Athanasios Votsis
(athanasios.votsis@helsinki.fi)

Session 4 Exercise: Geographically Weighted Regression

Spatial non-stationarity in regression coefficients refers to the case in which the coefficients at individual locations deviate from the global mean. Geographically weighted regression (GWR) aims at capturing these local variations. The method applies a moving Gaussian filter over the observations of a spatial dataset (points or polygons) and estimates regressions for each subset. GWR is considered exploratory, i.e. not a formal statistical technique.

1 – Importing and displaying a shapefile in R

1.1 Start RStudio and set your working directory by clicking `Session > Set Working Directory > Choose Directory…`

1.2 Install and/or load libraries `spdep`, `maptools`, `rgeos`, `spgwr`.

1.4 Import the shapefile `columbus` as a spatial polygons data-frame:

```
columbus <- readShapeSpatial("columbus.shp")
```

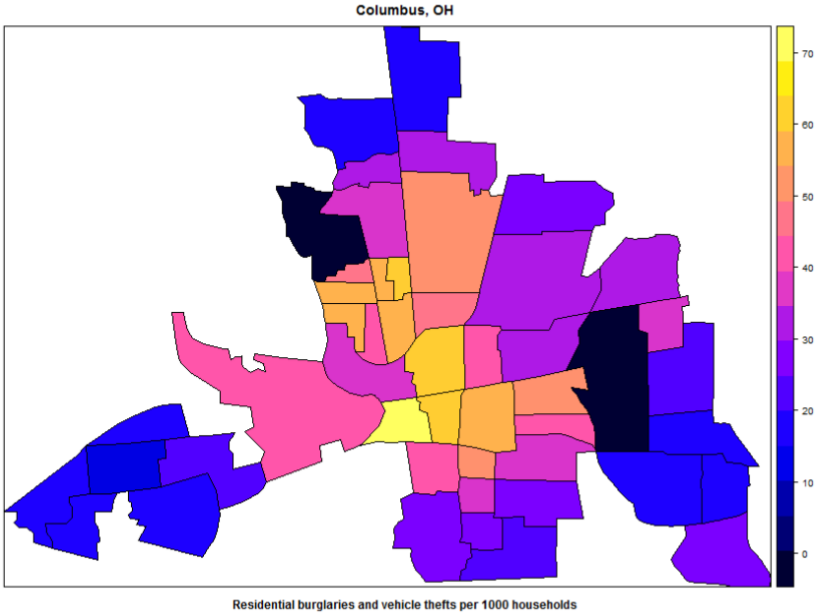
You can summarize the attribute data as usual by running the command:

```
summary(columbus),
```

You can overview your dependent variable (`CRIME`) and polygon geometry by running the command (with its default color scale and a couple of custom-defined titles):

```
spplot(columbus, "CRIME", main="Columbus, OH", sub="Residential
burglaries and vehicle thefts per 1000 households")
```

The results of the `spplot` command look like this:


Columbus, OH
70
60
50
40
30
20
10
0
Residential burglaries and vehicle thefts per 1000 households

2 – Using OLS regression to select independent variables

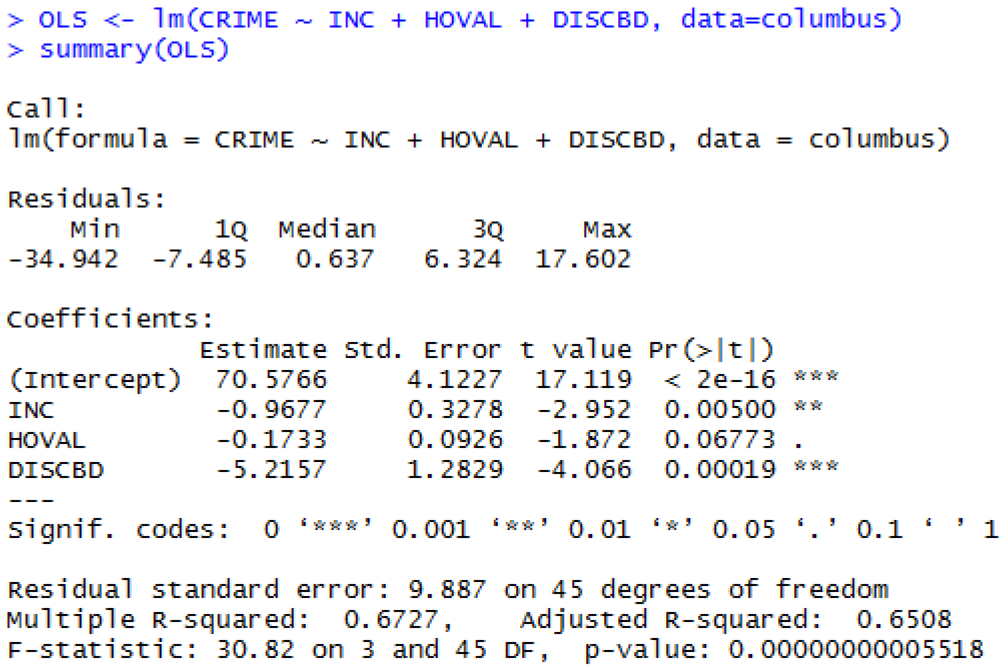
2.1 Our dependent variable is `CRIME`, and we assume that its global variation in crime (not taking into account local variation between polygons) can be explained adequately by variables indicating income (`INC`), property value (`HOVAL`), and distance to the city center (`DISCBD`). The list of the dependent and independent variables is as follows:

CRIME	residential burglaries and vehicle thefts per 1000 households
HOVAL	housing value (in $1,000)
INC	household income (in $1,000)
DISCBD	distance to CBD

2.2 Run an ordinary least squares (OLS) regression to see the average effects of `INC`, `HOVAL`, and `DISCBD` on `CRIM` for the whole study area:

```
OLS <- lm(CRIME ~ INC + HOVAL + DISCBD, data=columbus)
```

The summary of the results:



```
> OLS <- lm(CRIME ~ INC + HOVAL + DISCBD, data=columbus)
> summary(OLS)

Call:
lm(formula = CRIME ~ INC + HOVAL + DISCBD, data = columbus)

Residuals:
    Min      1Q  Median      3Q     Max 
-34.942  -7.485   0.637   6.324  17.602 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  70.5766     4.1227  17.119  < 2e-16 ***
INC          -0.9677     0.3278  -2.952  0.00500 ** 
HOVAL        -0.1733     0.0926  -1.872  0.06773 .  
DISCBD       -5.2157     1.2829  -4.066  0.00019 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 9.887 on 45 degrees of freedom
Multiple R-squared:  0.6727,	Adjusted R-squared:  0.6508 
F-statistic: 30.82 on 3 and 45 DF,  p-value: 0.0000000005518
```

3 – Estimating a GWR model

Based on the satisfactory results of the OLS model, we can use the same collection of variables and employ them in estimating a GWR model. The OLS results could have been explored in more detail by spatial regression specifications, as in the previous sessions. Here, we are interested in demonstrating how the average effects that global OLS and spatial regression models provide for a whole area can be broken down into locally varying effects.

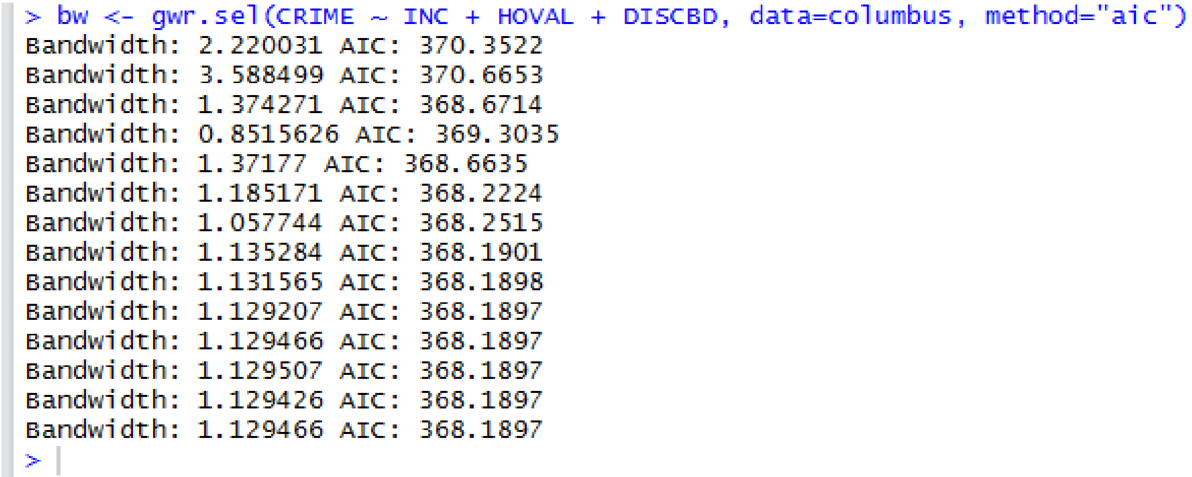
3.1 Bandwidth selection. Since GWR repeats the estimation for geographical subsets of your data (so it can get the local variation of the beat coefficients), it needs to know how extensive these geographical subsets should be. This is the idea behind bandwidth – it is loosely connected to the idea of spatial weights.

Option 1: you set the bandwidth by yourself.

Option 2: you employ an automatic search algorithm for finding the optimal bandwidth by running the command `grw.sel` and saving the result in the data-frame named `bw`. In our case:

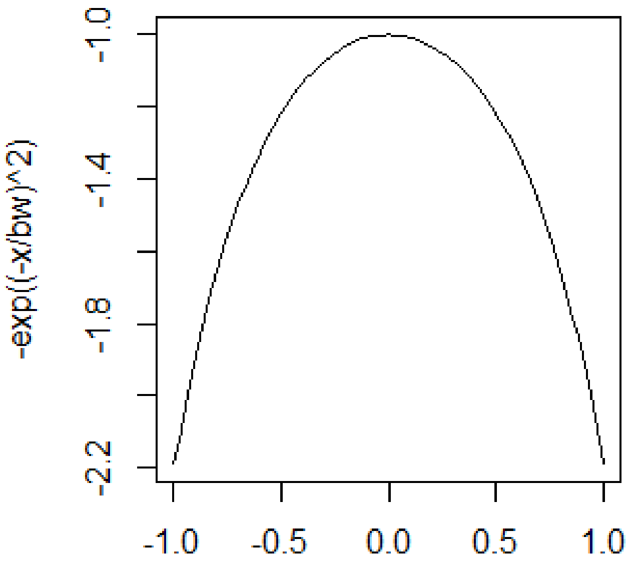
```
bw <- gwr.sel(CRIME ~ INC + HOVAL + DISCBD, data=columbus, method="aic")
```

The search stops when no further improvement in the performance of the bandwidth choice can be achieved. Performance can be measured by a couple of options and in our case we use the Akaike information criterion (AIC):

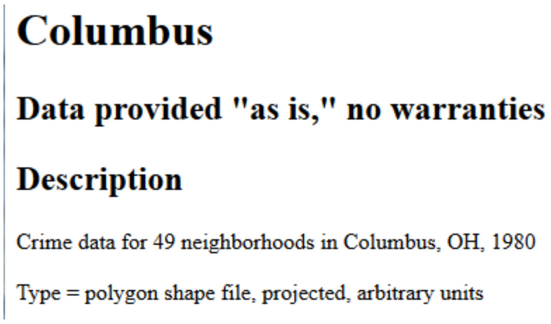


```
> bw <- gwr.sel(CRIME ~ INC + HOVAL + DISCBD, data=columbus, method="aic")
Bandwidth: 2.220031 AIC: 370.3522
Bandwidth: 3.588499 AIC: 370.6653
Bandwidth: 1.374271 AIC: 368.6714
Bandwidth: 0.8515626 AIC: 369.3035
Bandwidth: 1.37177 AIC: 368.6635
Bandwidth: 1.185171 AIC: 368.2224
Bandwidth: 1.057744 AIC: 368.2515
Bandwidth: 1.135284 AIC: 368.1901
Bandwidth: 1.131565 AIC: 368.1898
Bandwidth: 1.129207 AIC: 368.1897
Bandwidth: 1.129466 AIC: 368.1897
Bandwidth: 1.129507 AIC: 368.1897
Bandwidth: 1.129426 AIC: 368.1897
Bandwidth: 1.129466 AIC: 368.1897
> 
```

The Gaussian function that is implied by bandwidth = `1.129466`:



Important: If your shapefile is not in a projected coordinate system (e.g. in meters or arbitrary units), you need to include the setting `longlat=TRUE` (i.e. inform the algorithm that your file's geographical units are in degrees). In that case, the bandwidth will be measured in kilometers. If the shapefile is projected, then the bandwidth will be measured in the units of the file's projection system. The `columbus.shp` shapefile is projected, but its metadata do not tell us what the arbitrary units are. So, the bandwidth number `1.129466` cannot be interpreted:



Columbus

Data provided "as is," no warranties

Description

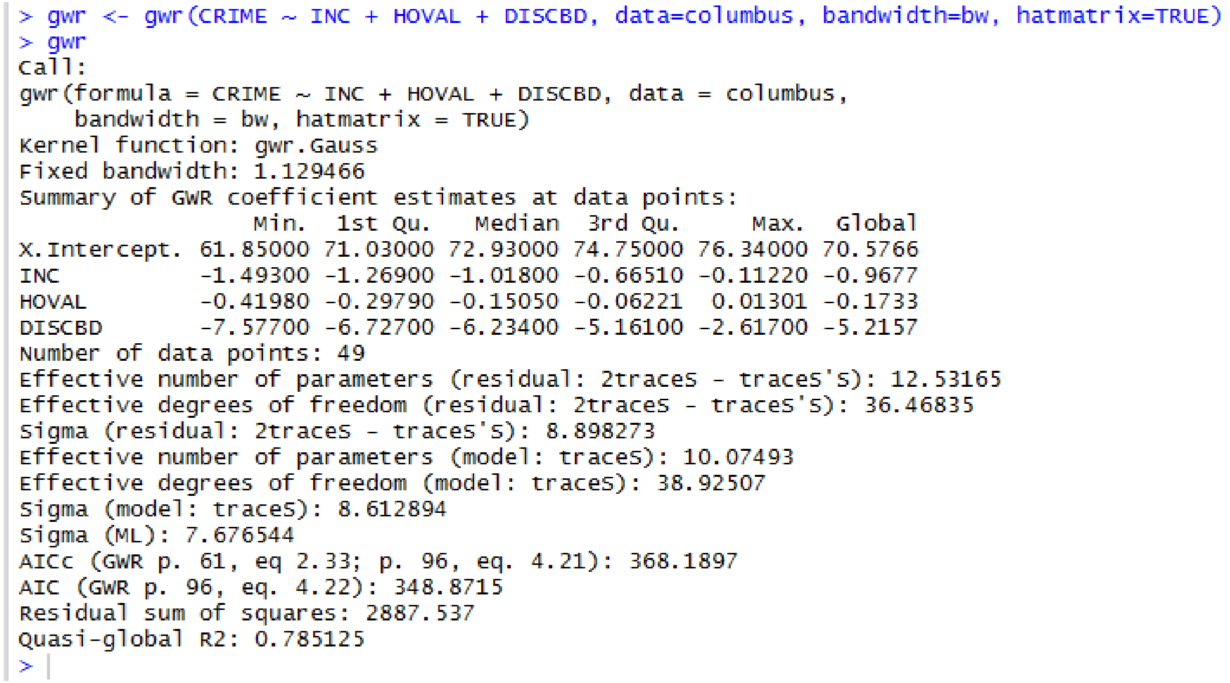
Crime data for 49 neighborhoods in Columbus, OH, 1980

Type = polygon shape file, projected, arbitrary units

3.2 Estimation of a GWR model. After finding the optimal bandwidth, you can estimate a GWR model by running the command `gwr`. The note about the `longlat` setting is valid for this command as well. In our case, the syntax will be:

```
gwr <- gwr(CRIME ~ INC + HOVAL + DISCBD, data=columbus, bandwidth=bw,
hatmatrix=TRUE)
```

You can display the results by typing `gwr` (i.e. no summary command is needed):



```
> gwr <- gwr(CRIME ~ INC + HOVAL + DISCBD, data=columbus, bandwidth=bw, hatmatrix=TRUE)
> gwr
Call:
gwr(formula = CRIME ~ INC + HOVAL + DISCBD, data = columbus,
    bandwidth = bw, hatmatrix = TRUE)
Kernel function: gwr.Gauss
Fixed bandwidth: 1.129466
Summary of GWR coefficient estimates at data points:
                  Min.  1st Qu.   Median  3rd Qu.     Max.  Global
X.Intercept. 61.85000 71.03000 72.93000 74.75000 76.34000 70.5766
INC          -1.49300 -1.26900 -1.01800 -0.66510 -0.11220 -0.9677
HOVAL        -0.41980 -0.29790 -0.15050 -0.06221  0.01301 -0.1733
DISCBD       -7.57700 -6.72700 -6.23400 -5.16100 -2.61700 -5.2157
Number of data points: 49
Effective number of parameters (residual: 2traceS - traceS'S): 12.53165
Effective degrees of freedom (residual: 2traceS - traceS'S): 36.46835
Sigma (residual: 2traceS - traceS'S): 8.898273
Effective number of parameters (model: traceS): 10.07493
Effective degrees of freedom (model: traceS): 38.92507
Sigma (model: traceS): 8.612894
Sigma (ML): 7.676544
AICc (GWR p. 61, eq 2.33; p. 96, eq. 4.21): 368.1897
AIC (GWR p. 96, eq. 4.22): 348.8715
Residual sum of squares: 2887.537
Quasi-global R2: 0.785125
> 
```

The above summary gives you the local variation of the estimated coefficients (min, max, median), as well as the global results (identical to the OLS results from step 2.1).

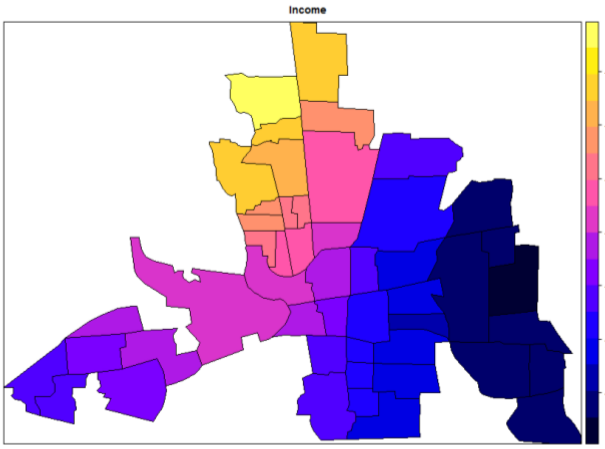
3.3 Mapping the estimations. The `gwr` dataframe borrows geographical structure from its input data (`columbus.shp`). This means that you can create thematic maps of the local coefficients by using the `spplot` command. For our three explanatory variables, the commands will be:

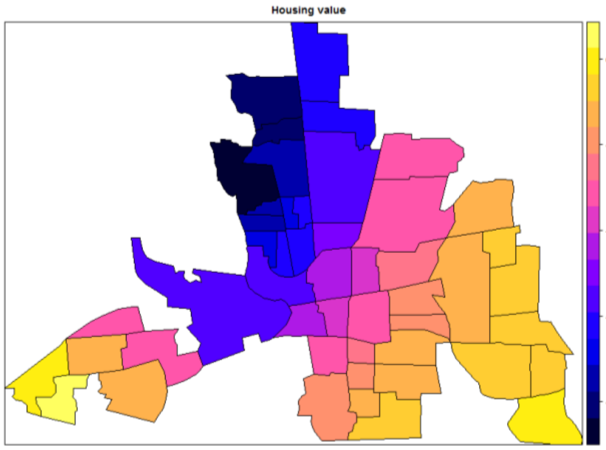
```
spplot(gwr$SDF, "INC", main="Income")
spplot(gwr$SDF, "HOVAL", main="Housing value")
spplot(gwr$SDF, "DISCBD", main="Distance to the CBD")
```

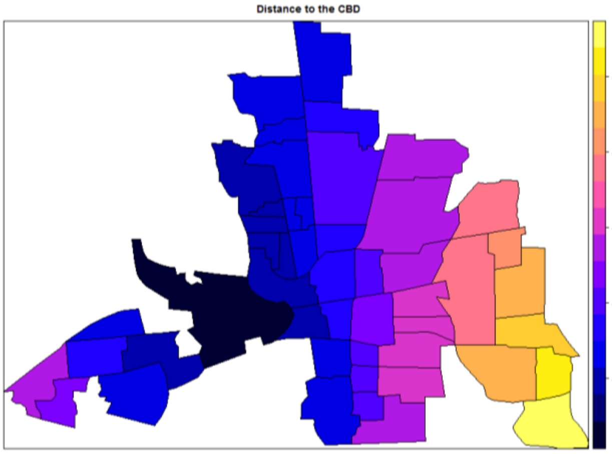
For the local R-squared of the estimates, the command will be:

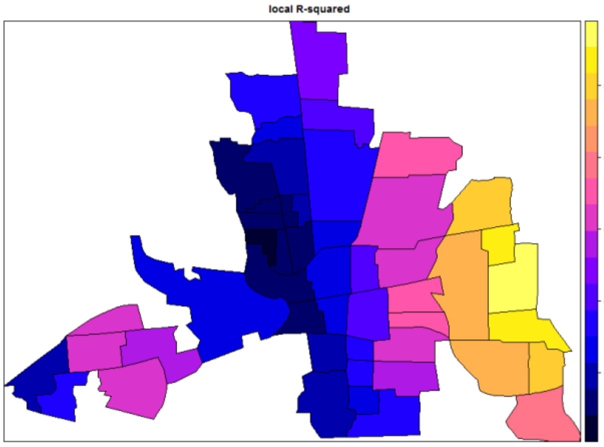
```
spplot(gwr$SDF, "localR2", main="local R-squared")
```

And the resulting maps will be:









In the above maps, we can see geographical patterns in the effect of `INC`, `HOVAL`, and `DISCBD` on `CRIME`.

Do we know, however, if the coefficients for each polygon are statistically significant?

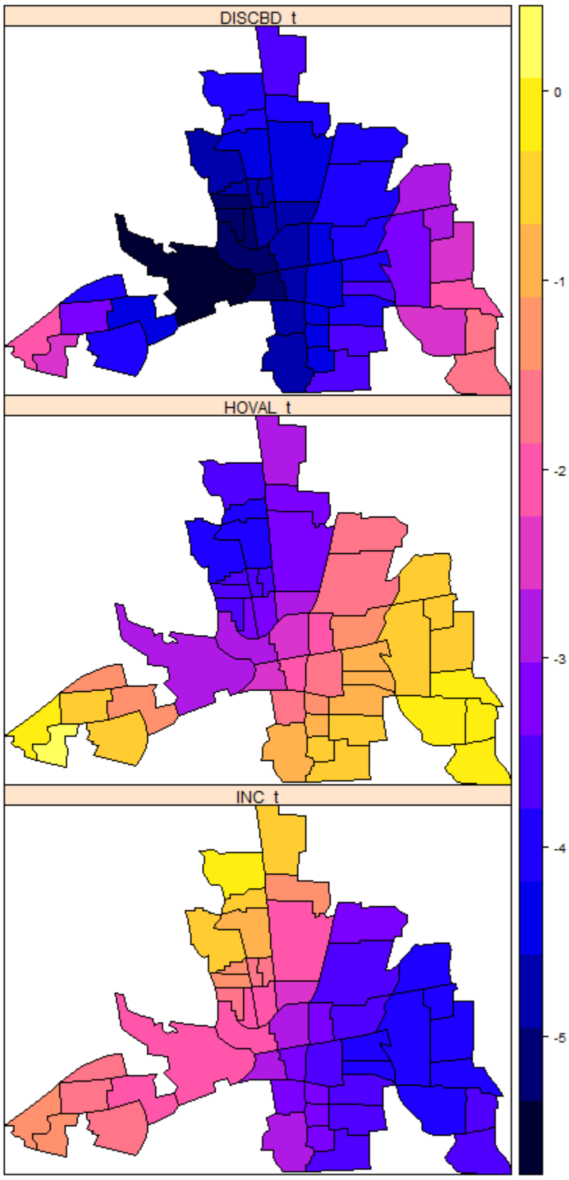
3.4 Calculate t-values for assessing the significance of the estimated effects. The GWR estimations also include the standard errors of the estimated coefficients. If you divide the estimated coefficient value by its standard error, you derive a t-statistic score:

```
gwr$SDF$INC_t <- gwr$SDF$INC/gwr$SDF$INC_se
gwr$SDF$HOVAL_t <- gwr$SDF$HOVAL/gwr$SDF$HOVAL_se
gwr$SDF$DISCBD_t <- gwr$SDF$DISCBD/gwr$SDF$DISCBD_se
```

And plot the three significance maps in a common figure:

```
          spplot(gwr$SDF, c("INC_t", "HOVAL_t", "DISCBD_t"))
```

The resulting map will be:



You can refer the OLS results to understand how the t-value's magnitude relates to the p-value of each estimated coefficient.

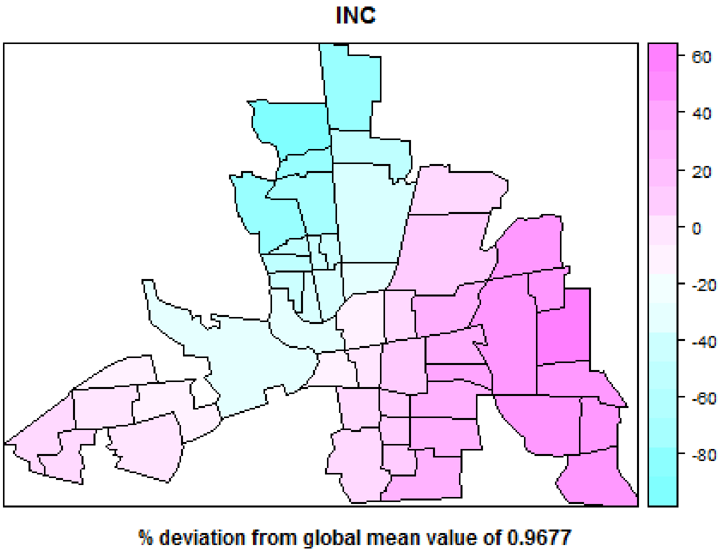
3.5 Deviations from global mean. In addition to step 3.3, you can visualize spatial non-stationarity in the estimated coefficients by calculating their % deviation from the global mean in each location. For the case of `INC`,

Append the % deviation in the spatial data-frame that holds the GWR results (-0.9677 is the global mean):

```
gwr$SDF$INC_pctdelta <- ((gwr$SDF$INC + 0.9677)/-0.9677)*100
```

And the thematic map with a custom color ramp:

```
spplot(gwr$SDF, "INC_pctdelta", col.regions=cm.colors(20), main="INC",
       sub="% deviation from global mean value of 0.9677")
```



3.6 Save the GWR results as a new shapefile. You will want to save the estimations in a new shapefile (they are not attached to your source `columbus` file) for more analytical or mapping flexibility. For instance, you may want to use more advanced mapping software or to analyze the estimations with GeoDa. The command to do the exporting is:

```
writeSpatialShape(gwr$SDF, "columbus_gwr")
```

The above command will save a new shapefile named `columbus_gwr.shp` inside your current working directory. The attribute table of the shapefile will contain (most of) the components found under `gwr$SDF` in R.

4 – Synthesis of spatial statistics tools

So far, we have learned about dealing with the end results (equilibrium results) of two fundamental aspects of spatial processes: spatial dependence and spatial heterogeneity:

- **Spatial dependence**: having to do with interaction between neighbors, and analyzed with spatial clustering indices and spatial regression models.
- **Spatial heterogeneity**: having to do with geographically non-constant effects of one variable on another, and analyzed with geographically weighted regression.

In practice both spatial dependence and spatial heterogeneity need to be explored, which means that all of the abovementioned tools of spatial statistics need to be employed.

Remember, however, that only spatial regression is considered formal analysis that may potentially verify causal relationships. Spatial clustering analysis and geographically weighted regression are considered more of exploratory tools and should not be used as indications of causality; just of interesting correlations.

The remaining exercises peek under the equilibrated end result and explore what happens in a spatial process as it evolves over time.