcel_off.dbf")
```

```
xy<-cbind(parcel$X,parceloff$Y)
xy1<-project(xy, "+proj=utm +zone=11 ellps=WGS84",inv=T)
subdist<-geodist(subcoor1,xy1)
parceloff$fsub<- apply(subdist,2,min)
parceloff$CBD<-as.vector(geodist(CBD,xy1))
```

Discussion of Step 3

Because the coordinates in parcel office data are under UTM 11N, we need to first project them to WGS84 so that they will be compatible with the coordinates of CBD and subcenters. “subcoor1” is the vector of coordinates of subcenters.

Appendix Nov 19th, 2011

This appendix discusses some unmentioned matters about the costar office data itself.

1. There are 11925 observations in total, after we discard observations without both addresses and geographical coordinates or outside the L.A. metro area. 7115 observations have a missing last sale value. 8406 observations have a missing rent per square foot.

2. There is one observation, with its attribute describing that it belongs to San Bernardino County, actually falling outside of San Bernardino County slightly, which can be identified by plotting the coordinates of the parcel in ArcGIS.

3. The costar office data has one attribute called “Building Class”. The value of that attribute is only observed in costar office data as A or B. We did not use “Building Class” as a predictor, because there is no such attribute in the parcel data.

4. Huiling Zhang has identified a few unusually high values per square foot in costar office data. Questions about the investigation of this issue can be directed to her.