GEOG 413 Unit 5

Aaron Goodman

2025-03-19

Libraries

```r
library(tmap)
library(sf)
```

```
## Linking to GEOS 3.12.2, GDAL 3.9.3, PROJ 9.4.1; sf_use_s2() is TRUE
```

```r
library(sp)
```

```
## Warning: package 'sp' was built under R version 4.4.3
```

```r
library(spdep)
```

```
## Warning: package 'spdep' was built under R version 4.4.3
```

```
## Loading required package: spData
```

```
## Warning: package 'spData' was built under R version 4.4.3
```

```
## To access larger datasets in this package, install the spDataLarge
## package with: `install.packages('spDataLarge',
## repos='https://nowosad.github.io/drat/', type='source')`
```

```r
library(spgwr)
```

```
## Warning: package 'spgwr' was built under R version 4.4.3
```

```
## NOTE: This package does not constitute approval of GWR
## as a method of spatial analysis; see example(gwr)
```

```r
library(spatialreg)
```

```
## Warning: package 'spatialreg' was built under R version 4.4.3
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: 'spatialreg'

## The following objects are masked from 'package:spdep':
##
##     get.ClusterOption, get.coresOption, get.mcOption,
##     get.VerboseOption, get.ZeroPolicyOption, set.ClusterOption,
##     set.coresOption, set.mcOption, set.VerboseOption,
##     set.ZeroPolicyOption
```

[1]. The zipped file USCovid contains county level data on COVID deaths per 100,000 of the population registered over the first year of the pandemic (roughly March 2020 – February 2021). The data in this file have all been normalized. Here are the meanings of the variables of interest:

```
population = county population in 2020
cases_per_ = Covid death rate (normalized)
pct_poc = Persons of color as share of total population (normalized)
pct_smoker = Share of population that smoke (normalized)
pct_povert = Share of population in poverty (normalized)
pct_obese = Share of population defined as obese (normalized)
pct_65plus = Share of elderly population (normalized)
per_dem = Share of Democratic registered voters (normalized)
```

```r
uscovid <- st_read("../data/USCovid/totals_apr20_feb21.shp")
```

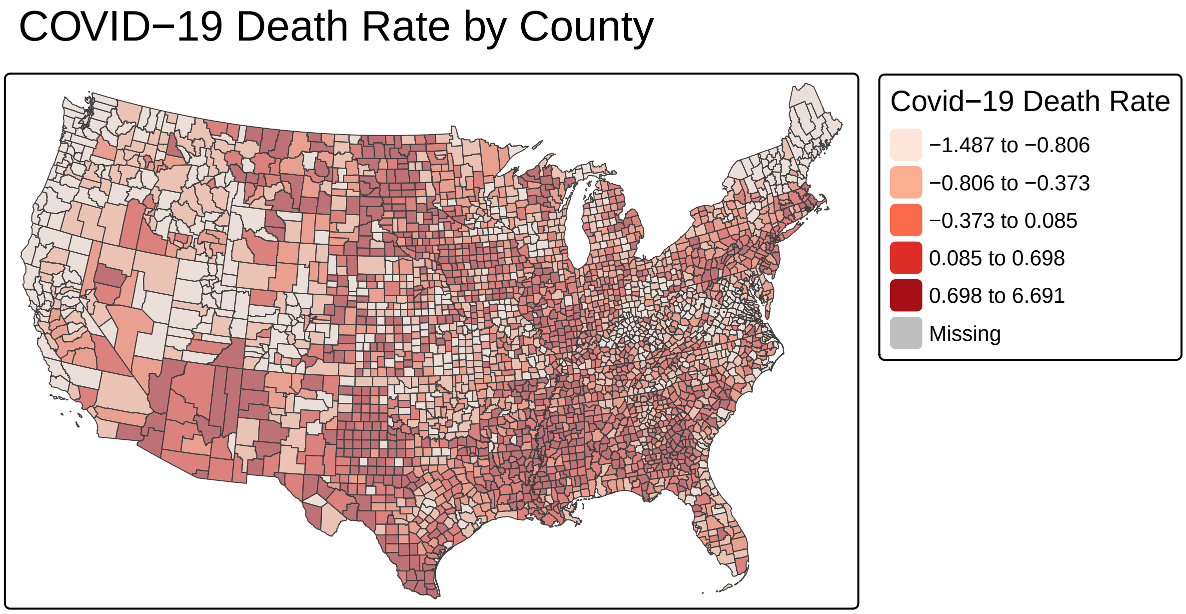
```
## Reading layer `totals_apr20_feb21' from data source
##   `F:\MAGIST\23_25w_GEOG_413_AppliedGeospatialStats\U5_Spatial-Regression\data\USCovid\totals_apr20_
##   using driver `ESRI Shapefile'
## Simple feature collection with 3108 features and 15 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -2031905 ymin: -2116976 xmax: 2516374 ymax: 732352.2
## Projected CRS: US National Atlas Equal Area
```

[a]. Generate a quantile map of the Covid death rate and report what you see.

```r
tm_shape(uscovid) +
  tm_fill("cases_per_",
          style="quantile",
          n=5,
          palette="Reds",
          title="Covid-19 Death Rate") +
  tm_borders(alpha=0.5, lwd=0.5) +
  tm_layout(main.title="COVID-19 Death Rate by County",
            main.title.position="center",
            legend.outside=TRUE)
```

```
##
```

```
## -- tmap v3 code detected -------------------------------------------------------

## [v3->v4] `tm_fill()`: instead of `style = "quantile"`, use fill.scale =
## `tm_scale_intervals()`.
## i Migrate the argument(s) 'style', 'n', 'palette' (rename to 'values') to
##   'tm_scale_intervals(<HERE>)'
## [v3->v4] `tm_fill()`: migrate the argument(s) related to the legend of the
## visual variable `fill` namely 'title' to 'fill.legend = tm_legend(<HERE>)'
## [v3->v4] `tm_borders()`: use 'fill' for the fill color of polygons/symbols
## (instead of 'col'), and 'col' for the outlines (instead of 'border.col').
## [v3->v4] `tm_borders()`: use `fill_alpha` instead of `alpha`.
## [v3->v4] `tm_layout()`: use `tm_title()` instead of `tm_layout(main.title = )`
## [cols4all] color palettes: use palettes from the R package cols4all. Run
## `cols4all::c4a_gui()` to explore them. The old palette name "Reds" is named
## "brewer.reds"
## Multiple palettes called "reds" found: "brewer.reds", "matplotlib.reds". The first one, "brewer.reds
```



Interpretation The quantile map shows the distribution of COVID-19 death rates across U.S. counties from March 2020 to February 2021. Considerable clusters of higher death rates can be observed in urban centers around the country, though there are still examples of higher death rates in rural areas.

[b]. Is there evidence of spatial autocorrelation in the Covid death rate data? (Use a reasonable spatial weights scheme.)

```
uscovid_nb <- poly2nb(uscovid, queen=TRUE)
```

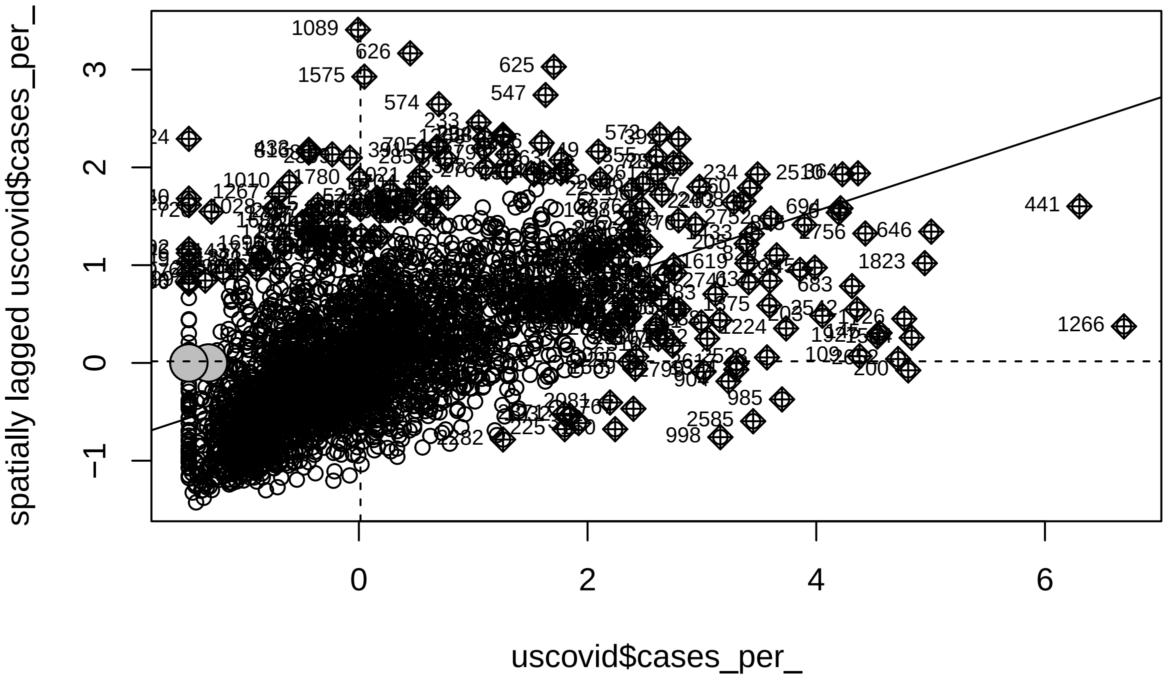
```
## Warning in poly2nb(uscovid, queen = TRUE): some observations have no neighbours;
## if this seems unexpected, try increasing the snap argument.
```

```
## Warning in poly2nb(uscovid, queen = TRUE): neighbour object has 3 sub-graphs;
## if this sub-graph count seems unexpected, try increasing the snap argument.
```

```
uscovid_weights <- nb2listw(uscovid_nb, style="W", zero.policy=TRUE)

uscovid_moranTest <-moran.test(uscovid$cases_per_, uscovid_weights, zero.policy=TRUE)
uscovid_moranTest
```

```
##
##  Moran I test under randomisation
##
## data:  uscovid$cases_per_
## weights: uscovid_weights
## n reduced by no-neighbour observations
##
## Moran I statistic standard deviate = 35.882, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.3856730726     -0.0003220612      0.0001157177
```

```
moran.plot(uscovid$cases_per_, uscovid_weights, zero.policy=TRUE)
```



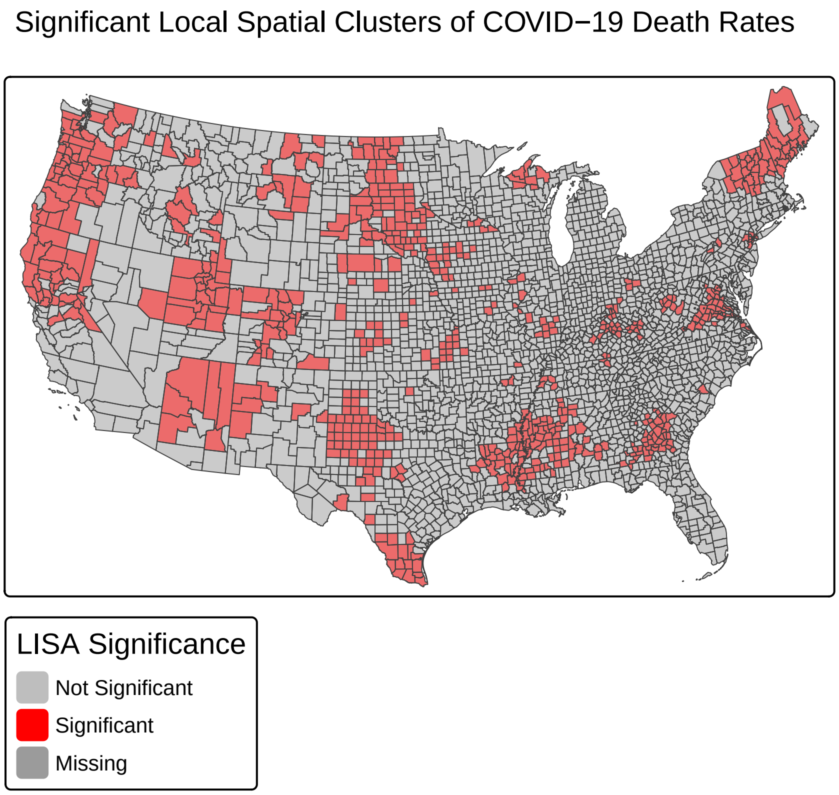
Interpretation The Moran's I statistic of 0.386 with p-value of 2.2e-16, which is far below typical significance threshold p < 0.05, indicates considerable positive spatial autocorrelation. This means similar values cluster together, i.e., counties with lower death rates have neighbors with lower death rates, and counties with higher death rates have neighbors with higher death rates.

[c]. Is there evidence of significant local spatial clusters of Covid death rates?

```r
uscovid_lisa <- localmoran(uscovid$cases_per_, uscovid_weights, zero.policy=TRUE)

uscovid$lisa_i <- uscovid_lisa[,1]
uscovid$lisa_p <- uscovid_lisa[,5]

uscovid$lisa_sig <- ifelse(uscovid$lisa_p < 0.05, "Significant", "Not Significant")

tm_shape(uscovid) +
  tm_fill("lisa_sig",
          palette=c("gray","red"),
          title="LISA Significance") +
  tm_borders(alpha=0.5, lwd=0.5) +
  tm_layout(main.title="Significant Local Spatial Clusters of COVID-19 Death Rates",
            main.title.position="center",
            legend.outside=TRUE)
```

```
##
```

```
## -- tmap v3 code detected -----------------------------------------------------

## [v3->v4] `tm_tm_fill()`: migrate the argument(s) related to the scale of the
## visual variable `fill` namely 'palette' (rename to 'values') to fill.scale =
## tm_scale(<HERE>).
## [v3->v4] `tm_fill()`: migrate the argument(s) related to the legend of the
## visual variable `fill` namely 'title' to 'fill.legend = tm_legend(<HERE>)'
## [v3->v4] `tm_borders()`: use `fill_alpha` instead of `alpha`.
## [v3->v4] `tm_layout()`: use `tm_title()` instead of `tm_layout(main.title = )`
## Multiple palettes called "gray" found: "matplotlib.gray", "tableau.gray". The first one, "matplotlib
##
## [plot mode] fit legend/component: Some legend items or map compoments do not
## fit well, and are therefore rescaled.
## i Set the tmap option `component.autoscale = FALSE` to disable rescaling.
```


Significant Local Spatial Clusters of COVID−19 Death Rates

Interpretation The map of local spatial clusters of Covid-19 death rates for p $<$ 0.05 shows several metropolitan regions around the country with statistically significant clustering.

[d]. Regress the Covid death rate against the six independent variables given above. Show and interpret your regression output (partial regression coefficients, goodness of fit statistics).

```r
uscovid_model <- lm(cases_per_ ~ pct_poc + pct_smoker + pct_povert +
                      pct_obsese + pct_65plus + per_dem, data=uscovid)
summary(uscovid_model)
```

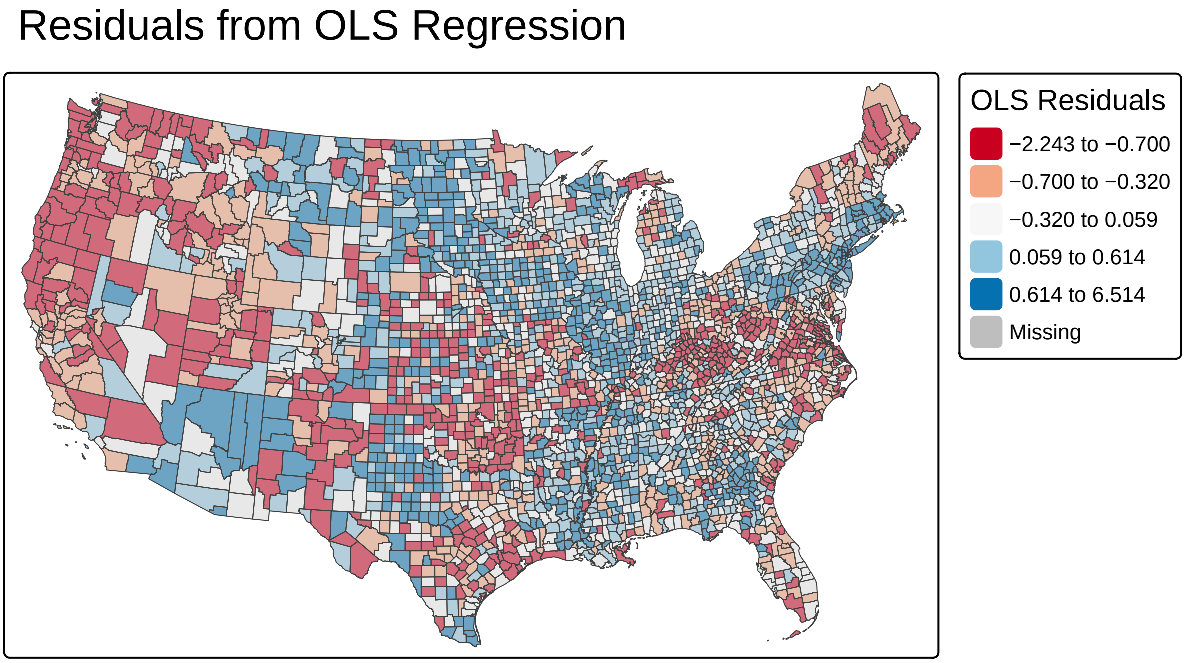
```
##
## Call:
## lm(formula = cases_per_ ~ pct_poc + pct_smoker + pct_povert +
##     pct_obsese + pct_65plus + per_dem, data = uscovid)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.2429 -0.5982 -0.1357  0.4312  6.5142
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.01362    0.01656   0.822    0.411
## pct_poc      0.28321    0.02737  10.347  < 2e-16 ***
## pct_smoker  -0.01943    0.02273  -0.855    0.393
## pct_povert   0.11754    0.01991   5.904 3.92e-09 ***
## pct_obsese   0.14833    0.01955   7.586 4.34e-14 ***
## pct_65plus   0.12800    0.01964   6.518 8.27e-11 ***
## per_dem     -0.16805    0.02148  -7.824 6.97e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9227 on 3101 degrees of freedom
## Multiple R-squared:  0.1432, Adjusted R-squared:  0.1415
## F-statistic: 86.38 on 6 and 3101 DF,  p-value: < 2.2e-16
```

Interpretation Multiple R-squared = 0.1432 and Adjusted R-squared = 0.1415 indicate that the six independent variables together explain about 14% of the variation in COVID-19 death rates. Variables $pct_smoker and $per_dem have negative regression coefficients, suggesting that counties with more smoking residents or Democratic voters experienced lower COVID-19 death rates. The other variables have positive regression coefficients, suggesting that counties with more people of color, residents living under poverty, obese residents, and senior residents experience higher COVID-19 death rates. The regression coefficient with the highest magnitude is that of $pct_poc, pointing to the strength of this variable as a predictor for death rates. The very small p-value (less than 2.2e-16) indicates that the model is statistically significant.

[e]. Capture and plot the residuals from your regression. Is there evidence of spatial autocorrelation in the residuals?

```r
uscovid$residuals <- residuals(uscovid_model)

tm_shape(uscovid) +
  tm_fill("residuals",
          style="quantile",
          n=5,
          palette="RdBu",
          title="OLS Residuals") +
  tm_borders(alpha=0.5, lwd=0.5) +
  tm_layout(main.title="Residuals from OLS Regression",
            main.title.position="center",
            legend.outside=TRUE)
```

```
##

## -- tmap v3 code detected ----------------------------------------------------

## [v3->v4] `tm_fill()`: instead of `style = "quantile"`, use fill.scale =
## `tm_scale_intervals()`.
## i Migrate the argument(s) 'style', 'n', 'palette' (rename to 'values') to
##   'tm_scale_intervals(<HERE>)'
## [v3->v4] `tm_fill()`: migrate the argument(s) related to the legend of the
## visual variable `fill` namely 'title' to 'fill.legend = tm_legend(<HERE>)'
## [v3->v4] `tm_borders()`: use `fill_alpha` instead of `alpha`.
## [v3->v4] `tm_layout()`: use `tm_title()` instead of `tm_layout(main.title = )`
## [cols4all] color palettes: use palettes from the R package cols4all. Run
## `cols4all::c4a_gui()` to explore them. The old palette name "RdBu" is named
## "brewer.rd_bu"
## Multiple palettes called "rd_bu" found: "brewer.rd_bu", "matplotlib.rd_bu". The first one, "brewer.r
##
## [scale] tm_polygons:() the data variable assigned to 'fill' contains positive and negative values, s
```

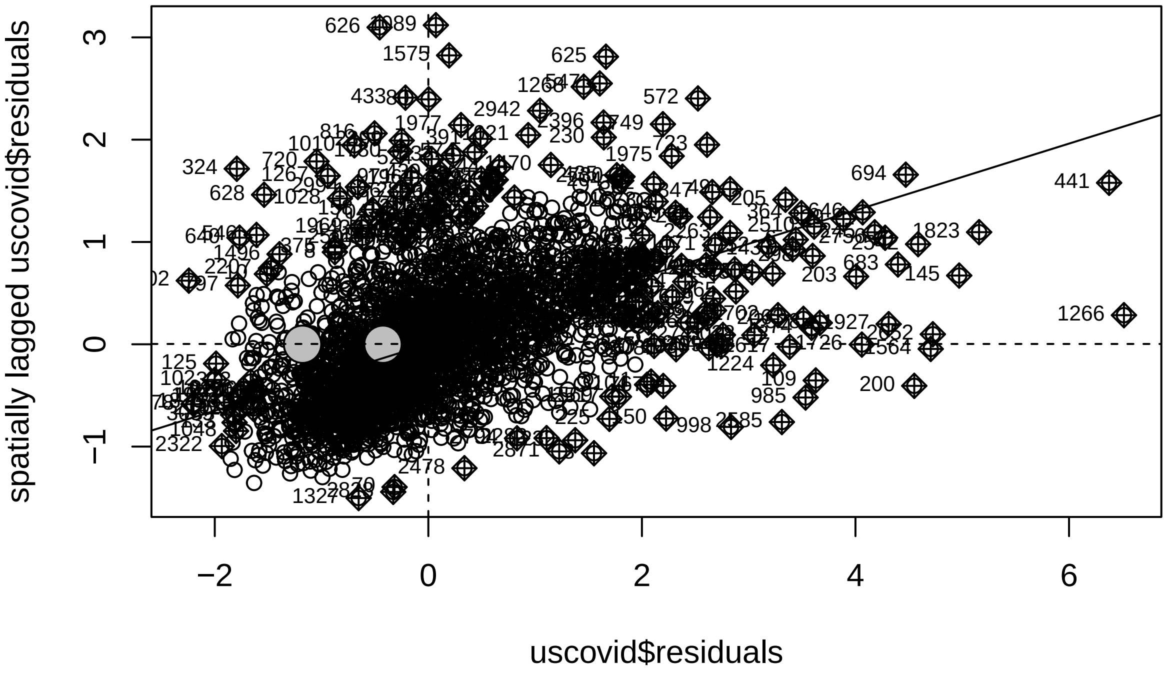


```r
uscovid_moran_residuals <- moran.test(uscovid$residuals, uscovid_weights, zero.policy=TRUE)
uscovid_moran_residuals
```

```
##
##  Moran I test under randomisation
```

```
##
## data:  uscovid$residuals
## weights: uscovid_weights
## n reduced by no-neighbour observations
##
## Moran I statistic standard deviate = 30.378, p-value < 2.2e-16
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic       Expectation          Variance
##      0.3264183321     -0.0003220612      0.0001156865
```

```
moran.plot(uscovid$residuals, uscovid_weights, zero.policy=TRUE)
```



Interpretation A positive Moran's I statistic of 0.326 (with p-value $< 2.2e-16$) indicates that there is statistically significant spatial autocorrelation in the residuals for the COVID-19 death rate. This means that counties with higher-than-expected death rates tend to be near other counties with higher-than-expected death rates.

[f]. Run spatial lag and spatial error models. Interpret the output from these and show if one of these models fits the data better than the other.

```
# Spatial lag
uscovid_lagModel <- lagsarlm(cases_per_ ~ pct_poc + pct_smoker + pct_povert +
```

```
                              pct_obsese + pct_65plus + per_dem, data=uscovid,
                            listw=uscovid_weights,
                            zero.policy=TRUE)
summary(uscovid_lagModel)
```

```
##
## Call:lagsarlm(formula = cases_per_ ~ pct_poc + pct_smoker + pct_povert +
##     pct_obsese + pct_65plus + per_dem, data = uscovid, listw = uscovid_weights,
##     zero.policy = TRUE)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.79084 -0.46531 -0.11704  0.35582  6.39417
##
## Type: lag
## Regions with no neighbours included:
##  1771 3033
## Coefficients: (asymptotic standard errors)
##               Estimate Std. Error z value  Pr(>|z|)
## (Intercept)  0.0054687  0.0145921  0.3748    0.7078
## pct_poc      0.1809474  0.0247723  7.3044 2.784e-13
## pct_smoker   0.0122800  0.0200300  0.6131    0.5398
## pct_povert   0.0883911  0.0176683  5.0028 5.650e-07
## pct_obsese   0.0732371  0.0173724  4.2157 2.490e-05
## pct_65plus   0.0932635  0.0174022  5.3593 8.355e-08
## per_dem     -0.0900365  0.0192713 -4.6720 2.982e-06
##
## Rho: 0.52385, LR test value: 605.5, p-value: < 2.22e-16
## Asymptotic standard error: 0.020435
##     z-value: 25.635, p-value: < 2.22e-16
## Wald statistic: 657.17, p-value: < 2.22e-16
##
## Log likelihood: -3853.77 for lag model
## ML residual variance (sigma squared): 0.66053, (sigma: 0.81273)
## Number of observations: 3108
## Number of parameters estimated: 9
## AIC: 7725.5, (AIC for lm: 8329)
## LM test for residual autocorrelation
## test value: 23.506, p-value: 1.2452e-06
```

```
# Spatial error
uscovid_errorModel <- errorsarlm(cases_per_ ~ pct_poc + pct_smoker + pct_povert +
                              pct_obsese + pct_65plus + per_dem, data=uscovid,
                            listw=uscovid_weights,
                            zero.policy=TRUE)
summary(uscovid_errorModel)
```

```
##
## Call:errorsarlm(formula = cases_per_ ~ pct_poc + pct_smoker + pct_povert +
##     pct_obsese + pct_65plus + per_dem, data = uscovid, listw = uscovid_weights,
##     zero.policy = TRUE)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -2.68261 -0.46172 -0.11085  0.33950  6.36561
##
## Type: error
## Regions with no neighbours included:
##  1771 3033
## Coefficients: (asymptotic standard errors)
##               Estimate Std. Error z value  Pr(>|z|)
## (Intercept)  0.0100684  0.0318838  0.3158  0.752168
## pct_poc      0.3150291  0.0369604  8.5234 < 2.2e-16
## pct_smoker   0.0019546  0.0288556  0.0677  0.945994
## pct_povert   0.1340235  0.0218652  6.1295 8.814e-10
## pct_obsese   0.0866651  0.0276598  3.1333  0.001729
## pct_65plus   0.1326076  0.0212619  6.2369 4.464e-10
## per_dem     -0.1779322  0.0277911 -6.4025 1.529e-10
##
## Lambda: 0.5462, LR test value: 631.3, p-value: < 2.22e-16
## Asymptotic standard error: 0.020724
##     z-value: 26.356, p-value: < 2.22e-16
## Wald statistic: 694.65, p-value: < 2.22e-16
##
## Log likelihood: -3840.868 for error model
## ML residual variance (sigma squared): 0.65135, (sigma: 0.80706)
## Number of observations: 3108
## Number of parameters estimated: 9
## AIC: 7699.7, (AIC for lm: 8329)
```

```r
# Compare
AIC(uscovid_model, uscovid_lagModel, uscovid_errorModel)
```

```
##                    df      AIC
## uscovid_model       8 8329.040
## uscovid_lagModel    9 7725.541
## uscovid_errorModel  9 7699.737
```

Interpretation Spatial Lag produces a rho or spatial parameter of 0.52385, while Spatial Error produces a more significant lambda or spatial parameter of 0.5462. Comparing the AIC values for the regular regression, Spatial Lag, and Spatial Error models shows that Spatial Error produced the lowest AIC. By these metrics, Spatial Error better fits the data.

Considering the AIC values of the regular regression, Spatial Lag, and Spatial Error models for COVID-19 death rates, Spatial Error

[2]. Use the zipped stlouis.shp file and the packages sf, sp, spdep, spgwr & tmap and any others you might need for spatial analysis and mapping.

```r
stlouis <- st_read("../data/stlouis/stlouis.shp")
```

```
## Reading layer `stlouis' from data source
##   `F:\MAGIST\23_25w_GEOG_413_AppliedGeospatialStats\U5_Spatial-Regression\data\stlouis\stlouis.shp'
##   using driver `ESRI Shapefile'
```

```
## Simple feature collection with 78 features and 23 fields
## Geometry type: POLYGON
## Dimension:     XY
## Bounding box:  xmin: -92.70068 ymin: 36.88181 xmax: -87.91657 ymax: 40.32957
## Geodetic CRS:  WGS 84
```

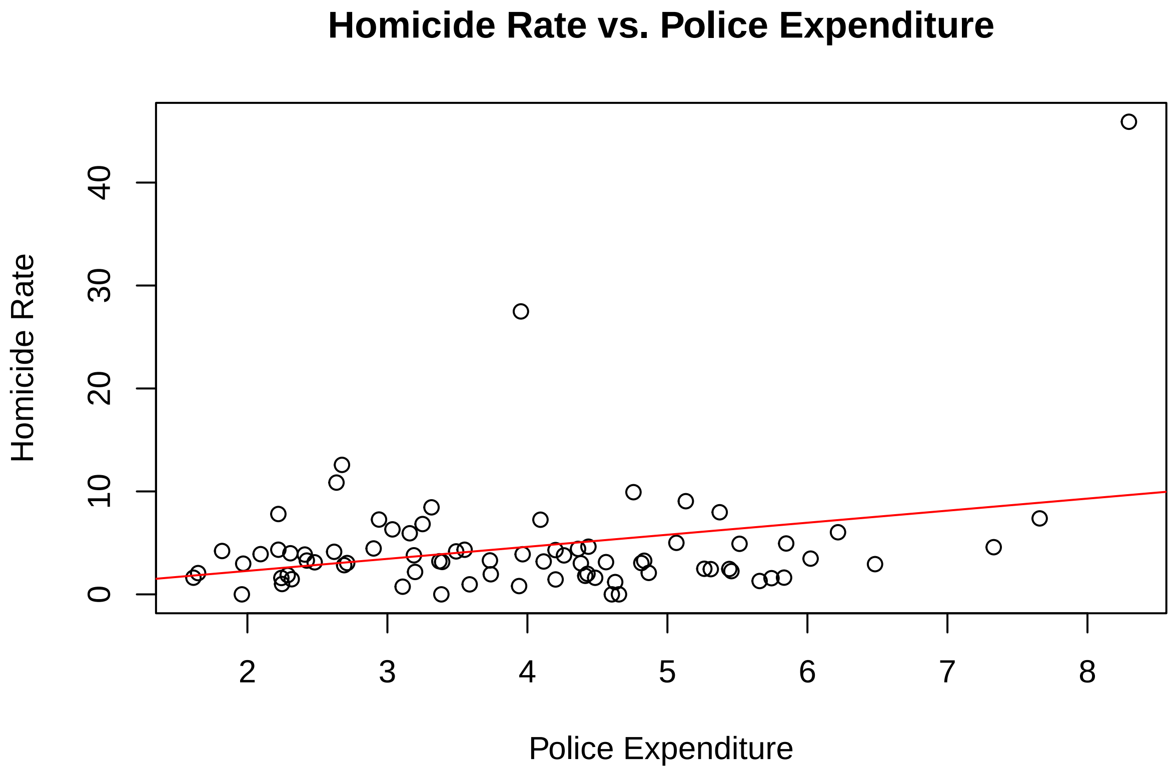
[a]. **Run a non-spatial regression of the homicide rate (HR8893) on police expenditure (PE87) and a local deprivation index (RDAC90) and interpret the output.**

```r
homicide_model <- lm(HR8893 ~ PE87 + RDAC90, data=stlouis)
summary(homicide_model)
```

```
##
## Call:
## lm(formula = HR8893 ~ PE87 + RDAC90, data = stlouis)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.8749 -2.7648 -0.6719  2.1715 20.2329
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.03748    1.54263   0.024    0.981
## PE87         1.56705    0.37128   4.221 6.75e-05 ***
## RDAC90       5.29091    0.82492   6.414 1.14e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.692 on 75 degrees of freedom
## Multiple R-squared:  0.4065, Adjusted R-squared:  0.3906
## F-statistic: 25.68 on 2 and 75 DF,  p-value: 3.194e-09
```

```r
plot(stlouis$RDAC90, stlouis$HR8893,
     main="Homicide Rate vs. Deprivation Index",
     xlab="Deprivation Index",
     ylab="Homicide Rate")
abline(lm(HR8893 ~ RDAC90, data=stlouis), col="red")
```

```r
plot(stlouis$PE87, stlouis$HR8893,
     main="Homicide Rate vs. Police Expenditure",
     xlab="Police Expenditure",
     ylab="Homicide Rate")
abline(lm(HR8893 ~ PE87, data=stlouis), col="red")
```



Interpretation Multiple R-squared = 0.4065 and Adjusted R-squared = 0.3906 indicate that the Deprivation Index and Police Expenditure account for about 40% of the variation in homicide rates. Variable $PE87 or Police Expenditure has a positive regression coefficient of 1.56705, while variable $RDAC90 has a larger positive regression coefficient of 5.29091. This suggests that regions with higher police expenditure and Deprivation Indices tend to have higher homicide rates. The greater magnitude of the Deprivation Index's regression coefficient indicates that variable is a stronger predictor for homicide rates. The very small p-value of 3.194e-09 indicates that the model is statistically significant.

[b]. Use the R package spgwr to show how the R2 and the partial regression coefficients in your model vary across your study region. Report and interpret all your output.

```
stlouis_sp <- as(stlouis, "Spatial")

gwr_bw <- gwr.sel(HR8893 ~ PE87 + RDAC90,
                  data=stlouis_sp,
                  adapt=TRUE)
```

```
## Adaptive q: 0.381966 CV score: 2462.794
## Adaptive q: 0.618034 CV score: 2493.324
## Adaptive q: 0.236068 CV score: 2309.391
## Adaptive q: 0.145898 CV score: 2094.679
## Adaptive q: 0.09016994 CV score: 1394.082
## Adaptive q: 0.05572809 CV score: 1032.717
```

```
## Adaptive q: 0.03444185 CV score: 903.0778
## Adaptive q: 0.02128624 CV score: 840.6013
## Adaptive q: 0.01315562 CV score: 843.9937
## Adaptive q: 0.01808047 CV score: 837.933
## Adaptive q: 0.01803978 CV score: 837.9085
## Adaptive q: 0.0161742 CV score: 837.8115
## Adaptive q: 0.01701687 CV score: 837.5713
## Adaptive q: 0.01705756 CV score: 837.5728
## Adaptive q: 0.01697618 CV score: 837.5709
## Adaptive q: 0.01666985 CV score: 837.6052
## Adaptive q: 0.01693549 CV score: 837.5716
## Adaptive q: 0.01697618 CV score: 837.5709
```

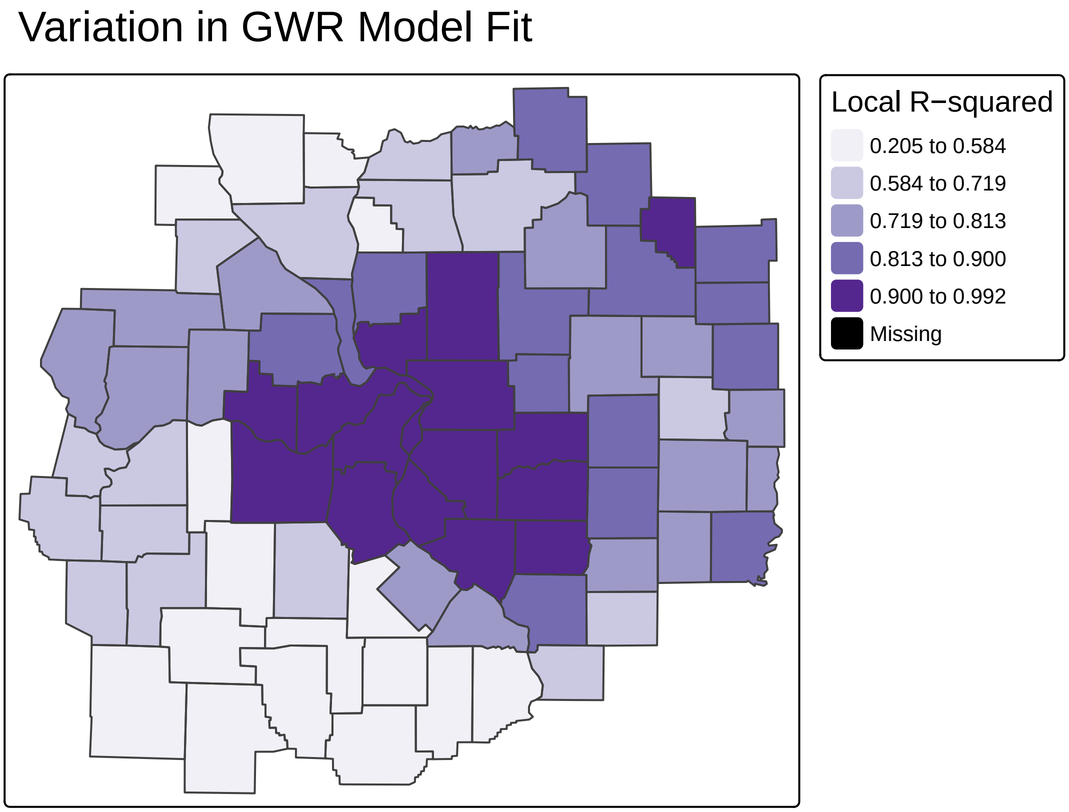
```
gwr_model <- gwr(HR8893 ~ PE87 + RDAC90,
                 data=stlouis_sp,
                 adapt=gwr_bw,
                 hatmatrix=TRUE,
                 se.fit=TRUE)

gwr_model
```

```
## Call:
## gwr(formula = HR8893 ~ PE87 + RDAC90, data = stlouis_sp, adapt = gwr_bw,
##     hatmatrix = TRUE, se.fit = TRUE)
## Kernel function: gwr.Gauss
## Adaptive quantile: 0.01697618 (about 1 of 78 data points)
## Summary of GWR coefficient estimates at data points:
##                   Min.  1st Qu.   Median  3rd Qu.     Max. Global
## X.Intercept. -1.86007  1.34844  3.38473  5.33793 18.08941 0.0375
## PE87         -3.31961 -0.18401  0.44314  1.00753  2.69020 1.5670
## RDAC90       -9.17972  0.57759  1.87845  6.50702 11.94509 5.2909
## Number of data points: 78
## Effective number of parameters (residual: 2traceS - traceS'S): 47.66878
## Effective degrees of freedom (residual: 2traceS - traceS'S): 30.33122
## Sigma (residual: 2traceS - traceS'S): 2.672647
## Effective number of parameters (model: traceS): 36.96118
## Effective degrees of freedom (model: traceS): 41.03882
## Sigma (model: traceS): 2.297678
## Sigma (ML): 1.66663
## AICc (GWR p. 61, eq 2.33; p. 96, eq. 4.21): 452.7336
## AIC (GWR p. 96, eq. 4.22): 338.001
## Residual sum of squares: 216.6572
## Quasi-global R2: 0.9221223
```

```
gwr_results <- as.data.frame(gwr_model$SDF)

stlouis_gwr <- st_as_sf(stlouis_sp)
stlouis_gwr <- cbind(stlouis_gwr, gwr_results)

# Local R-squared
tm_shape(stlouis_gwr) +
  tm_fill("localR2",
          style="quantile",
```

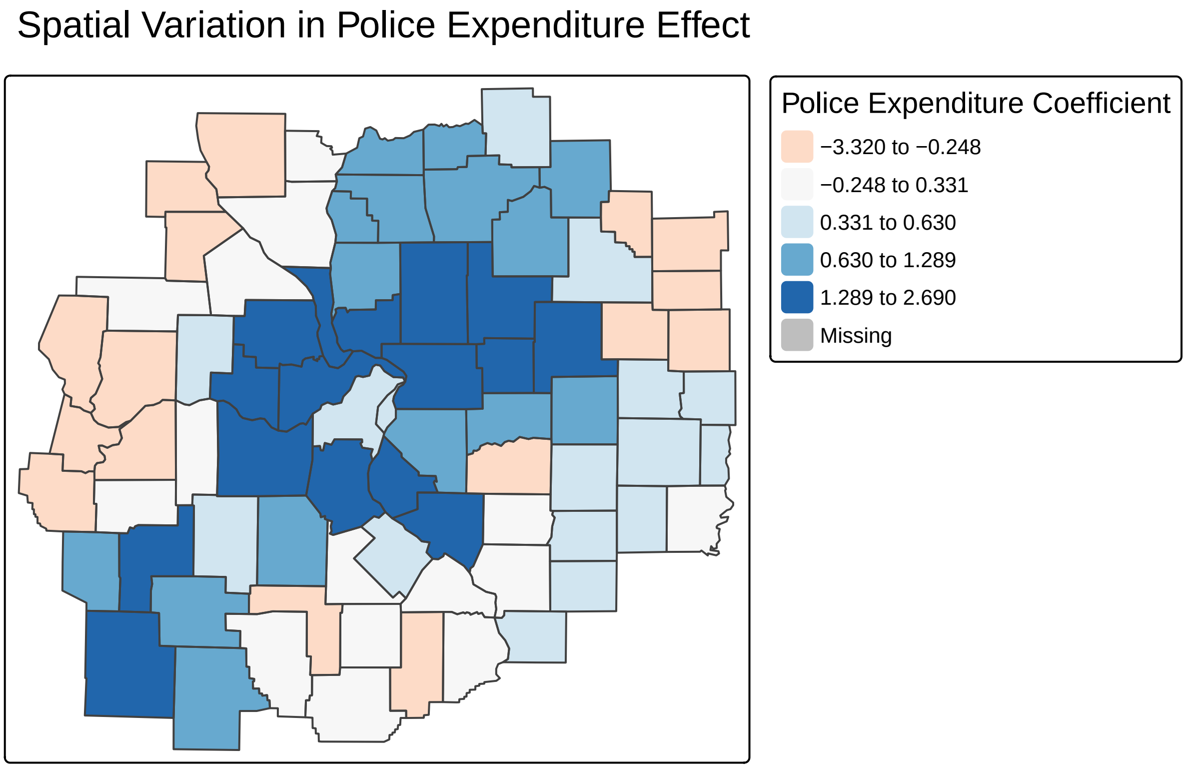
```
          n=5,
          palette="Purples",
          title="Local R-squared") +
  tm_borders() +
  tm_layout(main.title="Variation in GWR Model Fit",
            main.title.position="center",
            legend.outside=TRUE)
```

```
##

## -- tmap v3 code detected -------------------------------------------------------

## [v3->v4] `tm_fill()`: instead of `style = "quantile"`, use fill.scale =
## `tm_scale_intervals()`.
## i Migrate the argument(s) 'style', 'n', 'palette' (rename to 'values') to
##   'tm_scale_intervals(<HERE>)'
## [v3->v4] `tm_fill()`: migrate the argument(s) related to the legend of the
## visual variable `fill` namely 'title' to 'fill.legend = tm_legend(<HERE>)'
## [v3->v4] `tm_layout()`: use `tm_title()` instead of `tm_layout(main.title = )`
## [cols4all] color palettes: use palettes from the R package cols4all. Run
## `cols4all::c4a_gui()` to explore them. The old palette name "Purples" is named
## "brewer.purples"
## Multiple palettes called "purples" found: "brewer.purples", "matplotlib.purples". The first one, "br
```



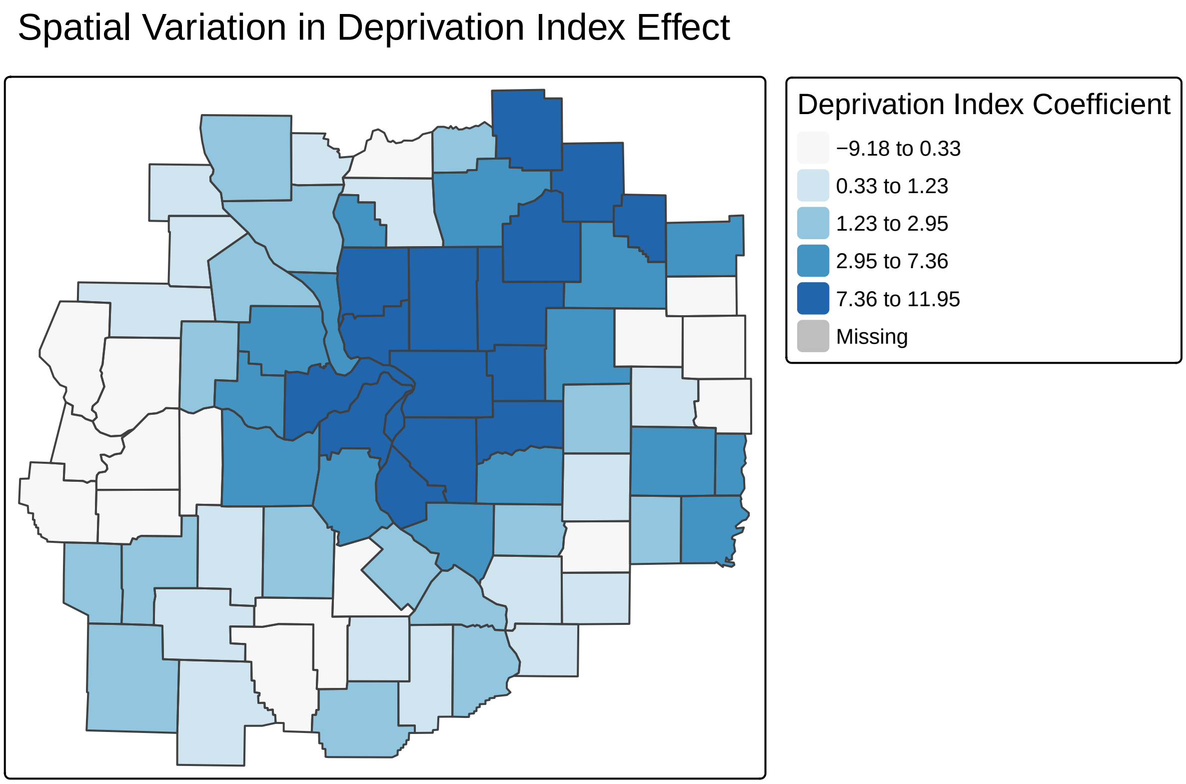
```r
# police expenditure coefficients
tm_shape(stlouis_gwr) +
  tm_fill("PE87.1",
          style="quantile",
          n=5,
          palette="RdBu",
          title="Police Expenditure Coefficient") +
  tm_borders() +
  tm_layout(main.title="Spatial Variation in Police Expenditure Effect",
            main.title.position="center",
            legend.outside=TRUE)
```

```
##
## -- tmap v3 code detected -------------------------------------------------------
## [v3->v4] `tm_fill()`: instead of `style = "quantile"`, use fill.scale =
## `tm_scale_intervals()`.
## i Migrate the argument(s) 'style', 'n', 'palette' (rename to 'values') to
##   'tm_scale_intervals(<HERE>)'[v3->v4] `tm_fill()`: migrate the argument(s) related to the legend of
## visual variable `fill` namely 'title' to 'fill.legend = tm_legend(<HERE>)'[v3->v4] `tm_layout()`: us
## `cols4all::c4a_gui()` to explore them. The old palette name "RdBu" is named
## "brewer.rd_bu"Multiple palettes called "rd_bu" found: "brewer.rd_bu", "matplotlib.rd_bu". The first
## [scale] tm_polygons:() the data variable assigned to 'fill' contains positive and negative values, s
## [plot mode] fit legend/component: Some legend items or map compoments do not
## fit well, and are therefore rescaled.
## i Set the tmap option `component.autoscale = FALSE` to disable rescaling.
```



```r
# deprivation index coefficients
tm_shape(stlouis_gwr) +
  tm_fill("RDAC90.1",
          style="quantile",
          n=5,
          palette="RdBu",
          title="Deprivation Index Coefficient") +
  tm_borders() +
  tm_layout(main.title="Spatial Variation in Deprivation Index Effect",
            main.title.position="center",
            legend.outside=TRUE)
```

```
##
## -- tmap v3 code detected -------------------------------------------------------
## [v3->v4] `tm_fill()`: instead of `style = "quantile"`, use fill.scale =
## `tm_scale_intervals()`.
## i Migrate the argument(s) 'style', 'n', 'palette' (rename to 'values') to
##   'tm_scale_intervals(<HERE>)'[v3->v4] `tm_fill()`: migrate the argument(s) related to the legend of
## visual variable `fill` namely 'title' to 'fill.legend = tm_legend(<HERE>)'[v3->v4] `tm_layout()`: us
## `cols4all::c4a_gui()` to explore them. The old palette name "RdBu" is named
## "brewer.rd_bu"Multiple palettes called "rd_bu" found: "brewer.rd_bu", "matplotlib.rd_bu". The first
## [scale] tm_polygons:() the data variable assigned to 'fill' contains positive and negative values, s
```



Interpretation The map “Variation in GWR Model Fit” plots Local R-squared onto regions in St. Louis. This means that the purple gradient of colors plotted on the map correspond to the percentage of variation in homicide rate that can be explained by police expenditure and the Deprivation Index. An unmistakable cluster of high local R-squared values in the center of the city (R-squared > 0.90) identify regions in which 90% of the variation in homicide rates is explained by police expenditure and the Deprivation Index.

The map “Spatial Variation in Police Expenditure Effect” shows the distribution of the Police Expenditure Coefficient across St. Louis. Regions with more positive values, shaded blue, exhibited a stronger relationship between police expenditure and homicide rates. Regions with more negative values, shaded lighter blue to pink, exhibited a weaker relationship between police expenditure and homicide rates. The map shows regions with higher Police Expenditure Coefficients clustered in the center of the city, with notable holes; some regions in the southwest corner of the city exhibited higher Coefficients as well.

The map “Spatial Variation in Deprivation Index Effect” shows the distribution of the Deprivation Index Coefficient across St. Louis. Regions with more positive values, shaded blue, exhibited a stronger relationship between the Deprivation Index and homicide rates. Regions with more negative values, shaded white, exhibited a weaker relationship between Deprivation Index and homicide rates. The map shows a cluster of regions with higher Deprivation Index Coefficients in the center of the city, stretching towards the northwest corner.

The magnitude of the regression coefficients for the Deprivation Index are considerably greater than those for police expenditure, again suggesting that the Deprivation Index as an independent variable is a stronger predictor for homicide rates than police expenditure.