# Spatial Randomization Tutorial

This is a simple worked example showing how a one line command for spatial noise randomization works, and how it may be used as a simple robustness check on the reliability of Mueller-Watson corrections.

The data and code are available at https://github.com/morganwkelly/Spatial_Randomization.

First load the necessary `R` libraries and the randomization code.

```
library(spdep)
library(estimatr)
library(fields)
library(foreign)
library(tidyverse)

source('randomize.R')
```

The code and data for this example can be found at https://github.com/morganwkelly/persistence

Remember:

1. There can be no missing values.
2. Longitude and latitude *must* be called `X` and `Y`.

Load data in Stata format.

```
study=read.dta("example.dta")
```

Now write out the formula that you want to estimate, which we will call `frml`. The main explanatory variable of interest should be first.

```
frml="share_ind_work1901 ~ share_refractory + lnpop1901 + lnproto_ind +
mean_lnprec + mean_lntemp"
```

Regressions using spatial data should always have controls for spatial trends. This equation does not so the chances are that it is severely misspecified.

# Randomization

Now we calculate randomized significance based on the explanatory variable. You must specify a search range in kilometres beyond which correlation becomes negligible. It is best to start with a wide range with large jumps between values to tie down the approximate value, and then re-estimate closer to it. The range values do not need to be very exact. If your values are too high or low you will get a warning.

The default kappa or smoothness of 0.5 gives an exponential falloff which is empirically realistic. You can increase this in increments of 0.5 and check how the likelihood changes but in practice values in the range 0.5 to 1.5 return very similar values.

In this case we will run only 2000 simulations: this is usually enough in practice to give to fairly accurate results, but you will need about 10,000 if you want nice looking randomization distribution plots. Here we are searching over a range from 400 to 1000 km

`var_num` is the position of the variable in the equation that you want to test. In our case we are looking at variable 1 `share_refractory`.

```
x_rand=x_randomize(study,
                   frml=frml,
                   var_num=1,
                   Range_Search=seq(40,1000,by=20),
                   kappa=0.5,
                   nSim=10000)
```

The output is here.

```
round(x_rand$Output_x[2:4],2)
```

```
##   p_exact_x p_orig t_orig
## 1      0.28      0   3.46
```

First we have the randomized significance: the fraction of simulations where the t value on the noise variable was more extreme than the original t value of -3.5. In this case it is 0.37 compared with an original significance level of 0.001.

Next there is information on the spatial structure of the data.

```
round(x_rand$Output_x[-(1:4)],2)
```

```
##   coefficient se_exact_x se_orig   R2  N direction_R2_x Effective_Range_x
## 1       -0.21       0.19    0.06 0.54 80           0.17               520
##   Structure_x kappa_x Moran_p Likelihood_x df_x
## 1        0.98     0.5       0         96.5 0.95
```

1. Directional R2 tells how much of the orthogonalized explanatory variable is explained by a quadratic in longitude and latitude. For a properly specified model this should be zero. The value of 0.23 here suggests that the explanatory variable is acting as a proxy for omitted directional trends.

2. Effective range (where correlation between points has fallen to 0.14) is large. It is 940 km which is the approximate distance across the study range.
3. Spatial structure of 0.87 is high: most variables lie close to the predicted spatial surface with little idiosyncratic variation.
4. Kappa gives the smoothness parameter used, which here is the default exponential.
5. The Moran statistic, which has an asymptotic Gaussian distribution, is 5.6 suggesting strong spatial correlation in the regression residuals. Remember it should be seen as a rough indicator of the strength of spatial correlation, not a yes-no test of whether it is present.
6. Finally there is the likelihood of the spatial parameter estimates.

We can now repeat the process for the dependent variable.

```
y_rand=y_randomize(study,
                   frm=frml,
                   Range_Search=seq(100,400,by=10),
                   kappa=0.5,
                   nSim=2000)

round(y_rand$Output_y[-1],2)
```

```
##   p_exact_y p_orig t_orig coefficient se_exact_y se_orig   R2  N direction_R2_y
## 1      0.11      0   3.46       -0.21       0.13    0.06 0.54 80           0.36
##   Effective_Range_y Structure_y kappa_y Moran_p Likelihood_y df_y
## 1               260        0.96     0.5       0       105.01 0.95
```

It can again be seen that the orthogonalized dependent variable has substantial structure and is strongly explained by quadratic longitude and latitude. The randomized p value is larger in this case than with the explanatory variable.

## Spatial Trend Controls

Now we consider what happens if we add longitude and latitude to our regression as a control for omitted variables.

```
frml_trend="share_ind_work1901~share_refractory+lnpop1901+lnproto_ind+
mean_lnprec+mean_lntemp+X+Y"
```

We can repeat the randomization of the explanatory variable.

```
x_rand_trend=x_randomize(study,frm=frml_trend,
                   Range_Search=seq(400,1000,by=10),
                   kappa=0.5,nSim=2000)

round(x_rand_trend$Output_x[-1],2)
```

```
##   p_exact_x p_orig t_orig coefficient se_exact_x se_orig   R2  N direction_R2_x
## 1      0.69   0.63   0.48       -0.02       0.06    0.05 0.79 80           0.08
##   Effective_Range_x Structure_x kappa_x Moran_p Likelihood_x df_x
## 1               420        0.97     0.5    0.04        99.35 0.95
```

Having controlled for spatial trends, the randomized significance of 0.69 is now similar to the nominal significance of 0.63. At the same time, the Moran significance is 0.04