Estatística Espacial

```
library(rgdal)
require(tmap)
library(spgwr)
library(spdep)
library(spatialreg)
```

Carregando dados

```
mg = rgdal::readOGR(dsn="crime_mg",layer="crime_mg")
```

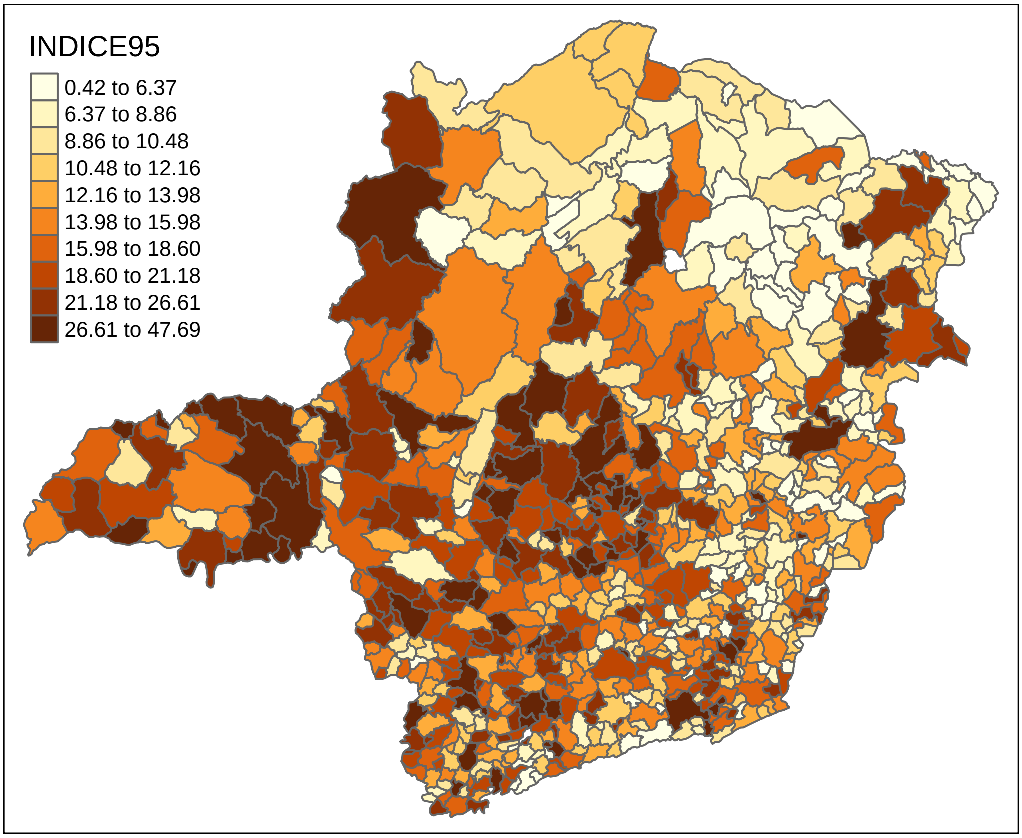
```
## OGR data source with driver: ESRI Shapefile
## Source: "/cloud/project/crime_mg", layer: "crime_mg"
## with 754 features
## It has 17 fields
## Integer64 fields read as strings:  POP_RUR POP_URB POP_FEM POP_MAS
```

```
mg$POP_RUR=as.numeric(mg$POP_RUR)
mg$POP_URB=as.numeric(mg$POP_URB)
mg$POP_FEM=as.numeric(mg$POP_FEM)
mg$POP_MAS=as.numeric(mg$POP_MAS)
mg$POP_TOT=as.numeric(mg$POP_TOT)
```

Tarefa 1

Produza um mapa de alta qualidade do shapefile crime_mg utilizando a extensão tmap. Apresente o código completo e o mapa produzido em sua resposta.

```
qtm(mg, fill="INDICE95", fill.n=10,
    fill.title="INDICE95", fill.style="quantile")
```



Pergunta 2

Qual das variáveis quantitativas apresentadas no shapefile crime_mg apresenta maior auto-correlação espacial? Descreva como implementou a matriz de vizinhança. Apresente o I de Moran e o mapa de auto-correlação espacial local (LISA map) da variável escolhida e também de pelo menos outras 3 variáveis.

A variável de maior auto-correlação espacial é AREA (I = 0.551), seguida por INDICE94 (I = 0.316) e INDICE95 (I = 0.303).

```
# Construir lista de vizinhos
sids_nbq <- poly2nb(mg, queen = TRUE)

# Criar matriz de pesos
sids_nbq_w <- nb2listw(sids_nbq)

moran.test(mg$AREA, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$AREA
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 25.381, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
```

```
## Moran I statistic       Expectation          Variance
##      0.5514077443     -0.0013280212      0.0004742435
```

```r
moran.test(mg$INDICE94, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$INDICE94
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 14.204, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.3157661993     -0.0013280212      0.0004983892
```

```r
moran.test(mg$INDICE95, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$INDICE95
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 13.608, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.3025179584     -0.0013280212      0.0004985351
```

```r
moran.test(mg$GINI_91, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$GINI_91
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 5.9405, p-value = 1.421e-09
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.1303874451     -0.0013280212      0.0004916158
```

```r
moran.test(mg$POP_94, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$POP_94
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 9.8057, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
```

```
##      0.1301703653     -0.0013280212      0.0001798394
```

```r
moran.test(mg$POP_RUR, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$POP_RUR
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 11.024, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.2421345018     -0.0013280212      0.0004877173
```

```r
moran.test(mg$POP_URB, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$POP_URB
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 9.3383, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.1157571185     -0.0013280212      0.0001572045
```

```r
moran.test(mg$POP_FEM, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$POP_FEM
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 9.1909, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.1171301230     -0.0013280212      0.0001661175
```

```r
moran.test(mg$POP_MAS, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$POP_MAS
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 9.493, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.1265234099     -0.0013280212      0.0001813857
```

```r
moran.test(mg$POP_TOT, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$POP_TOT
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 9.3382, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.1216123010     -0.0013280212      0.0001733254
```

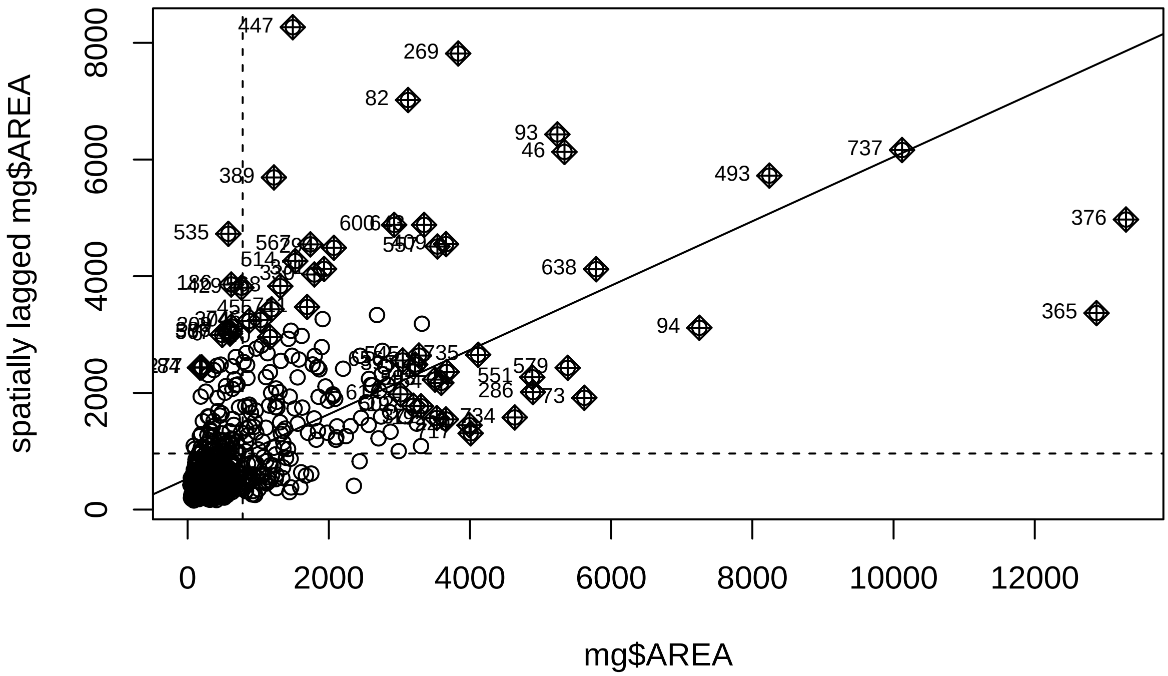
```r
moran.test(mg$URBLEVEL, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$URBLEVEL
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 13.097, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.2913346972     -0.0013280212      0.0004993312
```

```r
moran.test(mg$PIB_PC, listw = sids_nbq_w)
```

```
##
##  Moran I test under randomisation
##
## data:  mg$PIB_PC
## weights: sids_nbq_w
##
## Moran I statistic standard deviate = 5.7262, p-value = 5.135e-09
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.1222207337     -0.0013280212      0.0004655268
```

```r
moran.plot(
  mg$AREA,
  sids_nbq_w,
  labels=as.character(mg$ID)
  )
```

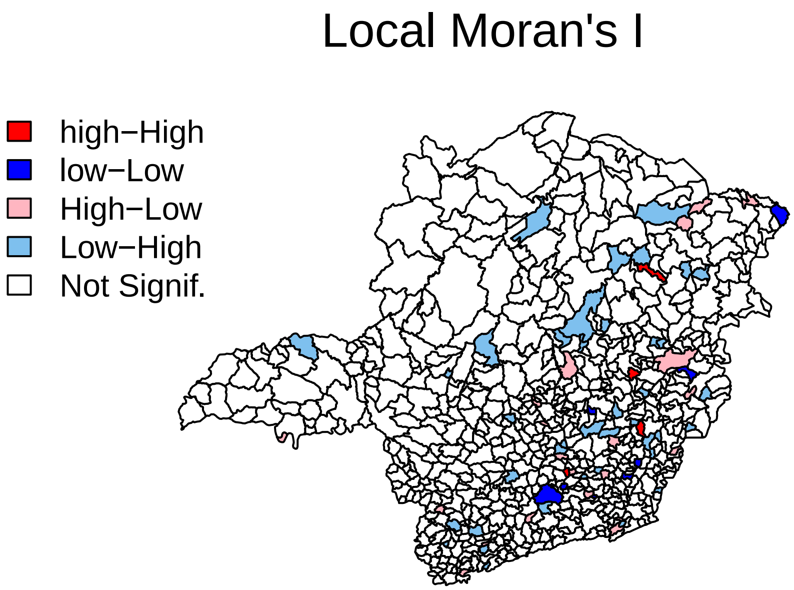



```r
# Set the breaks for the thematic map classes
breaks <- seq(1, 5, 1)

# Set the corresponding labels for the thematic map classes
labels <- c("high-High", "low-Low", "High-Low", "Low-High", "Not Signif.")

# see ?findInterval - This is necessary for making a map
np <- findInterval(mg$INDICE94, breaks)

# Assign colors to each map class
colors <- c("red", "blue", "lightpink", "skyblue2", "white")
plot(mg, col = colors[np])  #colors[np] manually sets the color for each county
mtext("Local Moran's I", cex = 1.5, side = 3, line = 1)
legend("topleft", legend = labels, fill = colors, bty = "n")
```



Pergunta 3

Implemente o modelo espacial auto-regressivo (SAR) da variável Indice95 (índice de criminalidade em 1995 de Minas Gerais) a partir de apenas uma variável independente (não pode ser Indice94, Codmuni, ID, X_coord nem Y_coord). Apresente o resultado da regressão linear simples e da regressão linear espacial. Apresente as equações e interprete seus coeficientes. Indique como criou a matriz de vizinhança.

Regressão Linear

```
lm.mg <- lm(INDICE95 ~ AREA,data=mg)
lm.mg
```

```
##
## Call:
## lm(formula = INDICE95 ~ AREA, data = mg)
##
## Coefficients:
## (Intercept)         AREA
##   1.504e+01    5.255e-04
```

```
summary(lm.mg)
```

```
##
## Call:
## lm(formula = INDICE95 ~ AREA, data = mg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.743  -5.883  -1.424   4.479  32.404
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.504e+01  3.425e-01  43.914   <2e-16 ***
## AREA        5.255e-04  2.391e-04   2.198   0.0283 *
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.89 on 752 degrees of freedom
## Multiple R-squared:  0.006383,   Adjusted R-squared:  0.005061
## F-statistic:  4.83 on 1 and 752 DF,  p-value: 0.02826
```

Rergressão SAR

```r
# Construir lista de vizinhos
sids_nbq <- poly2nb(mg, queen = TRUE)

# Criar matriz de pesos
sids_nbq_w <- nb2listw(sids_nbq)

sar.mg <- lagsarlm(
  INDICE95 ~ AREA,
  data=mg,
  sids_nbq_w,
  method="Matrix"
  )

summary(sar.mg)
```

```
##
## Call:lagsarlm(formula = INDICE95 ~ AREA, data = mg, listw = sids_nbq_w,
##     method = "Matrix")
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -15.9614  -5.0228  -1.2122   3.9451  31.9609
##
## Type: lag
## Coefficients: (asymptotic standard errors)
##               Estimate Std. Error z value  Pr(>|z|)
## (Intercept) 7.02014976 0.70193971 10.0011 < 2.2e-16
## AREA        0.00059072 0.00021526  2.7442  0.006065
##
## Rho: 0.50361, LR test value: 129.93, p-value: < 2.22e-16
## Asymptotic standard error: 0.042828
##     z-value: 11.759, p-value: < 2.22e-16
## Wald statistic: 138.28, p-value: < 2.22e-16
##
## Log likelihood: -2561.36 for lag model
## ML residual variance (sigma squared): 49.597, (sigma: 7.0425)
## Number of observations: 754
## Number of parameters estimated: 4
## AIC: 5130.7, (AIC for lm: 5258.6)
## LM test for residual autocorrelation
## test value: 13.49, p-value: 0.0002398
```

```r
SAR_SSE <- sar.mg$SSE
SST <- sum((mg$INDICE95 - mean(mg$INDICE95))^2)
r2_SAR <- 1 - (SAR_SSE/SST)
r2_SAR
```

```
## [1] 0.2062466
```

Pergunta 4

Para essa variável que você escolheu, o modelo espacial SAR apresentou ganhos significantes com relação ao modelo linear simples? Justifique sua resposta.

Pergunta 5

Implemente a regressão espacial GWR da variável Indice95 (índice de criminalidade em 1995 de Minas Gerais) a partir de apenas uma variável independente (não pode ser Indice94, Codmuni, ID, X__coord nem Y__coord). Apresente o resultado da regressão linear simples e da regressão linear espacial por GWR. Apresente medidas da distribuição dos coeficientes (min, Q1, Q2, Q3, máx), e da distribuição do R2 (min, Q1, Q2, Q3, máx) e apresente os resultados globais da regressão (R2 global, basicamente). Destaque a estratégia utilizada para a construção do kernel (fixo ou adaptativo, vizinhança, etc).

```r
bwGauss = gwr.sel(
  INDICE95 ~ AREA,
  data=mg,
  adapt=TRUE,
  method="aic",
  gweight=gwr.Gauss,
  verbose=FALSE
  )

gwr.ap = gwr(
  INDICE95 ~ AREA,
  data = mg,
  bandwidth=bwGauss,
  gweight=gwr.Gauss,
  adapt=bwGauss,
  hatmatrix=TRUE
  )

gwr.ap
```

```
## Call:
## gwr(formula = INDICE95 ~ AREA, data = mg, bandwidth = bwGauss,
##     gweight = gwr.Gauss, adapt = bwGauss, hatmatrix = TRUE)
## Kernel function: gwr.Gauss
## Adaptive quantile: 0.0132583 (about 9 of 754 data points)
## Summary of GWR coefficient estimates at data points:
##                      Min.     1st Qu.      Median     3rd Qu.        Max.
## X.Intercept.  5.87119478 11.13671509 13.32536380 16.77307460 26.89097874
## AREA         -0.01258273  0.00043952  0.00242702  0.00474641  0.02818024
##               Global
## X.Intercept. 15.0393
## AREA          0.0005
## Number of data points: 754
## Effective number of parameters (residual: 2traceS - traceS'S): 101.1285
## Effective degrees of freedom (residual: 2traceS - traceS'S): 652.8715
## Sigma (residual: 2traceS - traceS'S): 6.697497
## Effective number of parameters (model: traceS): 69.9958
## Effective degrees of freedom (model: traceS): 684.0042
## Sigma (model: traceS): 6.543302
```

```
## Sigma (ML): 6.23219
## AICc (GWR p. 61, eq 2.33; p. 96, eq. 4.21): 5055.97
## AIC (GWR p. 96, eq. 4.22): 4968.985
## Residual sum of squares: 29285.51
## Quasi-global R2: 0.378394
```

```
GWR_SSE <- gwr.ap$results$rss
r2_GWR <- 1 - (GWR_SSE/SST)
r2_GWR
```

```
## [1] 0.378394
```

Pergunta 6

Para essa variável que você escolheu, o modelo espacial GWR apresentou ganhos significantes com relação ao modelo linear simples? Justifique sua resposta.

Pergunta 7

Implemente um modelo de regressão linear multivariado stepwise da variável Indice95 (significante a 5% ou 10%, utilize o que achar melhor). Depois, "promova-o" a um modelo SAR. Indique como criou a matriz de vizinhança. Apresente os resultados comparados (equação, R2). Qual modelo você escolheria como final? Se desejar, apresente mapas que sustente sua justificativa.

```
lm.mg <- lm(
  INDICE95 ~ AREA + GINI_91 + POP_94 + POP_RUR + POP_URB + POP_FEM + URBLEVEL + PIB_PC ,
  data=mg
  )

lm.step = step(lm.mg)
```

```
## Start:  AIC=2852.51
## INDICE95 ~ AREA + GINI_91 + POP_94 + POP_RUR + POP_URB + POP_FEM +
##     URBLEVEL + PIB_PC
##
##            Df Sum of Sq   RSS    AIC
## - POP_94    1      29.8 32390 2851.2
## <none>                  32360 2852.5
## - PIB_PC    1     110.6 32471 2853.1
## - AREA      1     221.3 32582 2855.7
## - POP_RUR   1     255.0 32615 2856.4
## - POP_URB   1     292.2 32652 2857.3
## - POP_FEM   1     334.4 32695 2858.3
## - GINI_91   1    1039.3 33400 2874.3
## - URBLEVEL  1    5972.1 38332 2978.2
##
## Step:  AIC=2851.2
## INDICE95 ~ AREA + GINI_91 + POP_RUR + POP_URB + POP_FEM + URBLEVEL +
##     PIB_PC
##
##            Df Sum of Sq   RSS    AIC
## <none>                  32390 2851.2
## - PIB_PC    1     115.4 32505 2851.9
```

```
## - AREA      1     212.2 32602 2854.1
## - POP_RUR   1     266.5 32656 2855.4
## - POP_FEM   1     319.8 32710 2856.6
## - POP_URB   1     325.5 32716 2856.7
## - GINI_91   1    1055.8 33446 2873.4
## - URBLEVEL  1    6104.9 38495 2979.4
```

```r
summary(lm.step)
```

```
##
## Call:
## lm(formula = INDICE95 ~ AREA + GINI_91 + POP_RUR + POP_URB +
##     POP_FEM + URBLEVEL + PIB_PC, data = mg)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -14.015  -4.294  -1.127   3.419  32.054
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.132e+01  1.069e+00  10.587  < 2e-16 ***
## AREA         5.474e-04  2.476e-04   2.211  0.02734 *
## GINI_91     -1.251e+01  2.537e+00  -4.931 1.01e-06 ***
## POP_RUR      6.617e-04  2.671e-04   2.477  0.01346 *
## POP_URB      8.125e-04  2.967e-04   2.738  0.00633 **
## POP_FEM     -1.532e-03  5.644e-04  -2.714  0.00680 **
## URBLEVEL     1.831e+01  1.544e+00  11.858  < 2e-16 ***
## PIB_PC       1.606e-04  9.848e-05   1.631  0.10340
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.589 on 746 degrees of freedom
## Multiple R-squared:  0.3125, Adjusted R-squared:  0.306
## F-statistic: 48.44 on 7 and 746 DF,  p-value: < 2.2e-16
```

```r
sar.mg <- lagsarlm(
  lm.step,
  data=mg,
  sids_nbq_w,
  method="Matrix"
  )

summary(sar.mg)
```

```
##
## Call:
## lagsarlm(formula = lm.step, data = mg, listw = sids_nbq_w, method = "Matrix")
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -14.1205  -4.0436  -1.0144   3.2973  31.6477
##
## Type: lag
## Coefficients: (asymptotic standard errors)
##                Estimate  Std. Error z value  Pr(>|z|)
```

```
## (Intercept)  7.0028e+00  1.1982e+00  5.8444 5.084e-09
## AREA         4.3384e-04  2.3602e-04  1.8382   0.06603
## GINI_91     -1.1573e+01  2.4206e+00 -4.7812 1.743e-06
## POP_RUR      5.1505e-04  2.5443e-04  2.0243   0.04294
## POP_URB      5.7543e-04  2.8312e-04  2.0325   0.04211
## POP_FEM     -1.0811e-03  5.3854e-04 -2.0074   0.04470
## URBLEVEL     1.5976e+01  1.4938e+00 10.6953 < 2.2e-16
## PIB_PC       9.2380e-05  9.3934e-05  0.9835   0.32539
##
## Rho: 0.3222, LR test value: 53.338, p-value: 2.8078e-13
## Asymptotic standard error: 0.044253
##     z-value: 7.2809, p-value: 3.3151e-13
## Wald statistic: 53.012, p-value: 3.3151e-13
##
## Log likelihood: -2460.811 for lag model
## ML residual variance (sigma squared): 39.239, (sigma: 6.2641)
## Number of observations: 754
## Number of parameters estimated: 10
## AIC: 4941.6, (AIC for lm: 4993)
## LM test for residual autocorrelation
## test value: 8.011, p-value: 0.0046494
```

```r
SAR_SSE <- sar.mg$SSE
r2_SAR <- 1 - (SAR_SSE/SST)
r2_SAR
```

```
## [1] 0.3720061
```

Pergunta 8 (bônus)

Promova o modelo final linear da Pergunta 6 a um modelo GWR. Apresente os resultados comparados (equação, R2). Qual modelo você escolheria como final? Se desejar, apresente mapas que sustente sua justificativa. Destaque a estratégia utilizada para a construção do kernel (fixo ou adaptativo, vizinhança, etc).

```r
bwGauss = gwr.sel(
  lm.step,
  data=mg,
  adapt=TRUE,
  method="aic",
  gweight=gwr.Gauss,
  verbose=FALSE
  )

gwr.ap = gwr(
  lm.step,
  data = mg,
  bandwidth=bwGauss,
  gweight=gwr.Gauss,
  adapt=bwGauss,
  hatmatrix=TRUE
  )

gwr.ap
```

```
## Call:
## gwr(formula = lm.step, data = mg, bandwidth = bwGauss, gweight = gwr.Gauss,
##     adapt = bwGauss, hatmatrix = TRUE)
## Kernel function: gwr.Gauss
## Adaptive quantile: 0.05834981 (about 43 of 754 data points)
## Summary of GWR coefficient estimates at data points:
##                     Min.     1st Qu.      Median     3rd Qu.        Max.
## X.Intercept.  4.1747e+00  9.0415e+00  1.1046e+01  1.7478e+01  2.9082e+01
## AREA         -7.7581e-03  8.2386e-05  7.0947e-04  2.0481e-03  3.9313e-03
## GINI_91      -4.0172e+01 -2.1926e+01 -1.2112e+01 -9.0403e+00  5.8026e+00
## POP_RUR      -1.4911e-03  2.1936e-04  5.0289e-04  7.7563e-04  1.4388e-03
## POP_URB      -1.7487e-03  1.5991e-04  5.2723e-04  8.0487e-04  1.8017e-03
## POP_FEM      -3.3876e-03 -1.5177e-03 -9.7372e-04 -2.8520e-04  3.4374e-03
## URBLEVEL      9.2124e+00  1.2997e+01  1.6107e+01  2.0553e+01  2.8293e+01
## PIB_PC       -3.9891e-04 -4.8758e-05  4.6096e-05  1.5263e-04  5.2884e-04
##                 Global
## X.Intercept.  11.3210
## AREA           0.0005
## GINI_91      -12.5117
## POP_RUR        0.0007
## POP_URB        0.0008
## POP_FEM       -0.0015
## URBLEVEL      18.3134
## PIB_PC         0.0002
## Number of data points: 754
## Effective number of parameters (residual: 2traceS - traceS'S): 76.10872
## Effective degrees of freedom (residual: 2traceS - traceS'S): 677.8913
## Sigma (residual: 2traceS - traceS'S): 6.117164
## Effective number of parameters (model: traceS): 54.68954
## Effective degrees of freedom (model: traceS): 699.3105
## Sigma (model: traceS): 6.022754
## Sigma (ML): 5.80022
## AICc (GWR p. 61, eq 2.33; p. 96, eq. 4.21): 4911.1
## AIC (GWR p. 96, eq. 4.22): 4845.356
## Residual sum of squares: 25366.48
## Quasi-global R2: 0.4615781
```

```r
GWR_SSE <- gwr.ap$results$rss
r2_GWR <- 1 - (GWR_SSE/SST)
r2_GWR
```

```
## [1] 0.4615781
```