**DISTANCE BUFFER FOR RANDOMIZATION**

This tool aims to provide an example with its respective Stata code to randomly assign geographic units to treatment and control groups in Randomized Controlled Trials (RCT) minimizing the risk of contamination between geographic units due to their proximity to each other.

**STEPS TO INCORPORATE DISTANCE BUFFER IN A RANDOM WAY**

1. **Geolocalize (ie. obtain latitude and longitude) the units that are potentially eligible for treatment**
   * First you must obtain a fairly big list of geographical units, for example villages. This must be a really big group, much bigger than the total amount of villages in Treatment and Control you obtained in your power calculations, because when the process is over, only a few will be good to use[[1]](#footnote-1).
   * Second you must obtain latitude and longitude for each unit. If you don’t already have this, a possible way to go is to:
     + If you have the addresses, you can paste the list of addresses in a webpage like this one, and obtain the list of latitudes and longitudes ([https://geocode.localfocus.nl](https://geocode.localfocus.nl/))
     + If you don’t have the addresses, you can manually place each village in google maps (you can place the center of the village, or a relevant reference point), and obtain the latitude and longitude from google maps, or a similar service (I suggest you search online).
2. **Calculate distances between all possible pairs of villages** 
   * You must first form all possible pairs of villages among the villages in your dataset.
   * Then you must calculate distances between all the possible pairs of villages. For example, if you had only 3 villages, A, B, and C, you need to find distance between (A;B), (A;C); (B;A); (B;C); (C;A); (C;B). Of course this distance will be the same for pairs (A;B) and (B;A) but this will simplify the procedure afterwards.
3. **Keep only the villages which are sufficiently far away from each other**
   * Once you have your table with the distances among all possible pairs of villages, you must only keep the villages which are at least N kilometers away from all other villages, ie. only pairs which survive the distance cutoff. **Since you checked the distance between every possible pair, you can be sure that the villages in the remaining pairs are far away from \*all other\* villages in the database.**
4. **Proceed with randomization**
   * Finally, using the database with the remaining villages, which are all sufficiently far away from each other so as to avoid contamination, you carry out the randomization as you would have done it in a normal case. For this step, you must set a seed in your dofile, so that we can then run the code again and get the same result. This is important for testing purposes but also for registration and replication.

**Let's see an example.**

* + - 1. In dofile “Distance buffer for randomization.do” database “example\_data .dta" has the latitude and longitude of villages in three communes of Benin (Abomey-Calavi, Cotonou, and Parakou).

With the command ***cross*** we create all possible pairs of villages in “example\_data .dta".

1. With the command ***geodist*** we calculate the distance between villages. You can choose the distance you want[[2]](#footnote-2).

Later, we only keep the villages which are at least 10 kilometers away from all other villages.

We cancel duplication village observations.

Once these steps have been carried out, we save this selection of villages in "example\_data\_with\_distance.dta".

Later, we create a database for each communes (Abomey Calavi, Cotonou and Parakou).

1. Finally, we carry out the randomization of villages by communes. In this case, we make up three groups (two treatment groups and a control group) for each commune. Each group contains 14 randomly selected villages with ***sample*** command. ***set seed*** allows us to replicate the randomization.

1. Do *consider that increasing the distance among villages will also increase the budget destined to travel costs if needed. Remember to take a look at budget as well.*  [↑](#footnote-ref-1)
2. The *geodist* command <https://ideas.repec.org/c/boc/bocode/s457147.html> allows you to calculate the distance between two pairs of longitude and latitude points.

   Install the *geodist* command by typing *ssc install geodist* in the Stata window*.* [↑](#footnote-ref-2)