Using digital soil mapping to update, harmonize and disaggregate legacy soil maps.

Soil Security Laboratory

2018

Digital soil maps are contrasted from legacy soil maps mainly in terms of the underlying spatial data model. Digital soil maps are based on the pixel data model, while legacy soil maps will typically consist of a tessellation of polygons. The advantage of the pixel model is that the information is spatially explicit. The soil map polygons are delineations of soil mapping units which consist of a defined assemblage of soil classes assumed to exist in more-or-less fixed proportions. There is great value in legacy soil mapping because a huge amount of expertise and resources went into their creation. Digital soil mapping will be the richer by using this existing knowledge-base to derive detailed and high resolution digital soil infrastructures. However the digitization of legacy soil maps is not digital soil mapping. Rather, the incorporation of legacy soil maps into a digital soil mapping workflow involves some method (usually quantitative) of data mining, to appoint spatially explicit soil information — usually a soil class or even a measurable soil attribute — upon a grid the covers the extent of the existing (legacy) mapping. In some ways, this process is akin to downscaling because there is a need to extract soil class or attribute information from aggregated soil mapping units. A better term therefore is soil map disaggregation.

There is an underlying spatial explicitness in digital soil mapping that makes it a powerful medium to portray spatial information. Legacy soil maps also have an underlying spatial model in terms of the delineation of geographical space. However, there is often some subjectivity in the actual arrangement and final shapes of the mapping unit polygons. Yet that is a matter of discussion for another time. For disaggregation studies the biggest impediment to overcome in a quantitative manner is to determine the spatial configuration of the soil classes within each map unit. It is often known which soil classes are in each mapping unit, and sometimes there is information regarding the relative proportions of each too. What is unknown is the spatial explicitness and configuration of said soil classes within the unit. This is the common issue faced in studies seeking the renewal and updating of legacy soil mapping. Some examples of soil map disaggregation studies from the literature include Thompson et al. (2010) who recovered soil-landscape rules from a soil map report in order to map individual soil classes. This together with a supervised classification approach described by Nauman et al. (2012) represent manually-based approaches to soil map disaggregation. Both of these studies were successfully applied, but because of their manual nature, could also be

seen as time-inefficient and susceptible to subjectivity. The flip side to these studies is those using quantitative models. Usually the modeling involves some form of data mining algorithm where knowledge is learned and subsequently optimized based on some model error minimization criteria. Extrapolation of the fitted model is then executed in order to map the disaggregated soil mapping units. Such model-based or data mining procedures for soil map disaggregation include that by Bui and Moran (2001) in Australia, Haring et al. (2012) in Germany and Nauman and Thompson (2014) in the USA. Some fundamental ideas of soil map disaggregation framed in a deeper discussion of scaling of soil information are presented in McBratney (1998).

This chapter seeks to describe a soil map disaggregation method that was first described in Wei et al. (2010) for digital harmonization of adjacent soil surveys in southern Iowa, USA. The concept of harmonization has particular relevance in the USA because it has been long established that the underlying soil mapping concepts across geopolitical boundaries (i.e. counties and states) don't always match. This issue is obviously not a phenomenon exclusive to the USA but is a common worldwide issue. This mismatch may include the line drawings and named map units. Of course, soils in the field do not change at these political boundaries. These soil-to-soil mismatches are the result of the past structuring of the soil survey program. For example, soil surveys in the US were conducted on a soil survey area basis. Most times the soil surveys areas were based on county boundaries. Often adjacent counties were mapped years apart. Different personnel, different philosophies of soil survey science, new concepts of mapping and the availability of various technologies all have played a part in why these differences occur. These differences maybe even more exaggerated at state lines as each state was administratively and technically responsible for the soil survey program within a given state. The algorithm developed by Wei et al. (2010) addressed this issue, where soil mapping units were disaggregated into soil series. Instead of mapping the prediction of a single soil series, a probability distribution of all potential soil series was estimated. The outcome of this was the dual disaggregation and harmonization of existing legacy soil information into raster-based digital soil mapping product/s.

Odgers et al. (2014) using legacy soil mapping from an area in Queensland, Australia refined the algorithm to which they called DSMART or, Disaggregation and Harmonization of Soil Map Units Through Re-sampled Classification Trees. Besides the work of Odgers et al. (2014), The DSMART algorithm has been used is other projects throughout the world, with Chaney et al. (2014) using it to disaggregate the entire gridded USA Soil Survey Geographic (gSSURGO) database. The resulting POLARIS data set (Probabilistic Remapping of SSURGO) provides the 50 most probable soil series predictions at each 30-meter grid cell over the contiguous USA. DSMART has also been a critical component for the development of the Soil and Landscape Grid of Australia (SLGA) data set (Grundy et al., 2015). The SLGA is the first continental version of the GlobalSoilMap.net concept and the first nationally consistent, fine spatial resolution set of continuous soil attributes with Australia-wide coverage. The DSMART algorithm has been pivotal, together with the associated PROPR algorithm (Digital Soil Property

Mapping Using Soil Class Probability Rasters; Odgers et al. (2015)) in deriving high resolution digital soil maps where point-based DSM approaches cannot be undertaken, particularly where soil point data is sparse. In this chapter, the fundamental features of DSMART are described, followed its demonstration upon a small data set.

1 DSMART: an overview

Odgers et al. (2014) provide a detailed explanation of the DSMART algorithm. The aim of DSMART is to predict the spatial distribution of soil classes by disaggregating the soil map units of a soil polygon map. Here soil map units soil map are entities consisting of a defined set of soil classes which occur together in a certain spatial pattern and in an assumed set of proportions. The DSMART method of representing the disaggregated soil class distribution is as a set of numerical raster surfaces, with one raster per soil class. The data representation for each soil class is given as the probability of occurrence. In order to generate the probability surfaces, a re-sampling approach is used to generate n realizations of the potential soil class distribution within each map unit. Then at each grid cell, the probability of occurrence of each soil class is estimated by the proportion of times the grid cell is predicted as each soil class across the set of realizations. The procedure of the DSMART algorithm can be summarized in 6 main steps:

1. Draw n random samples from each soil map polygon.
2. Assign soil class to each sampling point.
    - Weighted random allocation from soil classes in relevant map unit
    - Relative proportions of soil classes within map units are used as the weights
3. Use sampling points and intersected covariate values to build a decision tree to prediction spatial distribution of soil classes.
4. Apply decision tree across mapping extent using covariate layers
5. Steps 1-4 repeated i times to produce i realizations of soil class distribution.
6. Using i realizations generate probability surfaces for each soil class.

The model type that Odgers et al. (2014) used was the C4.5 decision tree algorithm which was introduced by Quinlan (1993). The type of data mining algorithm implemented in DSMART is not prescriptive; as long as it is robust and importantly, computationally efficient. For example Chaney et al. (2014) used Random Forest models (Breiman, 2001) in their implementation of DSMART.

2 Implementation of DSMART

The DSMART algorithm has previously been written in the C++ and Python computing languages. It is also available in an `R` package, which was developed at the Soil Security Laboratory. Regardless of computing language preference, DSMART requires three chief sources of data:

1. The soil map unit polygons that will be disaggregated.
2. Information about the soil class composition of the soil map unit polygons
3. Geo-referenced raster covariates representing the *scorpan* factors of which have complete and continuous coverage of the mapping extent. There is no restriction in terms of the data type i.e. continuous, categorical, ordinal etc.

2.1 DSMART with `R`

The DSMART algorithm is packaged up in **`rdsmart`** and contains two working functions: **`disaggregate`** and **`summarise`**. More will be discussed about these shortly. The other items in the package are various inputs required to run the function. In essence these data provide some indication of the structure and nature of the information that is required to run the DSMART algorithm so that it can be easily adapted to other projects. First is the soil map to be disaggregated. This is saved to the **`dalrymple_polygons`** object. In this example the small polygon map is a clipped area of the much larger soil map that Odgers et al. (2014) disaggregated, which was the 1:250,000 soil map of the Dalrymple Shire in Queensland Australia by Rogers et al. (1999).In this example data set, there are 11 soil mapping units.the polygon object is of class **`SpatialPolygonsDataFrame`**, which is what would be created if you were to read in a shapefile of the polygons into `R` (Figure 1).

```
# To install rdsmart you need: library(devtools)
# install_bitbucket('brendo1001/dsmart/rPackage/dsmart/pkg')
library(rdsmart)
library(sp)
library(raster)

# Polygons
data("dalrymple_polygons")
class(dalrymple_polygons)
```

```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
summary(dalrymple_polygons$MAP_CODE)
```

```
## BUGA1t CGCO3t   DO3n   FL3d   HG2g   MI6t   MM5g   MS4g   PA1f   RA3t
##      1      1      1      1      1      1      1      1      1      1
## SCFL3g
```

```
##      1
```

```
plot(dalrymple_polygons)
invisible(text(getSpPPolygonsLabptSlots(dalrymple_polygons),
labels = as.character(dalrymple_polygons$MAP_CODE), cex = 1))
```

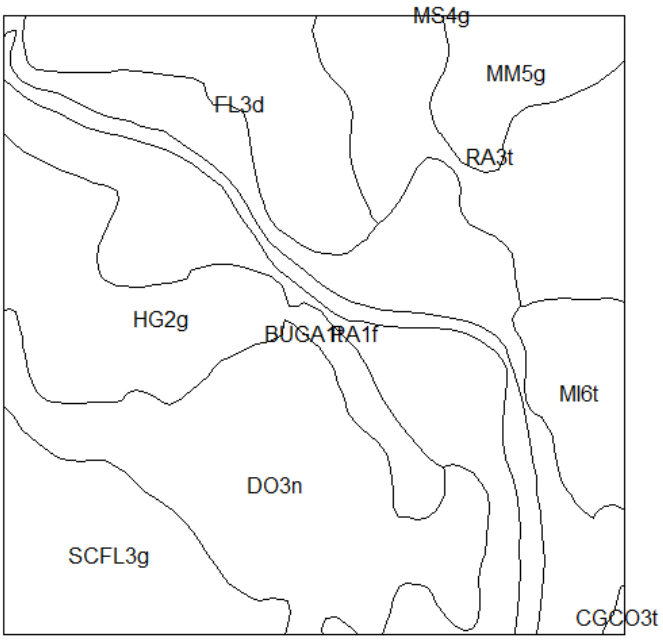

Figure 1: Subset of the polygon soil map from the Dalrymple Shire, Queensland Australia which was disaggregated by Odgers et al. (2014).

The next inputs are the soil map unit compositions which is saved to the `dalrymple_composition` object. This is a data frame that simply indicates in respective columns the map unit name, and corresponding numerical identifier label. Then there is the soil classes that are contained in the respective mapping unit, followed by the relative proportion that each soil class contributes to the map unit. The relative proportions will and probably should sum to 100.

```
# Map unit compositions
data("dalrymple_composition")
head(dalrymple_composition)
```

```
##   poly mapunit soil_class proportion
## 1  304    MM5g         RA         70
## 2  304    MM5g         EW         20
## 3  304    MM5g         PI         10
## 4  440  CGCO3t         CG         50
## 5  440  CGCO3t         CO         20
## 6  440  CGCO3t         DA         10
```

The last required inputs are the environmental covariates. This use used to inform the model fitting for each DSMART iteration, and ultimately be used for the spatial mapping. There are actually 20 different covariate rasters of which have been derived from a digital elevation model and gamma radiometric data. These rasters are organized into a `RasterStack` and are of 30m grid resolution. This class of data is the necessary format of the covariate data for input into DSMART.

```
# covariates
data("dalrymple_covariates")
class(dalrymple_covariates)
```

```
## [1] "RasterStack"
## attr(,"package")
## [1] "raster"
```

```
nlayers(dalrymple_covariates)
```

```
## [1] 20
```

```
res(dalrymple_covariates)
```

```
## [1] 30 30
```

Now it is time to run the DSMART algorithm. The actual `R` implementation is spread across two companion functions already mentioned: `disaggregate` and `summarise`. The `disaggregate` function is the workhorse of the two because it performs the sample/resampling, model fitting and iteration parts of DSMART. The `summarise` function works on the outputs of `disaggregate` to estimate the probabilities of classes, and derives some other useful outputs such as the most probable soil class and or n-most probable soil classes. Using the `disaggregate` function, we provide it with the inputs described above. Additional inputs include the parameters `rate`, which is a numeric value for the number of samples to take from each soil mapping polygon; `reals` is the number of model realizations to fit; and `cpus` is the number of compute nodes to use for the analysis. The default is to run the algorithm in sequential mode, however it does have the capability to be scaled up substantially in parallel mode, which helps to eliminate some computation time. In the example below we set `rate` to 15, `reals` to 10, and `cpus` to 3. An unusual feature of this function is that none of the output is saved to the `R` memory, but instead, directly to file, or specifically the current working directory into a folder called `outputs`. After the `disaggregate` function has terminated, you will find in the `outputs` folder a few other folders which contain rasters of the soil class prediction from each iteration. You will also encounter another folder contain text file outputs from each iteration of the C5 model structure plus

information on the quality of the fit.

```
library(parallel)
# Run disaggregate without adding observations
test.dsmart <- rdsmart::disaggregate(covariates = dalrymple_covariates,
 polygons = dalrymple_polygons,
    composition = dalrymple_composition, rate = 15, reals = 10, cpus = 3)
```

The `disaggregate` function has some added functionality that is not used in the above example. The help file for the function describes the added functionality which includes the allowance of observed data into the algorithm. The is also provision of different methods for sampling polygons. There is also an ability to include additional environmental variables by way of `strata` which constrains the occurrences of soil classes to certain areas and or positions within the landscape.

Of particular interest is to derive the soil class probabilities, and even the most probable soil class at each pixel, and even an estimate of the uncertainty, which is given in terms of the confusion index that was used earlier during the fuzzy classification of data for derivation of digital soil map uncertainties. The confusion index essentially measure how similar to classification is between (in most cases) most probable and second-most probable soil class predictions at a pixel. To derive the probability rasters we need the rasters that were generated from the `disaggregate` function. This can be done via the use of the `list.files` function and the `raster` package to read in the rasters and stack them into a `rasterStack`. Rather than doing this we can use pre-prepared outputs namely in the form of the `dalrymple_realisations` (raster outputs from `disaggregate`) and `dalrymple_lookup` (lookup table of soil class names and numeric counterparts which is an output from `disaggregate`) objects. As with `disaggregate`, `summarise` can be run in parallel mode via control of the `cpus` variable. The n-most probable rasters are also created by `summarise`, and are important for the follow on procedure of soil attribute mapping, of which is the focus of the study by Odgers et al. (2015) and integral to the associated PROPR algorithm. In many cases the user may just be interested in deriving the most probable soil class, or sometimes the n-most probable soil class maps. The `nprob` parameter provides user control on the number of n-probable outputs.

```
# run summarise run function getting most 3 most probable and creating
# probability rasters using 2 compute cores.
data(dalrymple_lookup)
data(dalrymple_realisations)
test.dsmart_s <- summarise(realisations = dalrymple_realisations,
lookup = dalrymple_lookup, nprob = 3, cpus = 2)
```

A few folders are automatically generated to the working directory which contain various outputs from the `summarise` function. These are: `mostprobable` and `probabilities`. The folder `mostprobable` contains the n-most probable soil class maps. A confusion index (Burrough et al., 1997) raster is returned. The folder: `probabilities`, contains the probability rasters for each candidate soil class. So as a final step, lets produce a map of the most probable soil class (Figure 2) , and the associated map of the

confusion index (Figure 3).

```
# plot most probable soil class
library(rasterVis)
ml.map <- raster("C:/Users/bmalone/Documents/output/mostprobable/mostprob_01_class.tif")
ml.map
ml.map <- as.factor(ml.map)
rat <- levels(ml.map)[[1]]
rat[["class"]] <- c("BL", "BU", "BW", "CE", "CG", "CK", "CO", "CP", "DA", "DO",
    "EW", "FL", "FR", "GA", "GR", "HG", "PA", "PI", "RA", "SC")
levels(ml.map) <- rat

# Randomly selected HEX colors
area_colors <- c("#9a10a8", "#cf68a0", "#e7cc15", "#4f043a", "#1129a3", "#a2d0a7",
    "#7b0687", "#3e28d1", "#2c8c04", "#d39014", "#66a5ed", "#978279", "#db6f1f",
    "#4070fc", "#fde864", "#acdb0a", "#d95a28", "#94561f", "#162972", "#8342e1")
levelplot(ml.map, col.regions = area_colors, xlab = "", ylab = "",
main = "Most probable soil class")

# Confusion Index
CI.map <- raster("C:/Users/bmalone/Documents/output/mostprobable/confusion.tif")
plot(CI.map)
```

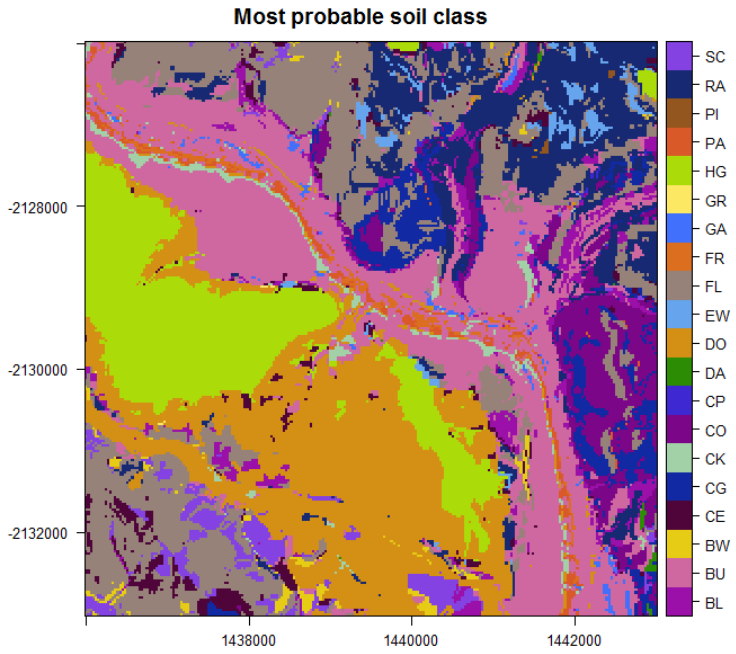

Figure 2: Map of the most probable soil class from DSMART.

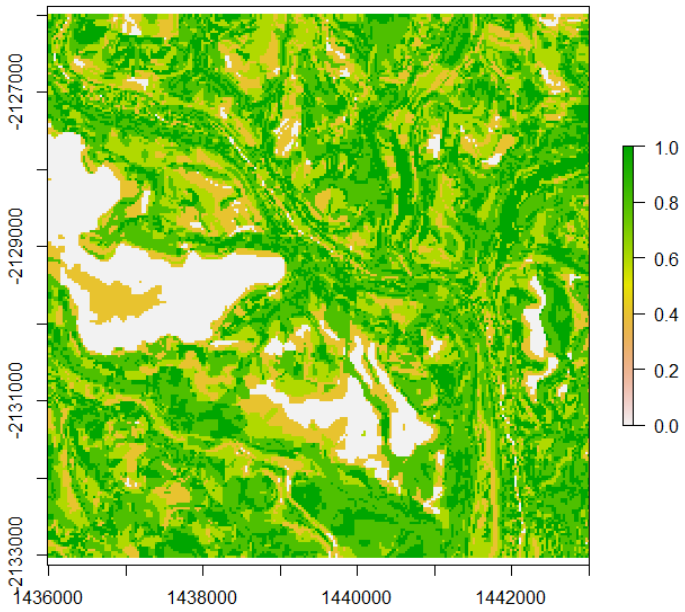

Figure 3: Confusion index of soil class predictions from DSMART.

DSMART can be quite a powerful algorithm for disaggregating legacy soil mapping as demonstrated by Chaney et al. (2014) in the USA. While the computational effort to generate disaggregated predictions could be a burden, the DSMART algorithm is relatively easy to parallelize, which was also demonstrated in Chaney et al. (2014), and with the implementation of the current `R` version of this algorithm. One other restrictive feature of the DSMART algorithm is the need for specific inputs, particularly in regards to the soil class compositions and their relative proportions within mapping units. Sometimes this information is not easily available, and needs to be approximated by some means.

Besides to advantage of generating detailed maps of soil classes, which will have their own use for some applications, the DSMART algorithm provides a pathway to first update and harmonize legacy soil maps, and then to realize soil property information from existing polygon soil maps, such as through the use and coupling of soil class probability rasters and modal soil class profiles as demonstrated in Odgers et al. (2015). It is this detailed soil attribute information that is required in land system modeling frameworks, and for ongoing assessment and monitoring of the soil resource.

References

Breiman, L.
2001. Random forests. *Machine Learning*, 41:5–32.

Bui, E. and C. Moran
2001. Disaggregation of polygons of surficial geology and soil maps using spatial modelling and legacy data. *Geoderma*, 103:79–94.

Burrough, P. A., P. F. M. van Gaans, and R. Hootsmans
1997. Continuous classifcation in soil survey: spatial correlation, confusion and boundaries. *Geoderma*, 77:115–135.

Chaney, N., J. W. Hempel, N. P. Odgers, A. B. McBratney, and E. F. Wood
2014. Spatial disaggregation and harmonization of gssurgo. In *ASA, CSSA and SSSA International Annual meeting*, Long Beach, CA. ASA, CSSA and SSSA.

Grundy, M. J., R. Viscarra Rossel, R. D. Searle, P. L. Wilson, C. Chen, and L. J. Gregory
2015. Soil and landscape grid of australia. *Soil Research*,
P. http://dx.doi.org/10.1071/SR15191.

Haring, T., E. Dietz, S. Osenstetter, T. Koschitzki, and B. Schroder
2012. Spatial disaggregation of complex soil map units: a decision-tree based approach in bavarian forest soils. *Geoderma*, Geoderma 185?186, 37?47.:37–47.

McBratney, A.
1998. Some considerations on methods for spatially aggregating and disaggregating soil information. In *Soil and Water Quality at Different Scales*, P. Finke, J. Bouma, and M. Hoosbeek, eds., volume 80 of *Developments in Plant and Soil Sciences*, Pp. 51–62. Springer Netherlands.

Nauman, T. W. and J. A. Thompson
2014. Semi-automated disaggregation of conventional soil maps using knowledge driven data mining and classification trees. *Geoderma*, 213:385 – 399.

Nauman, T. W., J. A. Thompson, N. P. Odgers, and Z. Libohova
2012. *Digital Soil Assessments and Beyond: Proceedings of the Fifth Global Workshop on Digital Soil Mapping*, chapter Fuzzy disaggregation of conventional soil maps using database knowledge extraction to produce soil property maps., Pp. 203–207. CRC Press, London, UK.

Odgers, N. P., A. B. McBratney, and B. Minasny
2015. Digital soil property mapping and uncertainty estimation using soil class probability rasters. *Geoderma*, 237-238:190–198.

Odgers, N. P., W. Sun, A. B. McBratney, B. Minasny, and D. Clifford
2014. Disaggregating and harmonising soil map units through resampled classification trees. *Geoderma*, 214-215:91–100.

Quinlan, J. R.

1993. *C4.5: Programs for Machine Learning.* Morgan Kaufmann, San Mateo, California.

Rogers, L., M. Cannon, and E. Barry
1999. *Land Resources of the Dalrymple Shire, 1. Land Resources Bulletin DNRQ980090.* Brisbane, Australia: Queensland Department of Natural Resources.

Thompson, J. A., T. Prescott, A. C. Moore, J. Bell, D. R. Kautz, J. W. Hempel, S. W. Waltman, and C. Perry
2010. Regional approach to soil property mapping using legacy data and spatial disaggregation techniques. In *19th World Congress of Soil Science*, Brisbane Australia. IUSS.

Wei, S., A. McBratney, J. Hempel, B. Minasny, B. Malone, T. D'Avello, L. Burras, and J. Thompson
2010. Digital harmonisation of adjacent analogue soil survey areas - 4 iowa counties. In *19th World Congress of Soil Science*, Brisbane, Australia. IUSS.