Spatial Data Analysis in R

Dealing With Spatial Dependence 3
Examples

Eco 697DR – University of Massachusetts, Amherst – Spring 2022
Michael France Nelson

Packages and Data

You'll need the following spatial packages to recreate the analyses in this deck:

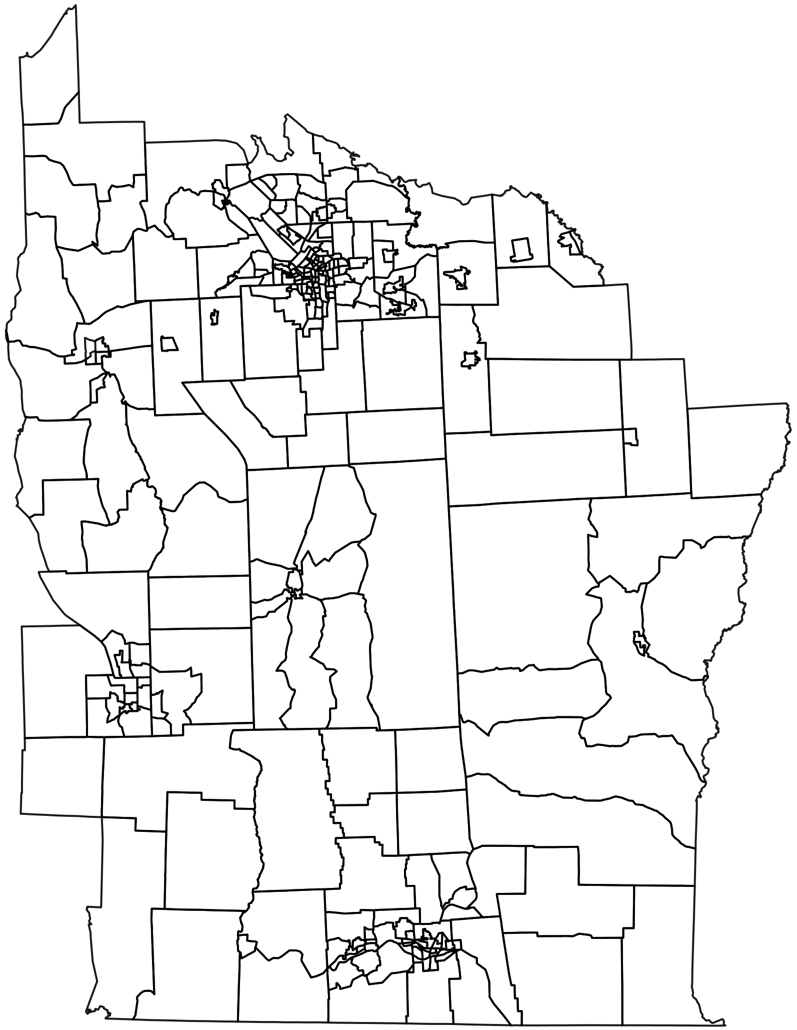
- spdep, spatialreg
- rgdal, sf, sp

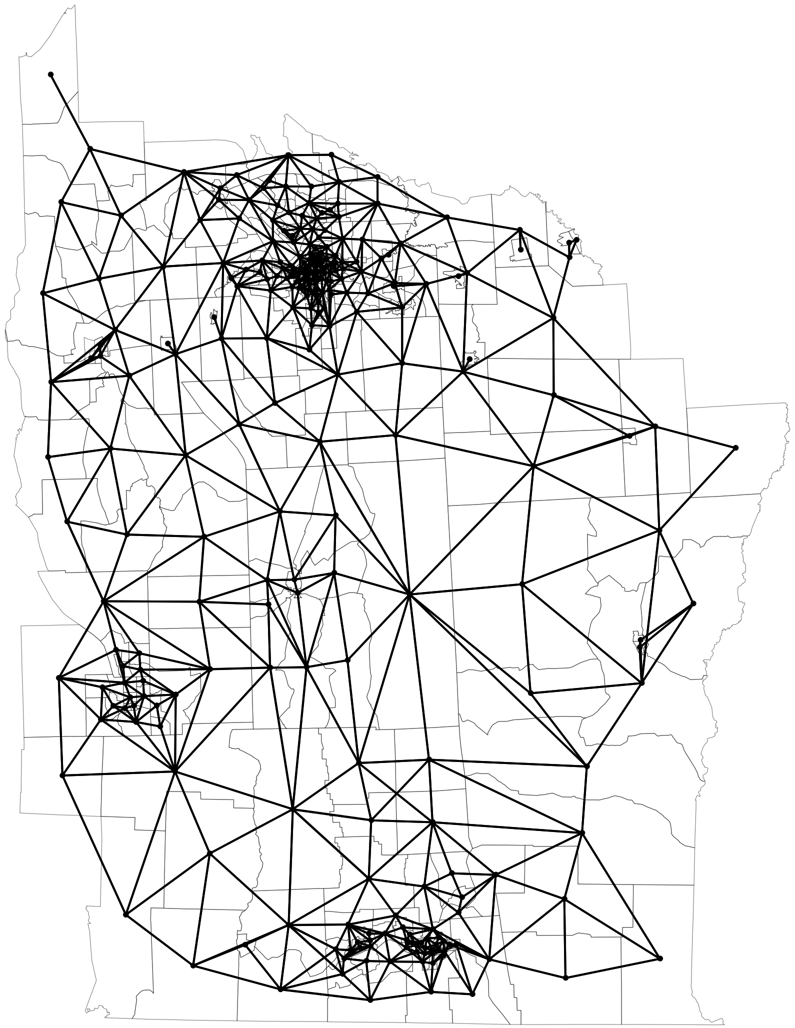
Leukemia Incidence

- Areal data: New York congressional district 8 (includes Syracuse).
- Responses:
  - Leukemia counts (Cases)
  - Leukemia incidence rate, Log-transformed (Z)
- Predictors:
  - Pct age 65 or more (PCTAGE65P)
  - TCE (pollution) exposure (PEXPOSURE)
  - Pct homeownership (PCTOWNHOME)

- Example inspired by Bivand et al. 2008 Chapter 9

New York Congressional District 8





New York Congressional District 8

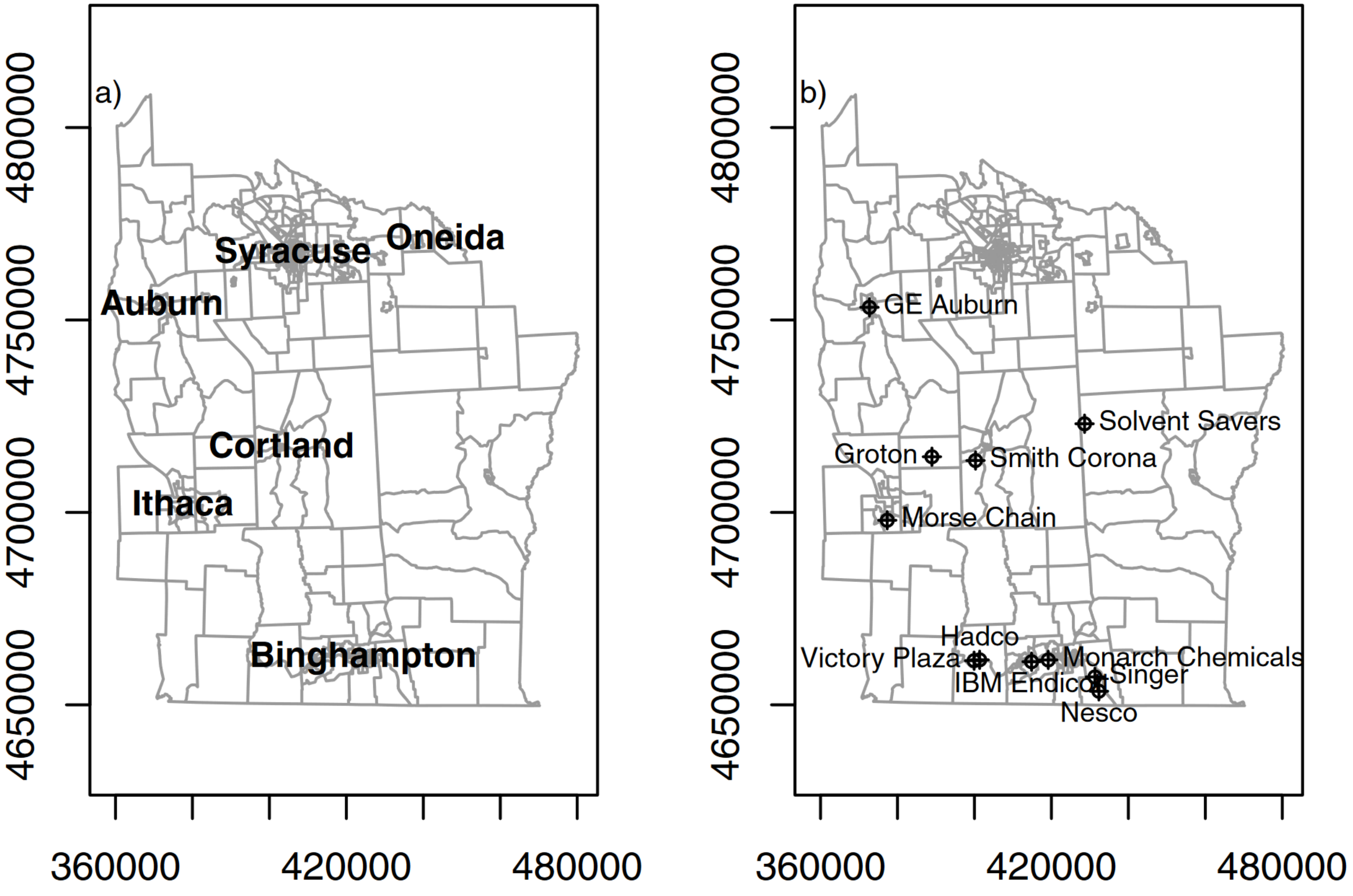


Fig. 9.1 (**a**) Major cities in the eight-county upper New York State study area; (**b**) locations of 11 inactive hazardous waste sites in the study area

We’ll Focus on Syracuse

We need to:

- Subset to Syracuse
- Create neighborhood object
- Create neighborhood weights

```
sy_sp = subset(ny_8, AREANAME == "Syracuse city")
sy_nb = poly2nb(sy_sp, queen = TRUE)
sy_nb_w = nb2listw(sy_nb)
```

ny_8 is a SPolygonsDF

Plot the Census Tracts and Neighborhoods

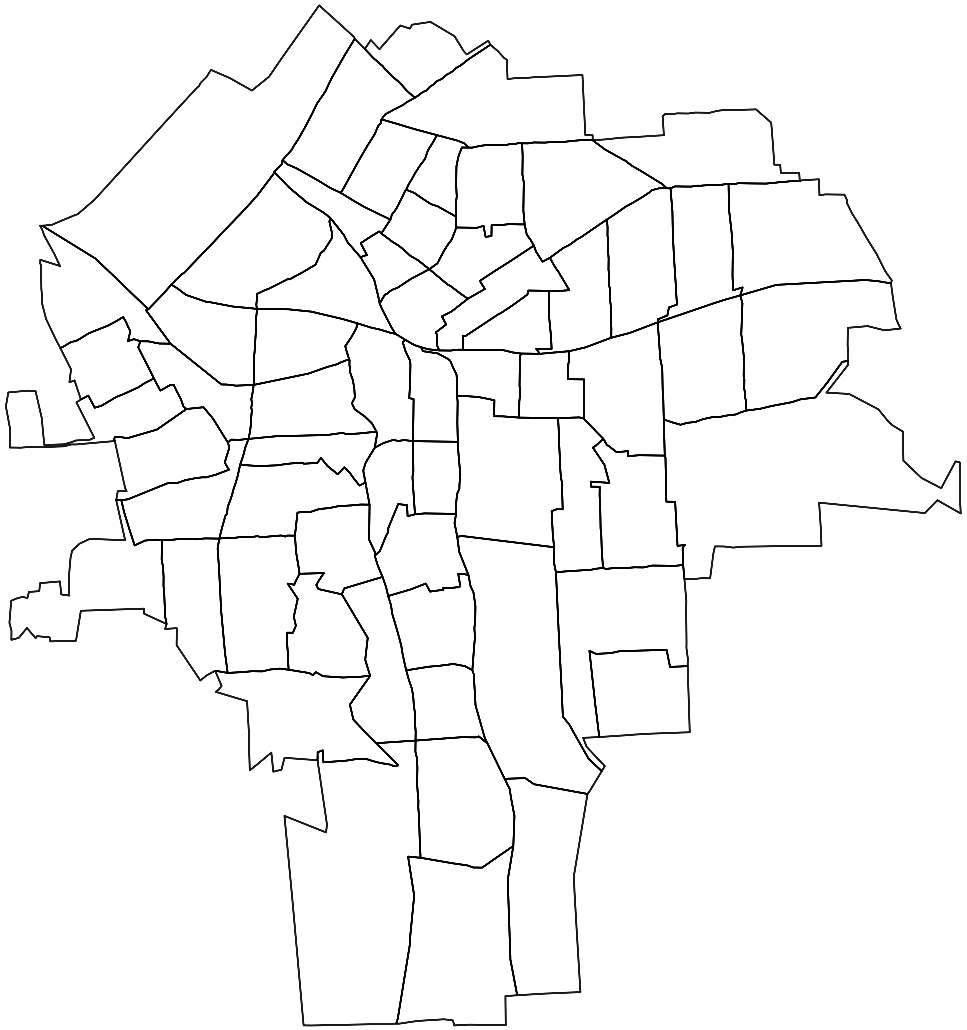
Plot the tract borders

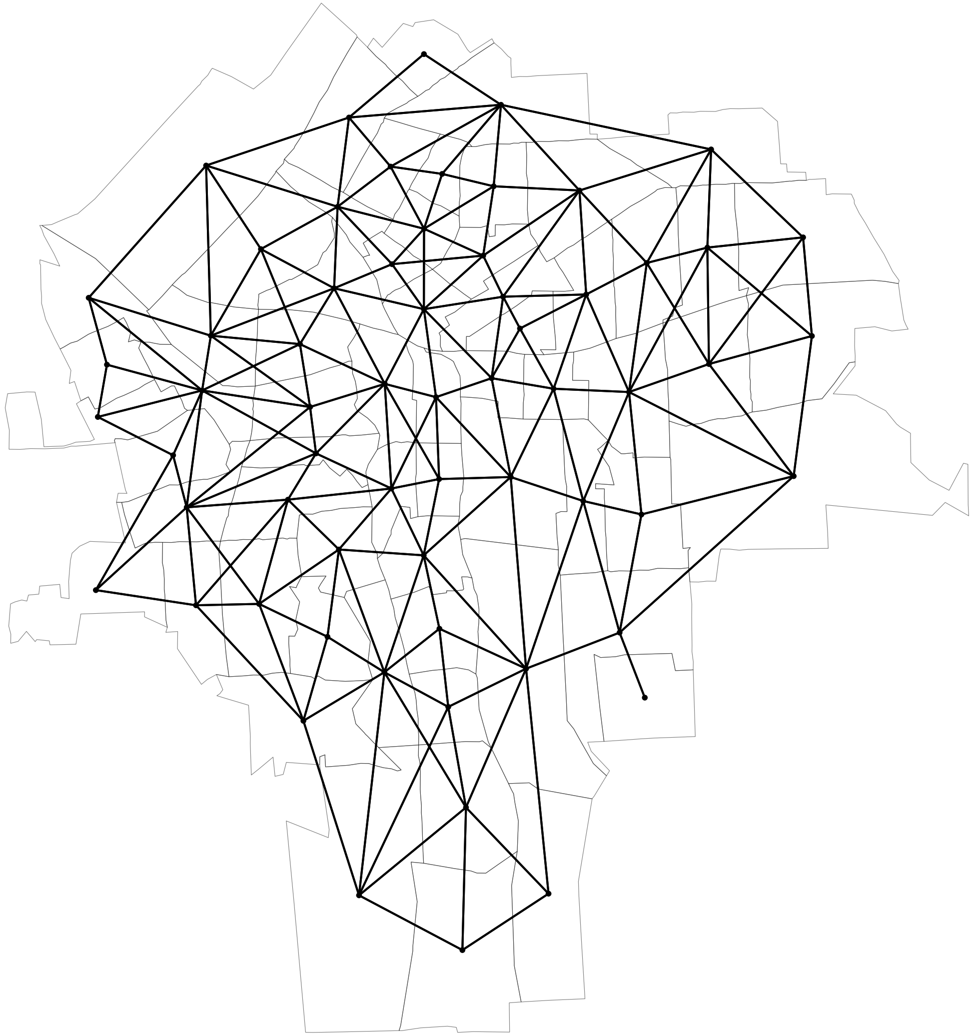
```
par(mar = c(0, 0, 0, 0))
plot(
  sy_sp,
  border = gray(0, 0.9),
  lwd = 1)
```

Overplot the neighborhood network

```
par(mar = c(0, 0, 0, 0))
plot(
  sy_sp,
  border = gray(0, 0.5),
  lwd = 0.3)
plot(
  sy_nb_w,
  coords = coordinates(sy_sp),
  add = T, pch = 16, cex = 0.4)
dev.off()
```

Plot the Census Tracts and Neighborhoods





Examine Autocorrelation in the Variables

- Count of leukemia cases in the census tracts:
  - Data are not integers!
  - "… because some cases could not be placed, they were added proportionally to other block groups, leading to non-integer counts."

```
moran.test(sy_sp$Cases, listw = sy_nb_w)
```

For Areal data, the function expects:

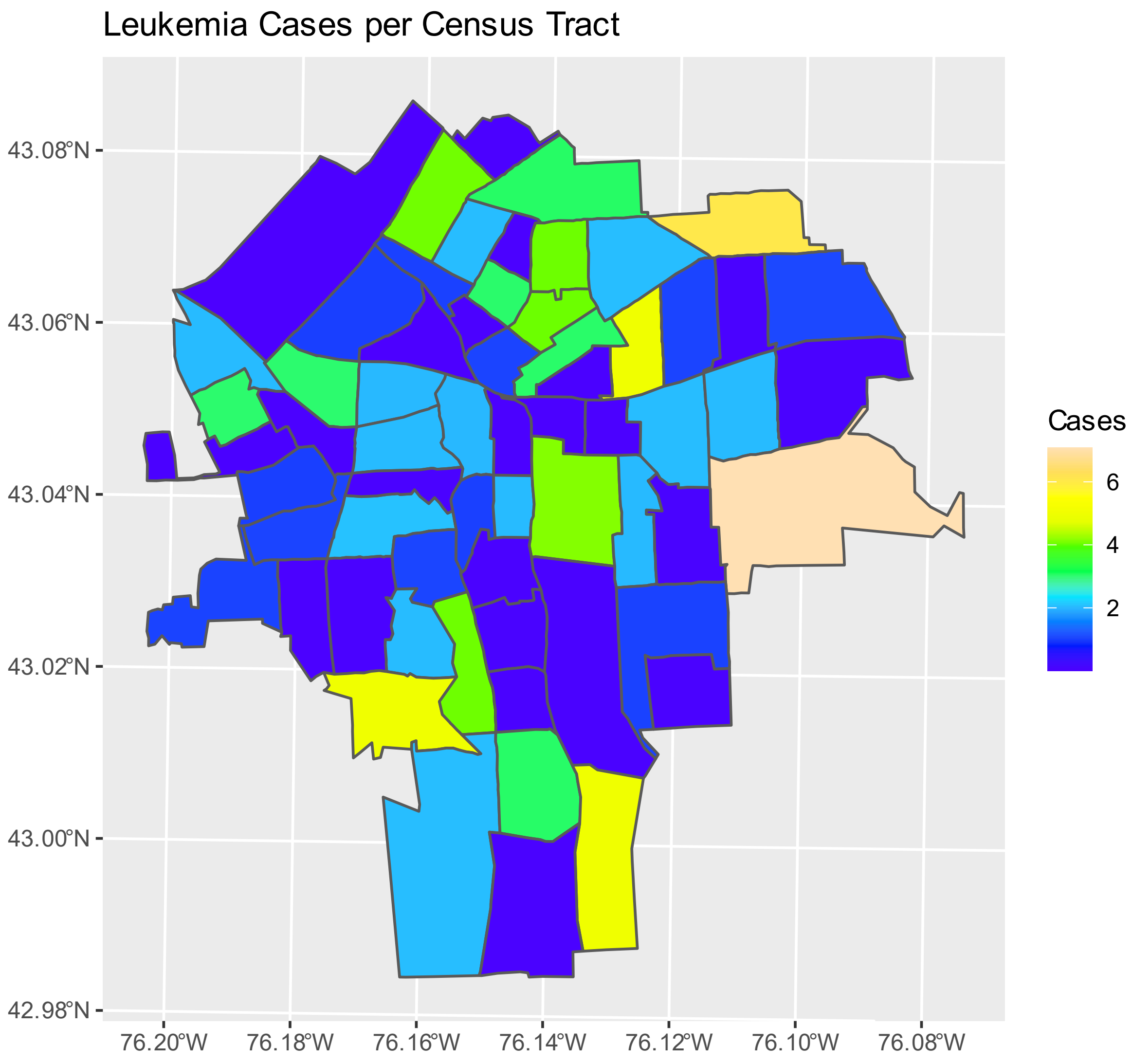
1. A vector of numbers
2. A neighbor weight object

Cases Per Tract

```
        Moran I test under randomisation

data:  sy_sp$Cases
weights: sy_nb_w

Moran I statistic standard deviate = -1.0531, p-value = 0.8538
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
     -0.095343683      -0.016129032       0.005658514
```

Cases Choropleth

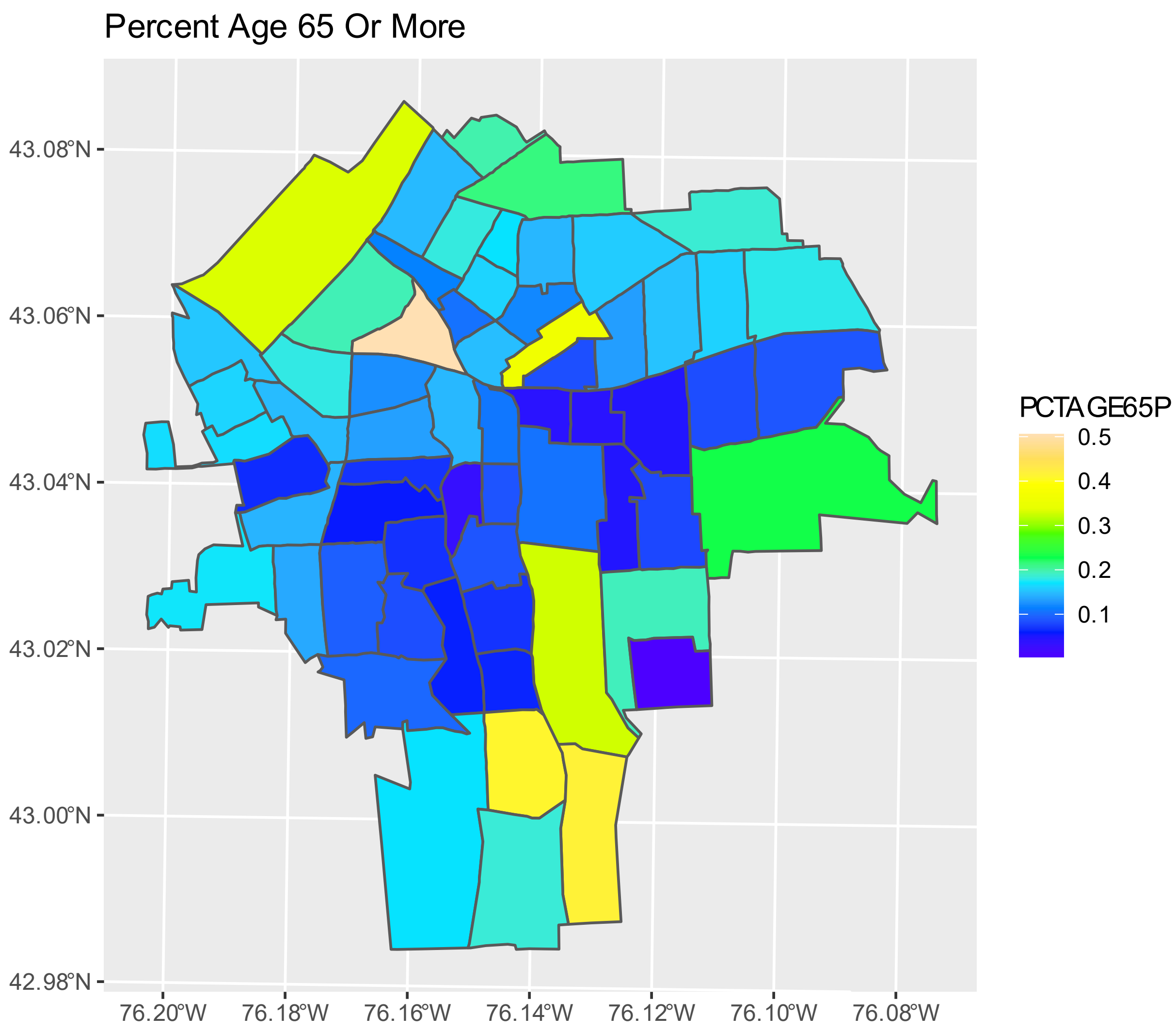
```
sy_sf = st_as_sf(sy_sp)
ggplot(sy_sf) +
  geom_sf(aes(fill = Cases))+
  ggtitle("Leukemia Cases per Census Tract")
```



Percent Age 65+

- Let's check for autocorrelation in the percentage of residents aged 65 or more:

```
ggplot(sy_sf) +
  geom_sf(aes(fill = PCTAGE65P))+
  ggtitle("Percent Age 65 Or More") +
  scale_fill_gradientn(
    colours = topo.colors(10))
```



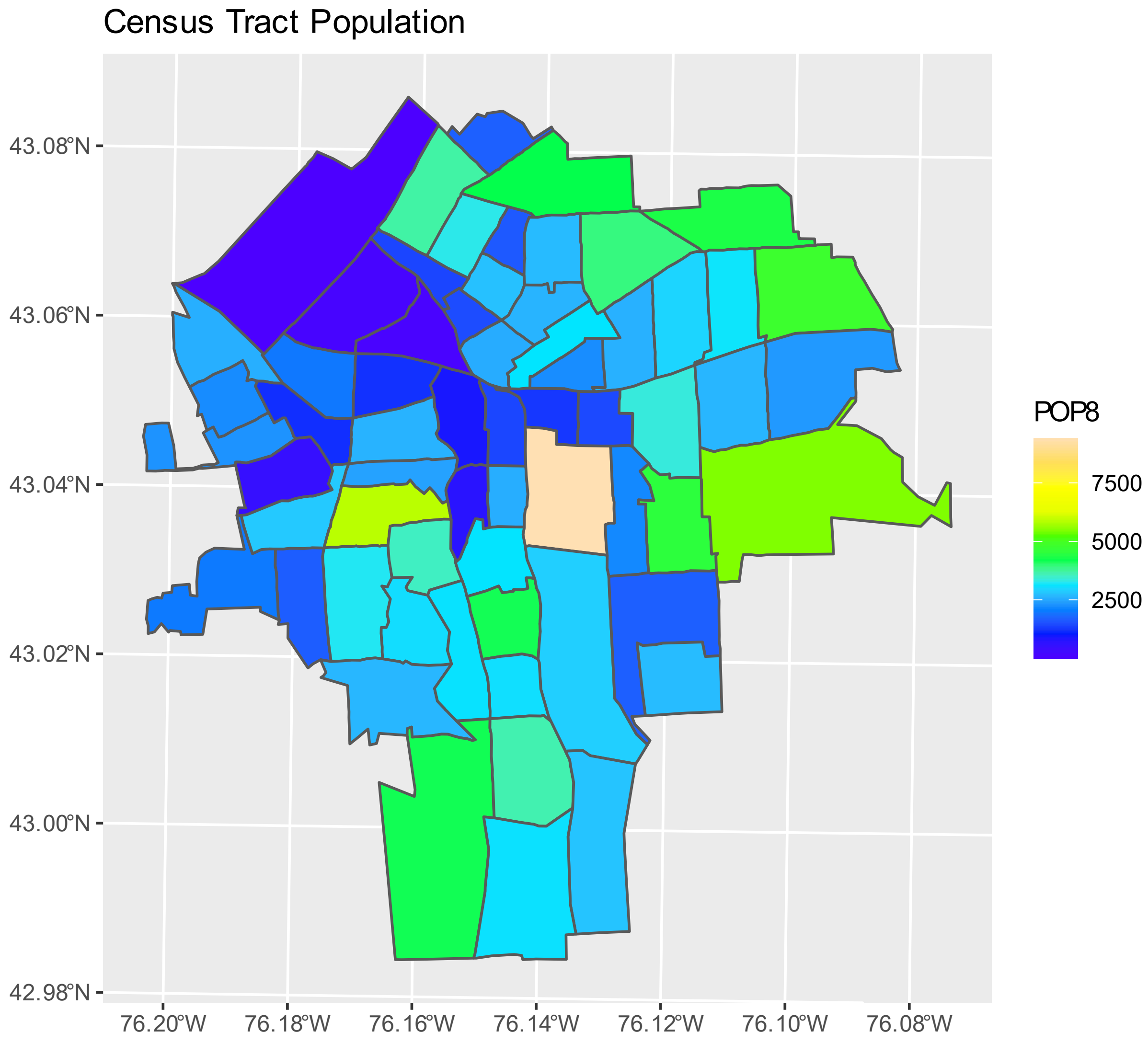
Percent Age 65+: Moran Test

```
moran.test(sy_sp$PCTAGE65P, listw = sy_nb_w)

	Moran I test under randomisation

data:  sy_sp$PCTAGE65P
weights: sy_nb_w

Moran I statistic standard deviate = 2.7341, p-value = 0.00312
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
      0.184687352      -0.016129032       0.005394923
```

Population

- Let's check for autocorrelation in the tract population:

```
ggplot(sy_sf) +
  geom_sf(aes(fill = POP8))+
  ggtitle("Census Tract Population") +
  scale_fill_gradientn(
    colours = topo.colors(10))
```

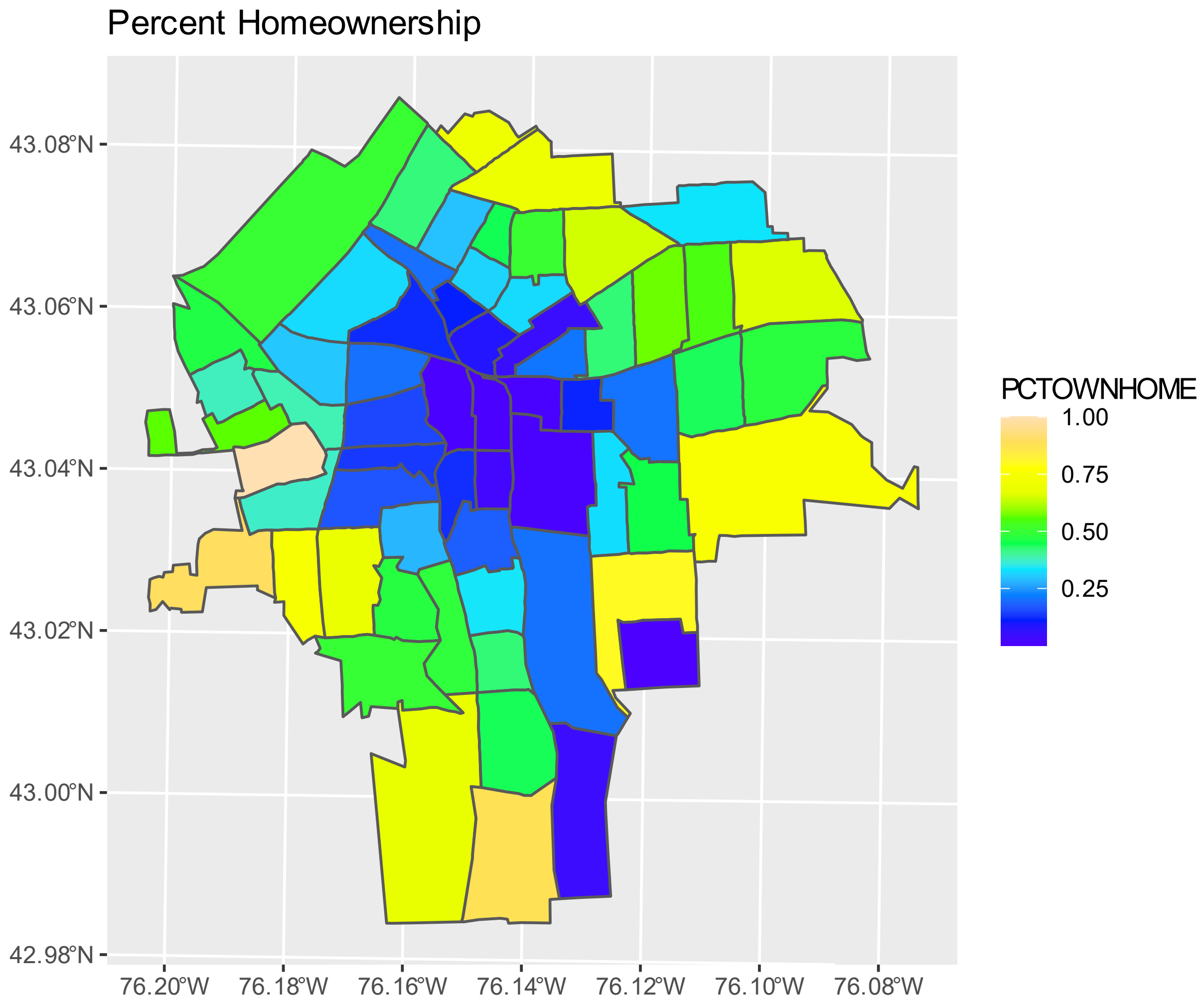


Percent Age 65+: Moran Test

```
moran.test(sy_sp$POP8, listw = sy_nb_w)


	Moran I test under randomisation

data:  sy_sp$POP8
weights: sy_nb_w

Moran I statistic standard deviate = 2.2158, p-value = 0.01335
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
       0.14361881       -0.01612903        0.00519757
```

Home Ownership

- Do you think there's autocorrelation?
- What do you notice about the city center?



Fit an Aspatial Model

Fit a model using two of the predictors:

- Percent age 65+
- Percent home ownership

```
fit_aspatial_1 = lm(
  Z ~ PCTAGE65P + PCTOWNHOME,
  data = sy_sf)
```

- Examine model summary

```
summary(fit_aspatial_1)
```

Model Summary

```
Call:
lm(formula = Z ~ PCTAGE65P + PCTOWNHOME, data = sy_sf)

Residuals:
    Min      1Q  Median      3Q     Max 
-1.8679 -0.5718 -0.2572  0.4032  3.9231 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)   
(Intercept)  -0.4938     0.2732  -1.807  0.07574 . 
PCTAGE65P     4.2242     1.2354   3.419  0.00113 **
PCTOWNHOME   -0.2536     0.4744  -0.535  0.59489   
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.9335 on 60 degrees of freedom
Multiple R-squared:  0.1642,    Adjusted R-squared:  0.1363 
F-statistic: 5.893 on 2 and 60 DF,  p-value: 0.004607
```

Autocorrelation in the Residuals?

```
moran.test(residuals(fit_aspatial_1), listw = sy_nb_w)


        Moran I test under randomisation

data:  residuals(fit_aspatial_1)
weights: sy_nb_w

Moran I statistic standard deviate = 2.844, p-value = 0.002228
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
      0.190565357      -0.016129032       0.005282012
```

Spatial Model: Spatial Filter

- Eigenvector filtering: add eigenvector as a model predictor

```
# Calculate most important eigenvector(s)
syr_me = ME(
  Z ~ PCTAGE65P + PCTOWNHOME,
  data = sy_sf, listw = sy_nb_w)

fit_filter_1 = lm(
  Z ~ PCTAGE65P + PCTOWNHOME + fitted(syr_me),
  data = sy_sf)

summary(fit_filter_1)
```

Spatial Filter Model Summary

We can interpret this just like a regular linear model summary. Note the eigenvector predictor coefficient.

```
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        -0.4739     0.2538  -1.867  0.06686 .
PCTAGE65P           3.8001     1.1547   3.291  0.00169 **
PCTOWNHOME         -0.1389     0.4420  -0.314  0.75447
fitted(syr_me)     -2.8447     0.8748  -3.252  0.00190 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.8669 on 59 degrees of freedom
Multiple R-squared:  0.2912,    Adjusted R-squared:  0.2552
F-statistic:  8.08 on 3 and 59 DF,  p-value: 0.0001354
```

Autocorrelation in the Residuals?

```
moran.test(residuals(fit_filter_1), listw = sy_nb_w)


        Moran I test under randomisation

data:  residuals(fit_filter_1)
weights: sy_nb_w

Moran I statistic standard deviate = 1.0184, p-value = 0.1542
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
      0.058416034      -0.016129032       0.005358211
```

Spatial Model: SAR

- Simultaneous Autoregressive Regression: model the variance/covariance matrix

```
fit_sar_1 = spautolm(
  Z ~ PCTAGE65P + PCTOWNHOME,
  data = sy_sf, listw = sy_nb_w)

summary(fit_sar_1)
```

SAR Model Summary

Model summary has a lot of info, we'll concentrate on the sign and significance of the coefficients…

… but notice the lambda value.

```
Residuals:
     Min       1Q   Median       3Q      Max
-1.59781 -0.46473 -0.24743  0.42949  3.61096

Coefficients:
              Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.5826945  0.3094914 -1.8827 0.0597344
PCTAGE65P    4.1023345  1.1988611  3.4219 0.0006219
PCTOWNHOME   0.0058742  0.5061358  0.0116 0.9907400

Lambda: 0.40776 LR test value: 5.2787 p-value: 0.021587
Numerical Hessian standard error of lambda: 0.16029

Log likelihood: -80.88484
ML residual variance (sigma squared): 0.73717, (sigma: 0.85859)
Number of observations: 63
Number of parameters estimated: 5
AIC: 171.77
```

Generalized Least Squares: Autoregressive Models

$$y = \alpha + x\beta + u$$

Response

Coefficients + predictors

Error Variance/Covariance Matrix.
The ‘lag’ or autoregressive part happens here

$$u = \lambda W \xi + \epsilon$$

Spatially independent error

Spatially dependent error

Spatial weight matrix

Autocorrelation in the Residuals?

```
moran.test(residuals(fit_sar_1), listw = sy_nb_w)


        Moran I test under randomisation

data:  residuals(fit_sar_1)
weights: sy_nb_w

Moran I statistic standard deviate = 0.074738, p-value = 0.4702
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
     -0.010660229      -0.016129032       0.005354309
```

Spatial Model: Lag

- Lag model adds an ‘autocovariate’ term: a function of the response and a weight matrix.

```
fit_lag_1 = lagsarlm(
  Z ~ PCTAGE65P + PCTOWNHOME,
  data = sy_sf, listw = sy_nb_w)
summary(fit_lag_1)
```

Spatial Lag Model Summary

Model summary has a lot of info, we'll concentrate on the sign and significance of the coefficients….

… but notice the rho term

```
Type: lag
Coefficients: (asymptotic standard errors)
             Estimate Std. Error z value  Pr(>|z|)
(Intercept) -0.51344    0.25293 -2.0299 0.0423651
PCTAGE65P    3.89572    1.15529  3.3721 0.0007461
PCTOWNHOME  -0.14013    0.43802 -0.3199 0.7490286

Rho: 0.37547, LR test value: 5.1601, p-value: 0.023112
Asymptotic standard error: 0.15455
    z-value: 2.4294, p-value: 0.015125
Wald statistic: 5.9018, p-value: 0.015125

Log likelihood: -80.94418 for lag model
ML residual variance (sigma squared): 0.74279, (sigma: 0.86185)
Number of observations: 63
Number of parameters estimated: 5
AIC: 171.89, (AIC for lm: 175.05)
LM test for residual autocorrelation
test value: 0.022373, p-value: 0.8811
```

Spatial Lag Models: Autocovariate

$$y_i = \underbrace{\rho WY}_{\text{Neighborhood-based lag component}} + \underbrace{\alpha + x_i\beta}_{\text{Coefficients + predictors}} + e_i$$

Response

Error
(single term)

Autocorrelation in the Residuals?

```
moran.test(residuals(fit_lag_1), listw = sy_nb_w)


       Moran I test under randomisation

data:  residuals(fit_lag_1)
weights: sy_nb_w

Moran I statistic standard deviate = 0.1613, p-value = 0.4359
alternative hypothesis: greater
sample estimates:
Moran I statistic       Expectation          Variance
     -0.004356871      -0.016129032       0.005326313
```

Recap

What did we learn?

- Age 65+ was positive and highly significant in all models.
  - Coefficients were similar, ranging from 3.8 – 4.2
- Spatial autocorrelation was present in the 65+ variable
- Spatial autocorrelation was present in the aspatial model's residuals
- All three spatially-aware models eliminated the autocorrelation.

Which model would you choose?

Model Comparison

- You could use RMSE or AIC

```r
# Model comparison by RMSE
rmse = function(fit) return (sqrt(mean(residuals(fit) ^ 2)))
fits_rmse = data.frame(
  model = c("aspatial", "SAR", "Lag", "Filter"),
  rmse = round(c(
    rmse(fit_aspatial_1),
    rmse(fit_sar_1),
    rmse(fit_lag_1),
    rmse(fit_filter_1)), 2))
```

```
     model  rmse
1 aspatial 0.911
2      SAR 0.859
3      Lag 0.862
4   Filter 0.839
```

Next Week

Plan for next week:

- Mini-lecture on weighted regressions
- In-class consultations about final project proposals
  - It’s advising season, and I won’t be able to schedule outside-of-class meetings.
- Work on labs and main projects in class
- Take a poll for topics to cover in the last few weeks of class!