APPENDIX D – DETAILED QUANTITATIVE RESULTS AND R CODE

Table 1: Winnipeg – Negative binomial regression spatial lag results

Independent Variables	Coefficients
percent visible minority	2.33 *
percent aboriginal	-1.70 .
percent first generation	5.80 ***
average household income	1.45^5***
percent owned units	-2.01 ***
percent single-detached dwelling	0.21
average value of dwelling (owner-occupied)	-3.99^6*
average gross rent	1.59^3***

---

significance codes: 0 (***) 0.001 (**) 0.01 (*) 0.05 (.)

residual deviance:

Min	1q	Median	3q	Max
-2.97	-0.66	-0.10	0.37	2.36

Table 2: Providence – Negative binomial regression spatial lag results

Independent Variables	Coefficients
percent white	-0.35
percent Latino	-0.76
percent foreign born	9.70 ***
percent naturalized citizen	-6.13
median household income	-8.63^6
percent owned unit	2.39 *
percent rented unit	3.29 **
median house value	1.37^6
median gross rent	7.61^4*
percent subsidized units	1.35

---

significance codes: 0 (***) 0.001 (**) 0.01 (*) 0.05 (.)

residual deviance:

Min	1q	Median	3q	Max
-3.91	-1.36	-0.40	0.48	3.60

Table 3: Copenhagen – Poisson regression spatial lag results

Independent Variables	Coeff cients
percent foreign born	-2.43 ***
percent non-citizen	0.32 ***
average family income	-4.53 ^6***
percent owned units	0.17 ***
percent private rent units	-6.96 ***
percent social housing units	-9.51 ***
percent limited cooperative units	0.16 ***
percent private cooperative units	-6.02 ***
percent public units	-0.48 ***

---

signif cance codes: 0 (***) 0.001 (**) 0.01 (*) 0.05 (.)

residual deviance:

Min	1q	Median	3q	Max
-33.02	-7.07	1.74	8.42	29.59

R code – Winnipeg, Providence, Copenhagen

```
##Regressions for new migrant settlement##
##Winnipeg##
library(spdep)

##processing data read in and incorporate neighbour weight matrix created in Geoda
#creating matched id
n.id = c(1:230)

#reading in data file
data = read.table ("peg_data_june2013.txt", header=T, sep=",",skip=1)

#adding id
data$n.id=n.id
data = data [order (data$n.id),]

#rename because you already wrote the rest of the code that way
names (data) = c ("AREA", "x", "y", "NEI_num", "totpop", "tothous","ab", "vismin",
 "imi","housinc","own","rent", "averent", "avehousval", "detach", "mig", "N2001_TOT_ ",
 "N2001_FIRS", "nrc", "DELTA_FB_0", "n.id")

#reading in weight matrix
data_weight <- read.gal("peg_rook.GAL", override=T)
attributes (data_weight)

#converting weight to list
data_rook <- nb2listw (data_weight, zero.policy=T)
data_rook$weights

#transforming data to matrix
attach (data)
data_matrix <- cbind (mig, totpop, vismin, ab, imi, housinc, own, rent, detach, avehousval,
 averent, tothous, nrc)
detach(data)

#created data list with spatial lag
data_lagged <- lag.listw(data_rook, data_matrix, zero.policy=T)

#transform spatial lag list to data frame
data_lag_db = as.data.frame (data_lagged)
names (data_lag_db) = c ("mig", "totpop", "vismin", "ab", "imi", "housinc", "own", "rent", "detach",
```

```
 "avehousval", "averent", "tothous", "nrc")
data_lag_db$neigh = data$NEI_NUM
data_lag_db$n.id = data$n.id
data_lag_db = data_lag_db [order (data_lag_db$n.id),]

##test variable relationships
layout(matrix(c(1,2,3,4,5,6,7,8),2,4, byrow=T))
plot ((log (mig+1)), pv)
plot ((log (mig+1)), pa)
plot ((log (mig+1)), pi)
plot ((log (mig+1)), (log(housinc)))
plot ((log (mig+1)), po)
plot ((log (mig+1)), pr)
plot ((log (mig+1)), (log(avehousval)))
plot ((log (mig+1)), (log(averent)))

layout(matrix(c(1,2,3,4,5,6,7,8),2,4, byrow=T))
plot (mig, pv)
plot (mig, pa)
plot (mig, pi)
plot (mig, housinc)
plot (mig, po)
plot (mig, pr)
plot (mig, avehousval)
plot (mig, averent)

##negative binomial, spatial lag data, independent percents
attach (data_lag_db)
pv = (vismin/totpop)
pa = (ab /totpop)
pi = (imi/totpop)
po = (own/tothous)
pr = (rent/tothous)
pd = (detach/tothous)
reg <- glm (mig ~ pv + pa + pi + housinc + po +pr + pd+ avehousval + averent, family =
 negative.binomial (2))
summary (reg)

#test residuals for spatial autocorrelation
reg_moran <- lm.morantest (reg, data_rook, zero.policy=T)

plot(reg)
```

```
#mapping residuals
residuals = data.frame(resid (reg))
write.table (residuals, "peg_residuals.csv",sep=",")

residgeo = data.frame(data$NEI_num, data$n.id)
write.table (residgeo, "residgeo.csv",sep=",")


##Providence

#creating matched id
n.id = c(1:234)

#reading in data file
data = read.table ("var3_05-09_shp.txt", header=T, sep=",",skip=1)

#adding id
data$n.id=n.id
data = data [order (data$n.id),]

#reading in weight matrix
data_weight <- read.gal("weight_rook.GAL", override=T)
attributes (data_weight)

#converting weight to list
data_rook <- nb2listw (data_weight, zero.policy=T)
data_rook$weights

#transfoming data to matrix
attach (data)
data_matrix <- cbind (MIG, TOT_POP, WHT_NONLAT, LAT, FB, NAT_, HOUS_INC, OWN,
 RENT, HOUS_VAL, GROSS_RENT, UNITS, TOT_HOUS, SUB_COUNT, CDC_TYPE)
detach (data)


#created data list with spatial lag
data_lagged <- lag.listw(data_rook, data_matrix, zero.policy=T)

#transform spatial lag list to data frame
data_lag_db = as.data.frame (data_lagged)
names (data_lag_db) = c ("MIG", "TOT_POP", "WHT_NONLAT", "LAT", "FB", "NAT_",
```

```
 "HOUS_INC", "OWN", "RENT", "HOUS_VAL", "GROSS_RENT", "UNITS", "TOT_HOUS",
 "SUB_COUNT", "CDC_TYPE")
data_lag_db$GEO = data$GEO
data_lag_db$n.id = data$n.id
data_lag_db = data_lag_db [order (data_lag_db$n.id),]


##poisson regression
attach (data)
reg <- glm (MIG ~ TOT_POP + WHT_NONLAT + LAT + FB + NAT_ + HOUS_INC + OWN +
 HOUS_VAL + GROSS_RENT + UNITS + offset (TOT_POP), start = coef(lm(MIG ~ TOT_POP
 + WHT_NONLAT + LAT + FB + NAT_ + HOUS_INC + OWN + HOUS_VAL + GROSS_RENT +
 UNITS), family = poisson))
summary (reg)

#test residuals for spatial autocorrelation
reg_moran <- lm.morantest (reg, data_rook, zero.policy=T)

##test variable relationships
layout(matrix(c(1,2,3,4,5,6,7,8,9,10),2,5, byrow=T))
plot ((log (MIG+1)), pw)
plot ((log (MIG+1)), pl)
plot ((log (MIG+1)), pf)
plot ((log (MIG+1)), pn)
plot ((log (MIG+1)), (log(HOUS_INC)))
plot ((log (MIG+1)), po)
plot ((log (MIG+1)), pr)
plot ((log (MIG+1)), (log(HOUS_VAL)))
plot ((log (MIG+1)), (log(GROSS_RENT)))
plot ((log (MIG+1)), pu)


layout(matrix(c(1,2,3,4,5,6,7,8,9,10),2,5, byrow=T))
plot (MIG, pw)
plot (MIG, pl)
plot (MIG, pf)
plot (MIG, pn)
plot (MIG, HOUS_INC)
plot (MIG, po)
plot (MIG, pr)
plot (MIG, HOUS_VAL)
plot (MIG, GROSS_RENT)
```

```
plot (MIG, pu)

##negative binomial, spatial lag data, independent percents
attach (data_lag_db)
pw = (WHT_NONLAT/TOT_POP)
pl = (LAT/TOT_POP)
pf = (FB/TOT_POP)
pn = (NAT_/TOT_POP)
po = (OWN/TOT_HOUS)
pr = (RENT/TOT_HOUS)
pu = (UNITS/TOT_HOUS)
reg <- glm (MIG ~ pw + pl + pf + pn + HOUS_INC + po +pr + HOUS_VAL + GROSS_RENT +
 pu, family = negative.binomial (2))
summary (reg)

#test residuals for spatial autocorrelation
reg_moran <- lm.morantest (reg, data_rook, zero.policy=T)

plot(reg)

#mapping residuals
residuals = data.frame(resid (reg))
write.table (residuals, "pvd_residuals.csv",sep=",")

residgeo = data.frame(data$GEO data$n.id)
write.table (residgeo, "residgeo_Pvd.csv",sep=",")



##Copenhagen##
#creating matched id
n.id = c(1:25)

#reading in data file
data = read.table ("kbh_vars_stage3_jul2013.txt", header=T, sep=",")

#adding id
data$n.id=n.id
data = data [order (data$n.id),]

#reading in weight matrix
data_weight <- read.gal("kbh_rook.GAL", override=T)
```

```
attributes (data_weight)

#converting weight to list
data_rook <- nb2listw (data_weight, zero.policy=T)
data_rook$weights

#transforming data to matrix
attach (data)
data_matrix <- cbind (mig, tot.pop, non.fb, non_cit, fam_inc, own, rent, priv_rent, alem,
 limi_coop, pri_coop, public, other, occ)
detach(data)

#created data list with spatial lag
data_lagged <- lag.listw(data_rook, data_matrix, zero.policy=T)

#transform spatial lag list to data frame
data_lag_db = as.data.frame (data_lagged)
names (data_lag_db) = c("mig", "tot.pop", "non.fb", "non_cit", "fam_inc", "own", "rent",
 "priv_rent", "alem", "limi_coop", "pri_coop", "public", "other", "occ")
data_lag_db$cit = data$number
data_lag_db$n.id = data$n.id
data_lag_db = data_lag_db [order (data_lag_db$n.id),]

##check variable relationships
layout(matrix(c(1,2,3,4,5,6,7,8,9,10),2,5, byrow=T))
plot ((log (mig+1)), pnfb)
plot ((log (mig+1)), pnc)
plot ((log (mig+1)), (log(fam_inc)))
plot ((log (mig+1)), po)
plot ((log (mig+1)), ppr)
plot ((log (mig+1)), pa)
plot ((log (mig+1)), plc)
plot ((log (mig+1)), ppc)
plot ((log (mig+1)), pp)
plot ((log (mig+1)), po)

layout(matrix(c(1,2,3,4,5,6,7,8, 9,10),2,5, byrow=T))
plot (mig, pnfb)
plot (mig, pnc)
plot (mig, fam_inc)
plot (mig, po)
plot (mig, prr)
```

```
plot (mig, pa)
plot (mig, plc)
plot (mig, ppc)
plot (mig, pp)
plot (mig, po)

##poisson regression, spatial lag data, independent percents
attach (data_lag_db)
pnfb = (non.fb/tot.pop)
pnc = (non_cit /tot.pop)
po = (own/occ)
pr = (rent/occ)
ppr = (priv_rent/occ)
pa = (alem/occ)
plc = (limi_coop/occ)
ppc = (pri_coop/occ)
pp = (public/occ)
po = (other/occ)


reg <- glm (mig ~ pnfb + pnc + fam_inc + po + ppr + pa +  plc + ppc + pp + po, family = poisson)
summary (reg)

#test residuals for spatial autocorrelation
reg_moran <- lm.morantest (reg, data_rook, zero.policy=T)

plot(reg)

#mapping residuals
residuals = data.frame(resid (reg))
write.table (residuals, kph_esiduals.csv",sep=",")

residgeo = data.frame(data$NEI_num, data$n.id)
write.table (residgeo, "kph_residgeo.csv",sep=",")
```