Soil Sampling Design

Technical Manual

Rodríguez Lado, L., Angelini, M.E, Naypewe, N., Luotto, I., Yigini, Y.

2023-12-07

Contents

Licence

Chapter 1

Introduction

Understanding the spatial distribution of soil properties is crucial for making informed decisions in various fields, from precision agriculture to environmental conservation. The success of soil mapping activities relies on the existence of proper data collated through meticulous soil sampling protocols. The spatial variation of soil properties across landscapes necessitates strategic planning to ensure representative and reliable data collection.

In this manual, we explore the intricacies of soil sampling design, including the critical factors that influence the accuracy and effectiveness of soil data field sampling. We present examples of various sampling methods, ranging from traditional grid-based approaches to advanced statistical sampling strategies. We include methods for systematic, random and stratified sampling, evaluating their strengths and weaknesses in the context of DSM. We aim to provide researchers and practitioners with the knowledge necessary to select the most suitable approach for their specific objectives.

We have used a common structure for file paths in the exercises. By default, the RStudio console points to the folder where the project is located. Thus, R scripts appear in the root of the working directory and data files are in a `'data/'` directory within the root, with `'.shp'` and `'.tif'` files located within the sub-folders `'data/shapes'` and `'data/rasters'` respectively. Following this recommendation simplifies the definition of paths and execution of the scripts. If users desire to change their storage paths, they have to properly adjust data paths in the R scripts.

We use examples based on the data and scripts at (Malone, Minansy and Brungard, 2019), which can be found at the repository.

1.1 Manual Structure

A centralised data repository for soil data offers a multitude of benefits, spanning from enhanced accessibility and data sharing to improved data quality and informed decision-making.

- Accessibility: A single repository provides a unified access point for various stakeholders, including researchers, farmers, policymakers, and educators, facilitating easy retrieval of information. Centralization promotes interdisciplinary research and collaboration by making soil data accessible across different scientific and agricultural disciplines.
- Quality: A centralised data repository ensures that data from various sources is standardised in format, making it easier to compare and analyse and providing better control over the quality of soil data, as it can be vetted and validated through standardised protocols.
- Efficient Data Management: Centralised repositories provide organised storage, making it easier to manage vast amounts of soil data efficiently. Central repositories ensure long-term preservation of soil data, protecting it from loss due to localised issues like technical failures or organisational changes.
- Improved Data Analysis and Research: Having a centralised repository means researchers can access more comprehensive data sets, leading to more robust and inclusive research outcomes. Centralization facilitates the application of advanced data analytics, including AI and machine learning, to uncover deeper insights and patterns in soil data.
- Support for Policy and Decision Making: Access to comprehensive soil data aids policymakers in developing informed, evidence-based agricultural and environmental policies. This support can be crucial in managing risks related to agriculture, such as soil degradation, contamination, and climate change impacts.
- Enhanced Educational and Outreach Opportunities: A centralised soil data repository serves as an invaluable resource for educational institutions, enhancing learning and research opportunities for students.and it can also play a role in raising public awareness about soil health and sustainable agricultural practices.
- Facilitation of Digital Initiatives: Centralised data repositories are essential for digital soil mapping initiatives, providing the necessary data to create detailed and accurate soil maps. They facilitate the integration of soil data with other technological tools, like GIS and remote sensing, enhancing the scope of soil analysis and interpretation.
- Global Collaboration and Benchmarking: A centralised repository can serve as a platform for international collaboration, sharing best practices and data across borders. It allows for benchmarking and comparative studies at a global scale, essential for understanding and addressing global soil health issues.

A centralised data repository for soil data is a powerful tool that can transform how soil information is managed and utilised. By providing a platform for standardised, high-quality, and accessible soil data, it supports a range of activities from scientific research to policy making, ultimately contributing to more sustainable and informed management of soil resources worldwide.

Conditioned Latin Hypercube Sampling (cLHS)

Conditioned Latin Hypercube Sampling (cLHS) is an advanced statistical method used for sampling multidimensional data developed within the context of digital Soil Mapping. It's an extension of the basic Latin Hypercube Sampling (LHS) technique, a statistical method for generating a distribution of samples of a random variable. The main advantage of LHS over simple random sampling is its ability to ensure that the entire range of the auxiliary variables are explored. It divides the range of each variable into intervals of equal probability and samples each interval.

The term "conditioned" refers to the way the sampling is adapted or conditioned based on specific requirements or constraints. It often involves conditioning the sampling process on one or more additional variables or criteria. This helps in generating samples that are not just representative in terms of the range of values, but also in terms of their relationships or distributions. cLHS is particularly useful for sampling from multivariate data, where there are multiple interrelated variables as it occurs in soil surveys. The main advantage of cLHS is its efficiency in sampling and its ability to better capture the structure and relationships within the data, compared to simpler sampling methods, and ensures that the samples are representative not just of the range of each variable, but also of their interrelations. Detailed information on cLHS can be found in (Minasny and Mcbratney, 2006)

cHLS is also used to determine the optimal number of samples that cover the entire auxiliary data variability.

(Sena *et al.*, 2021) proposed a strategy for sampling in difficult access areas using cLHS.

(Clifford *et al.*, 2014) Pragmatic soil survey design using flexible Latin hypercube sampling for difficult access.

Part one - Soil Legacy Data

Chapter 2

Evaluating Soil Legacy Data Sampling for DSM

Modelling techniques in Digital Soil Mapping involve the use of sampling point soil data, with its associated soil properties database, and a number of environmental covariates that will be used to ascertain the relationships of soil properties and the environment to then generalize the findings to locations where no samples have been compiled.

In soil sampling design, a crucial issue is to determine both the locations and the number of the samples to be compiled. In an optimal situation, soil sample database should adequately cover all the environmental diversity space in the study area with a frequency relative to the extent of the diversity in the environmental covariates.

When dealing with legacy soil data, a question that arises is if the data is representative of the environmental diversity within the study area. In this Chapter we present a method to answer this question and to build an alternative how many samples can be retrieved to cover the same environmental space as the existing soil data. The method follows the main findings in (Malone, Minansy and Brungard, 2019) and developed as {R} scripts.

We adapted the original scripts to make use of vector `'.shp'` and raster `'.tif'` files, as these are data formats commonly used by GIS analysts and in which both soil and environmental data is often stored. We also made some changes in order to simplify the number of R packages and to avoid the use of deprecated packages as it appears in the original code.

2.1 Data Preparation

We must load the required packages and data for the analyses. We make use of the packages `sp` and `terra` to manipulate spatial data, `clhs` for Conditioned Latin Hypercube Sampling, `entropy` to compute Kullback-Leibler (KL) divergence indexes, `tripack` for Delaunay triangulation and `manipulate` for interactive plotting within RStudio. Ensure that all these packages are installed in your system before the execution of the script.

We define the working directory to the directory in which the actual file is located and load the soil legacy sampling points and the environmental rasters from the `data` folder. To avoid the definition of each environmental covariate, we first retrieve all files with the `.tif` extension and then create a `SpatRaster` object with all of them in a row.

```
# Set working directory to source file location
  #setwd(dirname(rstudioapi::getActiveDocumentContext()$path))
```

```
## Load soil legacy point data
  p.dat <- terra::vect("data/shapes/legacy_soils.shp")

## Load raster covariate data----
  # Read Spatial data covariates as rasters with terra
  rasters <- "data/rasters"
  cov.dat <-  list.files(rasters, pattern = "tif$",  recursive = TRUE, full.names = TRU
  cov.dat <- terra::rast(cov.dat)
```

2.2 Representativeness of the Legacy Soil Data

The next step involves the determination of the distributions of environmental values in the soil samples data and its comparison with the existing distributions of each environmental variable to determine the representativeness of the soil points in the environmental space.

The comparison of distributions is performed through the Kullback-Leibler divergence (KL). It is a measure used to quantify the difference between two probability distributions. KL-divergence compares an 'objective' or reference probability distribution (here, the distribution of covariates in the complete covariate space - P) with a 'model' or approximate probability distribution (the space of covariates in the soil samples - Q). The main idea is to determine how much information is lost when Q is used to approximate P. In other words, KL-divergence measures how much the Q distribution deviates from the P distribution.

We cross soil and environmental data to create a dataset with the values of the environmental parameters at the locations of the soil samples.

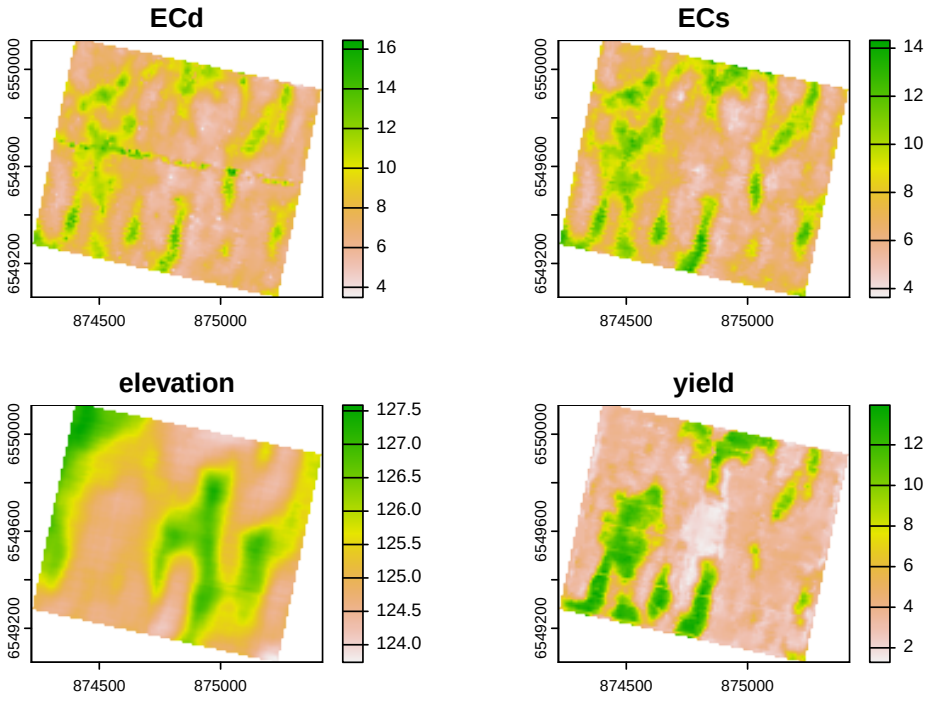


Figure 2.1: Covariates

```
# Extract environmental data from rasters at soil locations ----
  p.dat_I <- terra::extract(cov.dat, p.dat)
  p.dat_I <- na.omit(p.dat_I) # Remove soil points outside study area
  str(p.dat_I)
```

```
## 'data.frame':    238 obs. of  5 variables:
##  $ ID       : num  1 2 3 4 5 6 7 8 9 10 ...
##  $ ECd      : num  7.48 5.86 7.4 6.84 6.2 ...
##  $ ECs      : num  7.86 5.67 7.81 6.99 5.51 ...
##  $ elevation: num  127 127 127 127 127 ...
##  $ yield    : num  3.03 3.79 3.37 2.62 3.69 ...
```

We first calculate a 'n-matrix' with the values of the covariates dividing their distributions into 'n' equally-spaced bins. Each bin captures the environmental variability within its interval in the total distribution. In this exercise, 'n' equals to 25. The result is a 26×4 matrix, where the rows represent the upper and lower limit of the bin and (thus, 26 rows are required to represent 25 bins), and 4 correspond to the number of variables used as environmental proxies.

```
# Define Number of bins
  nb<- 25
  #quantile matrix (of the covariate data)
  q.mat<- matrix(NA, nrow=(nb+1), ncol= nlyr(cov.dat))
  j=1
  for (i in 1:nlyr(cov.dat)){ #note the index start here
  #get a quantile matrix together of the covariates
    ran1 <- minmax(cov.dat[[i]])[2] - minmax(cov.dat[[i]])[1]
    step1<- ran1/nb
    q.mat[,j]<- seq(minmax(cov.dat[[i]])[1], to = minmax(cov.dat[[i]])[2], by =step1)
    j<- j+1}
```

From this matrix, we compute the hypercube matrix of covariates in the whole covariate space.

```
# Hypercube of "objective" distribution (P) - covariates
  cov.dat.df <- as.data.frame(cov.dat) # convert SpatRaster to dataframe
  cov.mat<- matrix(1, nrow=nb, ncol=ncol(q.mat))
    for (i in 1:nrow(cov.dat.df)){ # the number of pixels
      cntj<- 1
    for (j in 1:ncol(cov.dat.df)){ #for each column
      dd<- cov.dat.df[i,j]
      for (k in 1:nb){  #for each quantile
        kl<- q.mat[k, cntj]
        ku<- q.mat[k+1, cntj]
        if (is.na(dd)) {
          print('Missing')
        }
```

```
        else if (dd >= kl & dd <= ku){cov.mat[k, cntj]<- cov.mat[k, cntj] + 1}
      }
      cntj<- cntj+1
    }
  }
```

We then calculate the hypercube matrix of covariates in the sample space.

```
# Sample data hypercube
  h.mat<- matrix(1, nrow=nb, ncol=ncol(q.mat))

  for (ii in 1:nrow(p.dat_I)){ # the number of observations
    cntj<- 1
    for (jj in 2:ncol(p.dat_I)){ #for each column
      dd<- p.dat_I[ii,jj]
      for (kk in 1:nb){  #for each bin
        kl<- q.mat[kk, cntj]
        ku<- q.mat[kk+1, cntj]
        if (dd >= kl & dd <= ku){h.mat[kk, cntj]<- h.mat[kk, cntj] + 1}
      }
      cntj<- cntj+1
    }
  }
```

- **KL-divergence**

We calculate the KL-divergence to measure how much the distribution of covariates in tbe sample space (Q) deviates from the distribution of covariates in the complete study area space (P).

```
## Compare covariate distributions in P and Q with Kullback-Leibler (KL) divergence
    kl.index <-c()
    for(i in 1:ncol(cov.dat.df)){
      kl <-    KL.empirical(c(cov.mat[,i]), c(h.mat[,i]))
      kl.index <- c(kl.index,kl)
      klo <-  mean(kl.index)
    }
    print(kl.index) # KL divergences of each covariate
```

```
## [1] 0.04115895 0.04241792 0.02779852 0.04328375
```

```
    print(klo) # KL divergence in the existing soil samples
```

```
## [1] 0.03866478
```

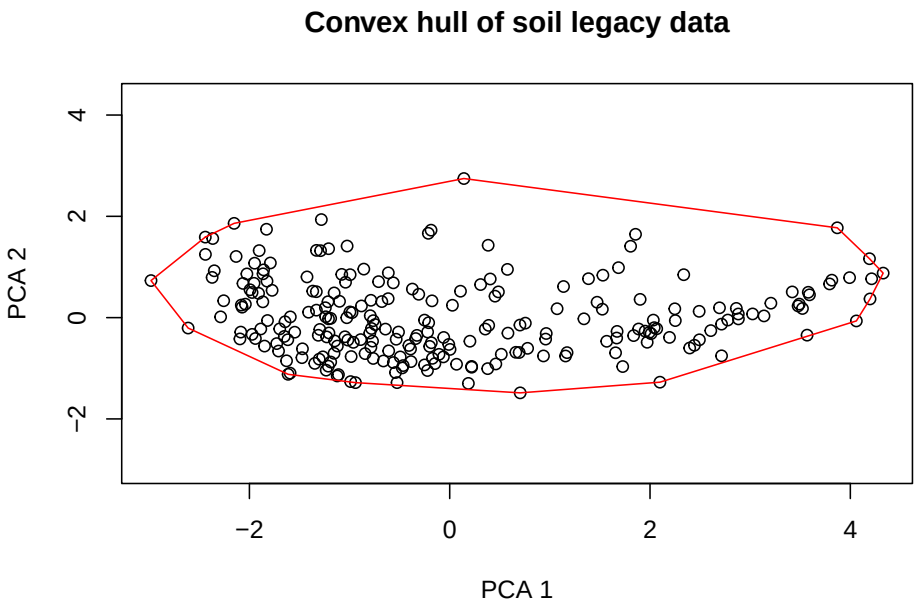
The KL-divergence is always greater than or equal to zero, and reaches its minimum value (zero) only when P and Q are identical. Thus, lower values of KL-divergence are indicative of a good match between both the sample and the study area spaces, indicating that the sample space is a fair representation of

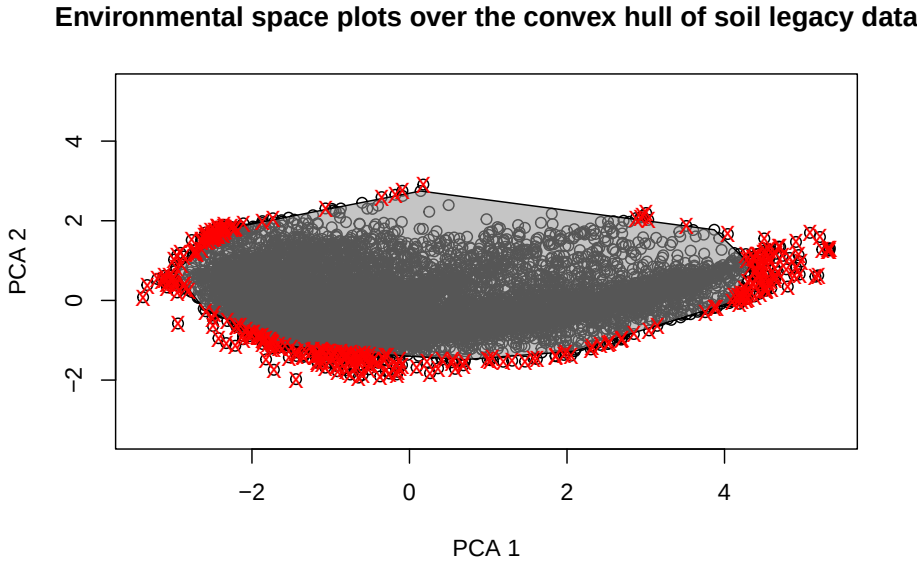
the environmental conditions in the study area.

In this case, the KL-divergence value is 0.039, indicating that the legacy samples capture most of the environmental variability in the study area.

- **Percent of representativeness in relation to the overall environmental conditions**

Finally, we can also determine the degree in which our legacy soil dataset is representative of the existing environmental conditions in the study area. For that, we calculate the proportion of pixels in the study area that would fall within the convex hull polygon delineated upon the environmental conditions found at the soil legacy data locations only. The convex hull polygon is created upon a Principal Component transformation of the covariate data in the soil legacy data and using the outter limits of the scores of the points projected on the two main Components.






```
## [1] 96.50188
```

This indicates that 96.5% of the existing conditions in the study area fall within the convex hull delineated with the data in the soil samples, showing the adequacy of the proposed legacy data for DSM.

Part I

Part two - Soil Sampling Design

Chapter 3

Creating a sampling design

3.1 Determining the optimal sample size

Several strategies exist for designing soil sampling, including regular, random, and stratified sampling. Each strategy comes with its own set of advantages and limitations, which must be carefully considered before commencing a soil sampling campaign. Regular sampling, also called grid sampling, is straightforward and ensures uniform coverage, making it suitable for spatial analysis and detecting trends. However, it may introduce bias and miss small-scale variability. Generally, random sampling may require a larger number of samples to accurately capture soil variability compared to stratified sampling, which is more targeted. Nonetheless, from a statistical standpoint, random sampling is often preferred. It effectively minimizes selection bias by giving every part of the study area an equal chance of being selected. This approach yields a sample that is truly representative of the entire population, leading to more accurate, broadly applicable conclusions. Random sampling also supports valid statistical inferences, ensures reliability of results, and simplifies the estimation of errors, thereby facilitating a broad spectrum of statistical analyses.

The determination of both the number and locations of soil samples is an important element in the success of any sampling campaign. The chosen strategy directly influences the representativeness and accuracy of the soil data collected, which in turn impacts the quality of the conclusions drawn from the study.

In this exercise, we make use of the data provided by (Malone, Minansy and Brungard, 2019) with 4 raster covariates in a 100 has area. We want to determine the minimal number of soil samples that must be collated to capture at least the 95% of variability within the environmental covariates. The procedure start with random distribution of a low number of samples in the area, determine the values of the spatial covariates, and compare them with those representing the

whole diversity in the area at pixel scale. The comparisons are made using the `'KL divergence'` and the `'% of representativeness'` - i.e. the variability of covariate information in the complete area related to the variability of covariate information in the dataset of samples. Further information can be found in the original work of (Malone, Minansy and Brungard, 2019).

The initial section of the script is related to setup options in the methodology. We load of R packages, define the working directory, load covariate data and store it as `SpatRaster` object. Here, parameter related to further aspects of the analyses such as the initial and final number of samples, and the increment step are defined.

```
## Load raster covariate data----
# Read Spatial data covariates as rasters with terra
rasters <- "data/rasters"
cov.dat <-  list.files(rasters, pattern = "tif$",  recursive = TRUE, full.names = TRU
cov.dat <- terra::rast(cov.dat)
```

We can see the covariates in a plot.

```
plot(cov.dat)
```

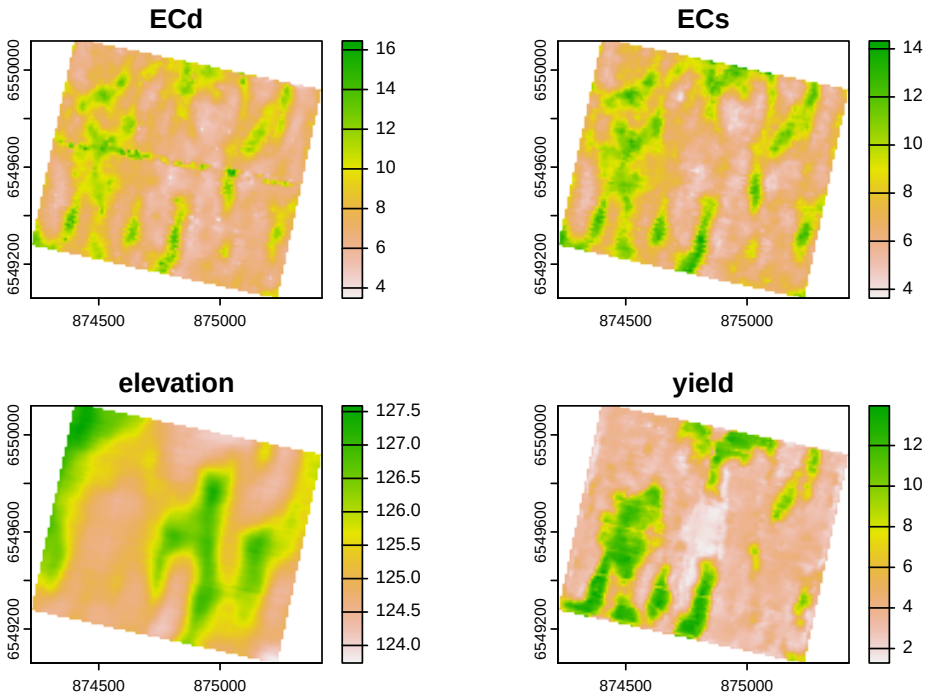


Figure 3.1: Plot of the covariates

```
  # Define the number of samples to be tested in a loop (from initial to final) and the step of t
  initial.n <- 10
  final.n <- 300
  by.n <- 10
  iters <- 10
```

The second section is where the analyses of divergence and representativeness of the sampling scheme are calculated.

The analyses are performed in a loop using growing numbers of samples at each trial. Some empty vectors are defined to store the output results at each loop. At each trial of sample size `'N'`, soil samples are located randomly on the space of covariates, and 10 replicates are calculated to determine the amount inter variability in KL divergence and representativeness in the trial. The final results for each sample size correspond to the mean results obtained from each iteration at the corresponding sample size. The optimal sample size selected correspond to the minimum sample size that accounts for at least 95% of the variability of information in the covariates within the area. The optimal sampling schema proposed correspond to the random scheme at the optimal sample size with higher value of representativeness.

Figure @ref(fig:chunk-compute_n_optimal_03) shows the distribution of covariates in the sample space.

```
# Plot the polygon and all points to be checked
     plot(newScores[,1:2], xlab="PCA 1", ylab="PCA 2", xlim=c(min(newScores[,1:2]), max(newScores
          col='black', main='Environmental space plots over the convex hull of soil legacy data')
     polygon(pr_poly,col='#99999990')
    # # Plot points outside convex hull
     points(newScores[which(newScores$pip==0),1:2], col='red', pch=12, cex =1)
```

We determine the optimal sample size and plot the evaluation results.

The following figure shows the evolution of the KL divergence and % of repressentativenes with growing sample sizes. The red dot identifies the trial with the optimal sample size for the area in relation to the covariates analysed.

```
# PLot cdf and optimal sampling point
x <- xx
y <- normalized

mydata <- data.frame(x,y)
opti <- mydata[mydata$x==optimal_n,]

plot_ly(mydata,
        x = ~seq(initial.n, final.n,by.n),
        y = ~fitted(fit1),
        mode = "lines+markers",
```

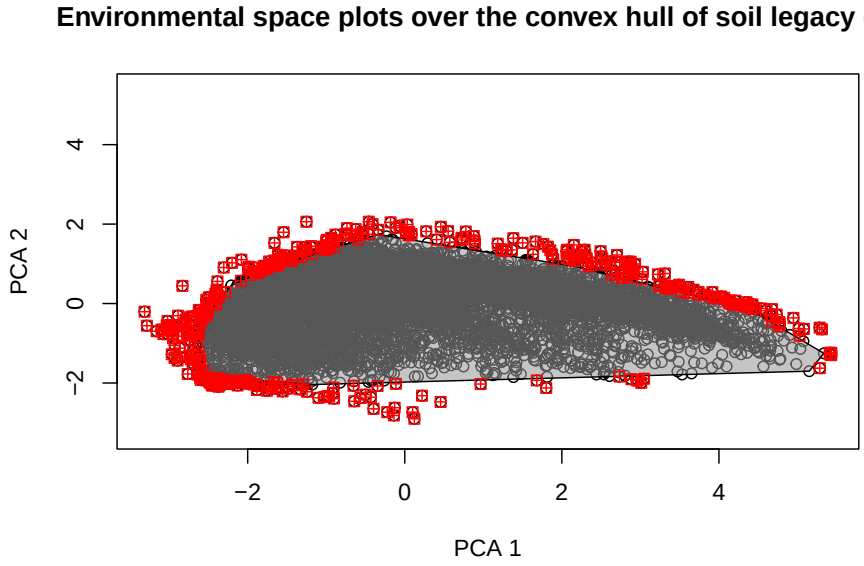


Figure 3.2: Distribution of covariates in the sample space

```
       type = "scatter",
       name = "KL divergence") %>%
add_trace(x = ~x,
          y = ~y,
          mode = "lines+markers",
          type = "scatter",
          yaxis = "y2",
          name = "Fitted representativeness")  %>%
add_trace(x = ~opti$x,
          y = ~opti$y,
          yaxis = "y2",
          mode = "markers",
          name = "Optimal N",
          marker = list(size = 8, color = '#d62728',line = list(color = 'black', wid
layout(xaxis = list(title = "N",
                    showgrid = T,
                    dtick = 50,
                    tickfont = list(size = 11)),
       yaxis = list(title = "mean KL divergence", showgrid = F ),
       yaxis2 = list(title = "Representativeness (%)",
                     overlaying = "y", side = "right"),
       legend = list(orientation = "h", y = 1.2, x = 0.1,
                     traceorder = "normal"),
       margin = list(t = 50, b = 50, r = 100, l = 80),
```

```
        hovermode = 'x')  %>%
  config(displayModeBar = FALSE)
```

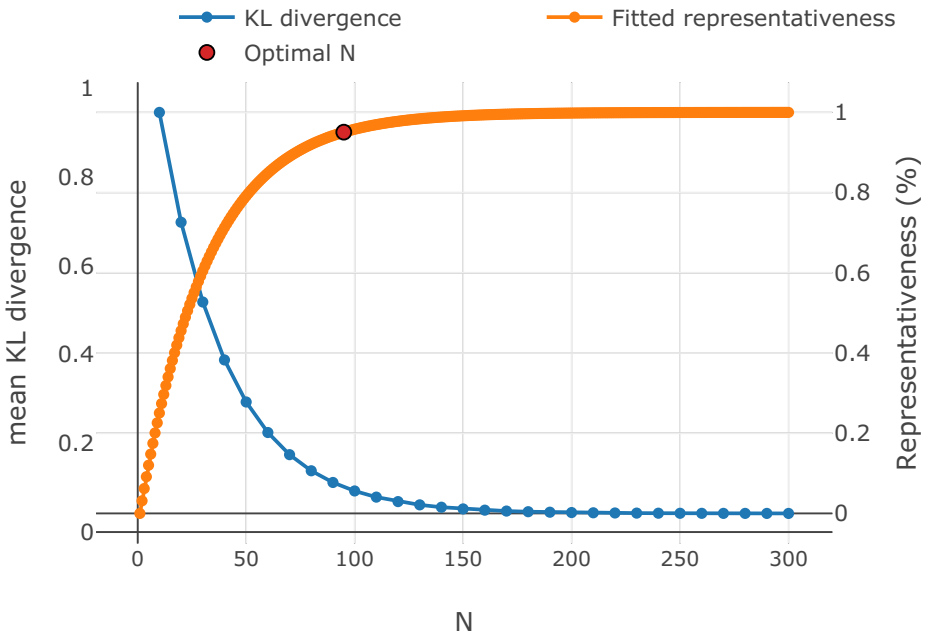


Figure 3.3: KL Divergence and Proportion of Representativeness as function of sample size

According to Figure 3.3, the optimal sampling size for the area, that captures at least 95% of the environmental variability of covariates is 95.

Finally, in Figure 3.4 we represent the optimal distribution of samples over the study area according to these specific results, and taking into account the optimal sampling size and the increasing interval in the sample size.

```
optimal_iteration <- results %>%
  filter(N==ceiling(optimal_n/by.n)*by.n) %>%
  mutate(IDX = 1:n()) %>%
  filter(Perc==max(Perc))

plot(cov.dat[[1]])
N_final <- samples_storage[paste0("N",ceiling(optimal_n/by.n)*by.n,"_", optimal_iteration$IDX)][[
points(N_final)
```

In summary, we utilize the variability within the covariate data to ascertain the minimum number of samples required to capture a minimum of 95% of this variability. Our approach involves assessing the similarities in variability between

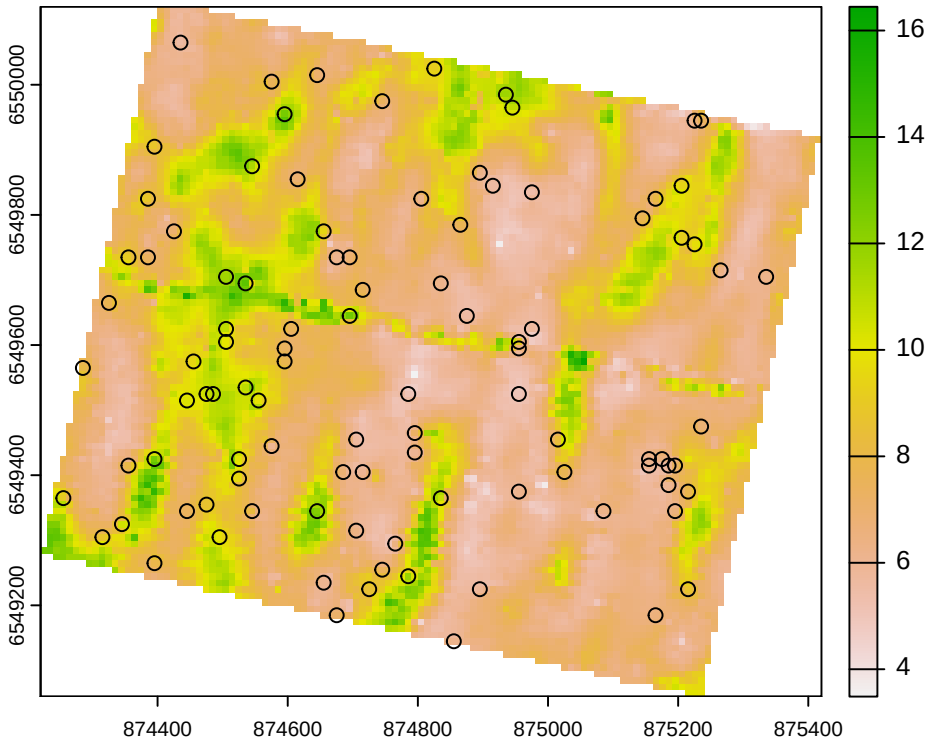


Figure 3.4: Covariates and optimal number and distribution of samples

the sample space and the population space (study area) through calculations of the Kullback-Leibler (KL) divergence and the percentage of representativeness at various stages of increasing sample sizes. These results are then utilized to fit a model representing the expected distribution of representativeness as a function of sample size. This model guides us in determining the minimum sample size necessary to achieve a representation of at least 95% of the environmental diversity within the area

Chapter 4

Stratified Sampling Design

Stratified random sampling is a technique where the study area is divided into different `groups` or `strata` based on certain environmental traits, and a number of random samples are taken from within each group. One of the primary advantages of stratified sampling is its ability to capture the diversity within a population by making sure each group is represented. It can provide a more accurate reflection of the entire population compared to random sampling, especially when the groups are distinct and have unique qualities. This approach is particularly beneficial when certain subgroups within the population are specifically noteworthy. It also allows for more precise estimates with a smaller total sample size compared to simple random choice. Stratified sampling presents some disadvantages. Achieving effective categories requires a proper definition and delineation of the initial information to create the `strata`. The classification of the environmental information into categories and ensuring fair portrayal of each can be intricate and time-taking, and mislabeling elements into an improper group can lead to skewed outcomes.

4.1 General Procedure

The creation of a stratified random sampling design involves the identification of relevant features describing the environmental diversity in the area (soil and land use are the environmental variables generally used to define strata), delineation of the strata, determination of the number of samples to distribute to each stratum, followed by random sampling within it. By identifying relevant classes, combining them to define strata, and allocating an appropriate number of samples to each stratum, a representative sample can be obtained. Random sampling within each stratum helps to ensure that the sample is unbiased and provides a fair representation of the overall conditions in the area.

The first question is about how many samples must be retrieved from each strata. The sampling scheme starts with the definition of the total number of samples to collect. In this case, the determination of the sample size is a complex and highly variable process based, among others, on the specific goals of the study, the variability of environmental proxies, the statistical requirements for accuracy and confidence, as well as additional considerations such as accessibility, costs and available resources. The optimal number of samples can be determined following the method proposed in Chapter 2 of this manual. The number of samples within each stratum is calculated using an area-weighted approach taking into account the relative area of each stratum. The sampling design in this section must also comply with the following requirements:

- All sampling strata must have a minimum size of 100 hectares.
- All sampling strata must be represented by at least 2 samples.

This sampling process ensures the representativeness of the environmental combinations present across the area while maintaining an efficient and feasible field sampling campaign.

4.1.1 Strata creation

We must determine the kind of information that will be used to construct the `strata`. In this manual, we present a simple procedure to build strata based on data from two environmental layers: soil groups and land use classification data. The information should be provided in the form of vector shapefiles with associated information databases. The data on both sets often comprises a large number of categories, that would lead to a very large number of `strata`. Thus, it is desirable to make an effort of aggregating similar categories within each input data set, to reduce, as much as possible, the number of categories while still capturing the most of the valuable variability in the area.

The fist step is to set-up the RStudio environment and load the required packages:

We must define the number of samples to distribute in the sampling design, and the soil and land use information layers to build the strata. We also define a REPLACEMENT parameter to account for a reduction of the sampling area according to a certain area using predefined bounding-box, that can be also here defined.

We proceed with the calculation of soil groups. In this example, soil information is stored in the field `TYPES`. We have analysed the extent to which the information in this field can be synthesized to eliminate redundancy when creating the `strata`. [1]

The soil classes used to build the strata are shown in Figure 4.1.

[1] This exploratory work is a prerequisite and must be adapted specifically to each soil and land use dataset

```
# Plot final map with the aggregated soil information
mapview(soil["USDA_CLASS"], alpha=0, homebutton=T, layer.name = "Soils")
```

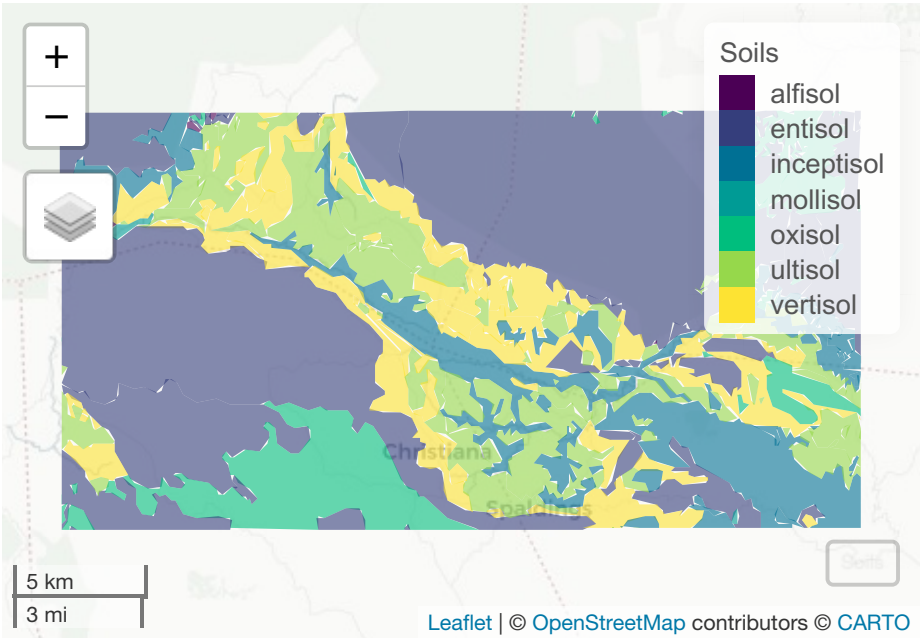


Figure 4.1: Plot of the soil classes

A similar procedure is performed on the land use dataset.

Figure 4.2 shows the landuse classes to build the strata.

```
# Plot final map with the aggregated land use information
mapview(lc["LU"], alpha=0, homebutton=T, layer.name = "Landuse")
```

To create the soil-land use `strata` we must combine both classified datasets.

```
# Combine soil and land use layers
soil_lc <- st_intersection(soil, lc)
soil_lc$soil_lc <- paste0(soil_lc$USDA_CLASS, "_", soil_lc$LU)
soil_lc <- soil_lc %>% dplyr::select(soil_lc, geometry)
```

Finally, to comply with the initial requirements of the sampling design, we calculate the areas of each polygon, delete all features with extent lesser than 100 has.

The final strata map is shown in Figure 4.3.

```
 # Plot final map of stratum
mapview(soil_lc["soil_lc"], alpha=0, homebutton=T)
```

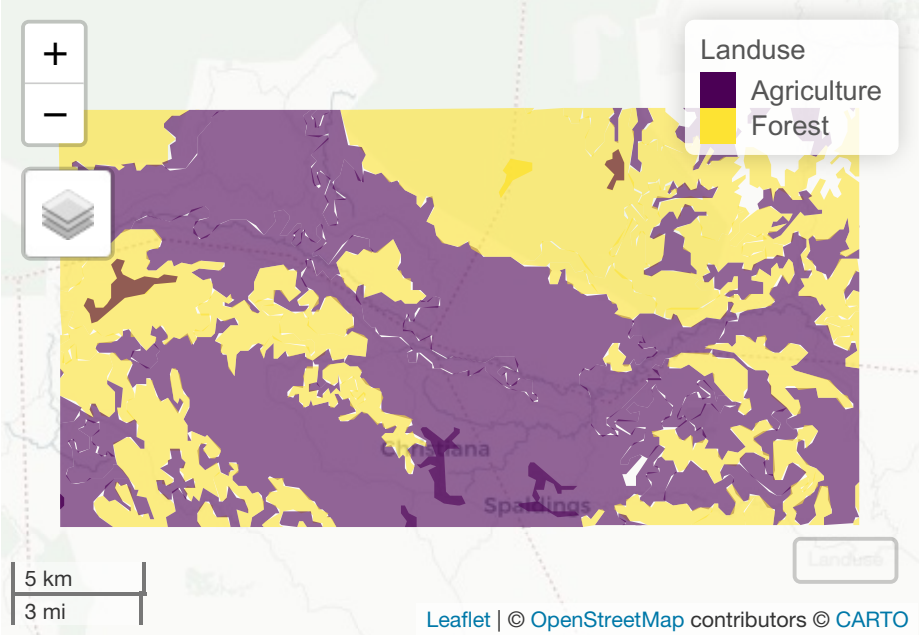


Figure 4.2: Plot of the land use classes

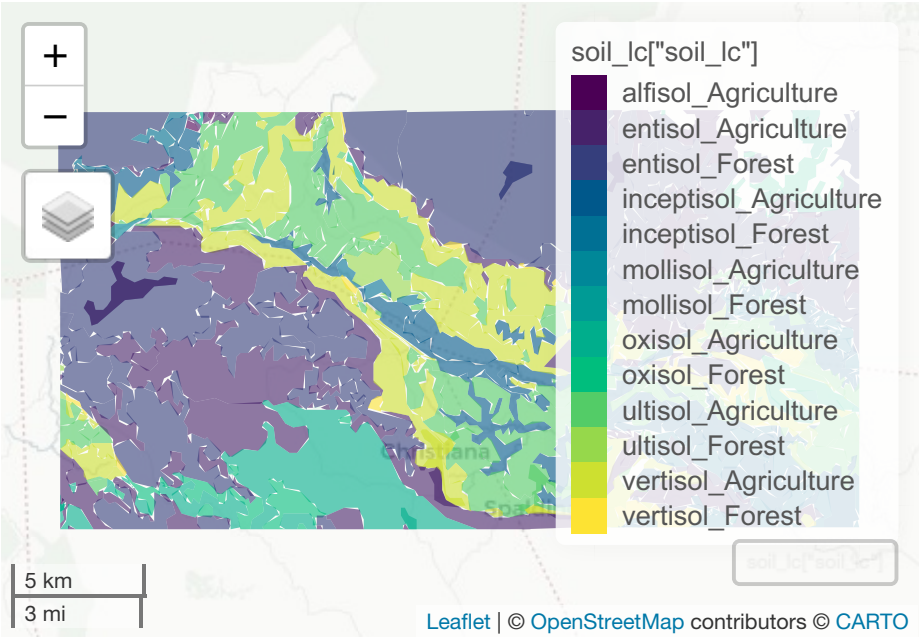


Figure 4.3: Plot of strata

```
#terra::plot(soil_lc["soil_lc"], border=NA, main="Strata classes")
```

4.1.2 Stratified random sampling

```
##soil_lc <- st_read("../soil_sampling/JAM/strata_diss.shp")
#soil_lc <- st_cast(soil_lc, 'POLYGON')
#target <- st_read("../soil_sampling/JAM/sampling_points.shp")
#target <- target[target$type=="Target",]
# Plot final map with the aggregated land use information
mapview(soil_lc["soil_lc"], alpha=0, homebutton=T, layer.name = "Strata") + mapview(z[z$type=='
```

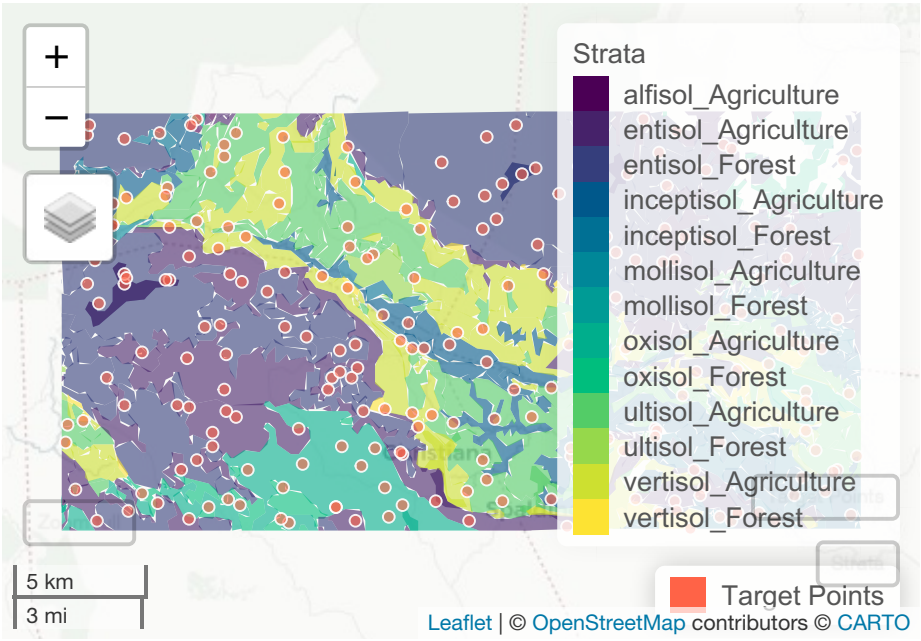


Figure 4.4: Plot of strata and sampling points

```
#terra::plot(soil_lc["soil_lc"], border=NA, main="Strata classes")
#terra::points(target[target$type=="Target",])
```

4.2 Stratified random sampling for large areas

The implementation of a stratified random sampling, along with target and replacement points, can present operating difficulties when dealing with areas of significant size and with locations that are hard to reach. To address this

issue, the sampling approach can be modified by excluding areas with limited accessibility.

This modification can streamline fieldwork operations and establish a feasible sampling method while still retaining the essence of the stratified random sampling framework. By excluding areas with limited accessibility, the sampling design can be adjusted to ensure a more practical and effective approach to data collection.

- **Delineation of sampling accessibility:** The sampling area can be further limited based on accessibility considerations. Areas with very limited accessibility, defined as regions located more than 1 kilometre away from a main road or access path, may be excluded from sampling areas. To accomplish this, a map of main roads and paths can be used to establish a sampling buffer that includes areas within a 1-kilometre buffer around the road infrastructures. This exclusion helps to eliminate the most remote and challenging-to-access areas. An additional layer of accessibility information can be incorporated based on population distribution in the country, considering that, if population is present, there is a high change that points in the surroundings can be accessible for sampling. In this case, populated nuclei are vectorized into points, and a 250-meter buffer is then generated around each point. These resulting areas can be then added to the 1-kilometre buffer around the roads, which collectively defined the final sampling area.
- **Substitution of replacement points with replacement areas in close proximity to the target points:** The sampling design presented before included designated replacement points to serve as substitutes for each target point in the case that it would be inaccessible during fieldwork. However, this approach presented challenges, particularly for large areas, as the replacement point could be located far from the target point, resulting in significant logistical efforts. This limitation posed a risk of delays in completing the sampling campaign within the allocated time frame. To address this challenge, an alternative strategy is to replace the idea of replacement points with replacement areas situated in the immediate vicinity of the target point. The replacement area for each target point is now confined within a 500-meter buffer surrounding the target and falls within the same sampling stratum. This approach concentrates sampling and replacement activities within a specific geographic area, streamlining the overall process. By reducing the need for extensive travel, this method enhances efficiency and facilitates sample collection. Figure 2 illustrates the distribution of sampling points and replacement areas for visualization.
- **Additional area exclusion:** Some areas can be identified as not suitable for sampling purposes. This is the case of certain natural protected areas, conflict regions presenting risks for field operators, etc. These areas must be identified masked at an initial stage of the design to exclude them from the sampling strata.

The procedure is the same as that previously presented, with the difference that buffers and exclusion areas must be masked-out from the strata map before performing the random-sampling.

```
# Compute sampling areas WITH REPLACEMENT -----
 if(REPLACEMENT){
     # Load strata
     soil_lc <- st_read("../soil_sampling/JAM/strata.shp")

     # Read sampling. points from previous step
     z <- st_read("../soil_sampling/JAM/sampling_points.shp")

     # Define buffer of 500 meters (coordinate system must be in metric base)
     samples.buffer = 500
     buf.samples <- st_buffer(z, dist=samples.buffer)

     # Intersect buffers
     samples_buffer = st_intersection(soil_lc, buf.samples)
     samples_buffer <- samples_buffer[samples_buffer$type=="Target",]
     samples_buffer <- samples_buffer[samples_buffer$soil_lc==samples_buffer$group,]
     # Save Sampling areas
     st_write(samples_buffer, paste0('../soil_sampling/JAM/replacement_areas_', samples.buffer,

     # Write target points only
     targets <- z[z$type=="Target",]
     st_write(targets, '../soil_sampling/JAM/sampling_points_TAR.shp', delete_dsn = TRUE)
 }
```

4.3 Stratified regular sampling

The procedure for creating a stratified regular sampling design is identical to that presented for stratified random sampling, with the only distinction that the locations of the sampling points are distributed in a regular spatial grid. This transformation is achieved by changing the method from 'random' to 'regular' in the spatSample functions within the script above.

```
  mapview(soil_lc["soil_lc"], alpha=0, homebutton=T, layer.name = "Strata") + mapview(z[z$type=='
```

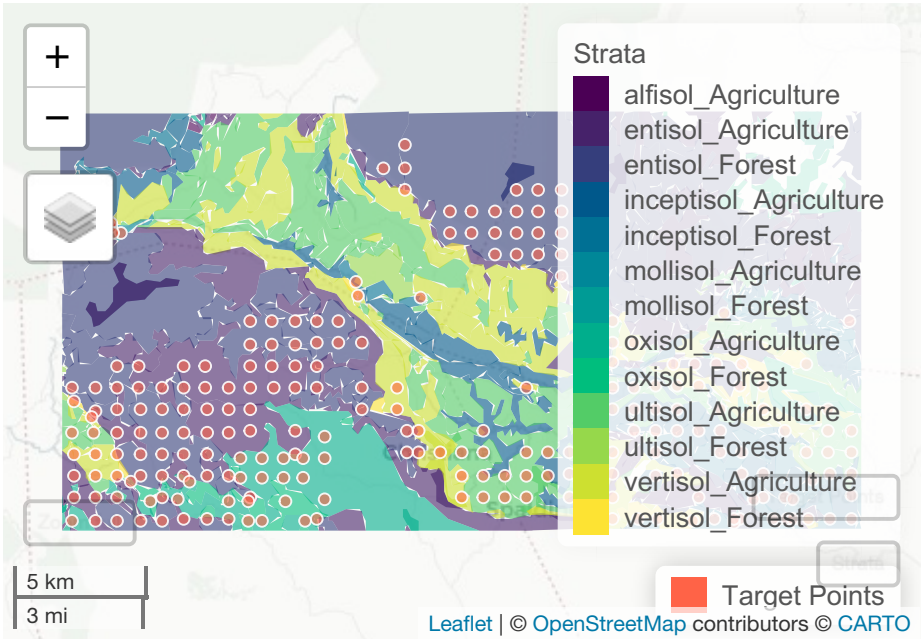


Figure 4.5: Plot of strata and regular sampling points

References

Clifford, D., Payne, J.E., Pringle, M.J., Searle, R. & Butler, N. 2014. Pragmatic soil survey design using flexible latin hypercube sampling. *Computers & Geosciences*, 67: 62–68. https://doi.org/10.1016/j.cageo.2014.03.005

Malone, B.P., Minansy, B. & Brungard, C. 2019. Some methods to improve the utility of conditioned latin hypercube sampling. *PeerJ*, 7: e6451. https://doi.org/10.7717/peerj.6451

Minasny, B. & Mcbratney, A. 2006. A conditioned latin hypercube method for sampling in the presence of ancillary information. *Computers & Geosciences*, 32: 1378–1388. https://doi.org/10.1016/j.cageo.2005.12.009

Sena, N.C., Veloso, G.V., Lopes, A.O., Francelino, M.R., Fernandes-Filho, E.I., Senra, E.O., Silva Filho, L.A. da, Condé, V.F., Arruda Silva, D.L. de & Araújo, R.W. de. 2021. Soil sampling strategy in areas of difficult acess using the cLHS method. *Geoderma Regional*, 24: e00354. https://doi.org/10.1016/j.geodrs.2020.e00354