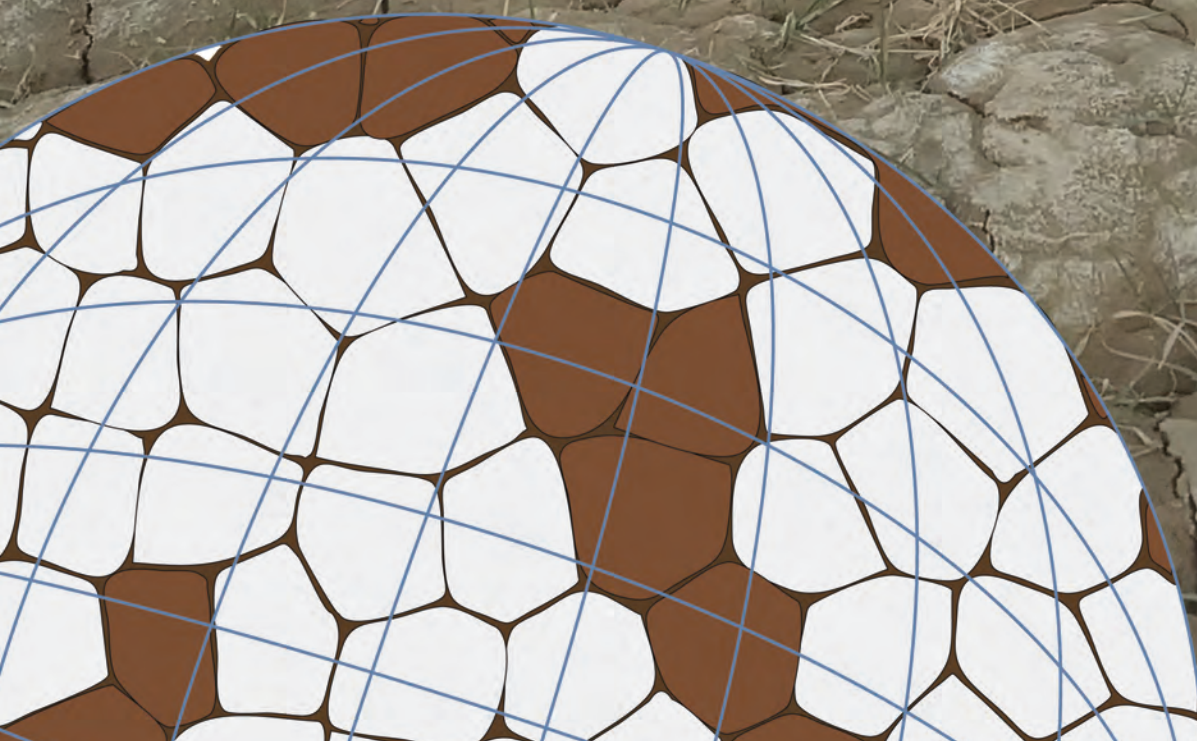




Food and Agriculture  
Organization of the  
United Nations

# **MAPPING OF SALT-AFFECTED SOILS**

## **LESSON 4** **Spatial modelling of salt-affected soils**





## Mapping of salt-affected soils: Lesson 4 – Spatial modelling of salt-affected soils

Capacity building program is part of the country-driven framework for updating national and global level information of salt-affected soils. It aims at mobilizing country-level resources and expertise to assess the status of salt-affected soils and build foundation for future monitoring and management of these soils. The program is a systematic procedure to strengthen national capacities as well as harmonize global approaches for building information of salt-affected soils. This document illustrates the fourth lesson of the capacity-building program and focuses on spatial modelling of salt-affected soils using input soil indicators.

FOOD AND AGRICULTURE ORGANIZATION OF THE UNITED NATIONS  
ROME, 2020

## Disclaimer and copyright

### **Recommended citation:**

Omuto, C.T., Vargas, R., Viatkin, K., Yigini, Y., 2020. Mapping of salt-affected soils: Lesson 4 – Spatial modelling of salt-affected soils. Rome

The designations employed and the presentation of material in this information product do not imply the expression of any opinion whatsoever on the part of the Food and Agriculture Organization of the United Nations (FAO) concerning the legal or development status of any country, territory, city or area or of its authorities, or concerning the delimitation of its frontiers or boundaries. The mention of specific companies or products of manufacturers, whether or not these have been patented, does not imply that these have been endorsed or recommended by FAO in preference to others of a similar nature that are not mentioned.

The views expressed in this information product are those of the author(s) and do not necessarily reflect the views or policies of FAO.

## Summary

This Lesson is the fourth step of the capacity-building program, which is designed to build national capacities as well as harmonize procedures for developing information of salt-affected soils at the national and global levels. The overall goal of this Lesson is to support participants in developing national maps of salt-affected soils. At the end of the lesson, the participants are expected to produce national maps of salt-affected soils in their countries.

Summary outputs from spatial modelling of salt-affected soils

Item	Description		Timeline
Input maps	<b>Input maps (between 0-30 and 30-100 cm)</b>		
	Electrical Conductivity	dS/m	
	pH	-	
	Exchangeable Sodium percent	-	
Software	R and its contributed packages		
Maps in geoTiff files	<b>Outputs</b>		2 days
	Major types of salt-affected soils	0-30 and 30-100 cm	
	Uncertainty maps of salt-affected soils	0-30 and 30-100 cm	
Accuracy assessment	<b>Accuracy indices</b>		
	Kappa index		
	Graph of confusion matrix		



## Table of Contents

Disclaimer and copyright .....	i
Summary .....	ii
List of Figures .....	iii
List of Tables .....	iii
1 Introduction .....	1
1.1 Overview .....	1
1.2 Objective .....	1
1.3 Expected outcomes.....	1
2 Activities.....	1
2.1 Classifying salt-affected soils .....	1
2.2 Accuracy assessment .....	3
2.3 Uncertainty assessment.....	5
3 Resources .....	8
4 Outputs .....	8

### List of Figures

Figure 1: Maps of topsoil (0-30 cm) salt-affected soils.....	2
Figure 2: Topsoil (0-30 cm) map of intensity of salt problems .....	3
Figure 3: Classification accuracy for salt-affected soils .....	4
Figure 4: Correlation functions for EC (0-30 cm) .....	6
Figure 5: Simulated realizations of normalized input soil indicators.....	7

### List of Tables

Table1: Characteristics of salt-affected soils .....	1
Table 2: Identifying intensity of salt problems in soil .....	2

# 1 Introduction

## 1.1 Overview

Salt-affected soils are groups of soils with high content of soluble salts and/or high amounts of sodium ions. The type and concentration of these salts and their drivers are important characteristics and the basis for mapping salt-affected soils. The three main categories for mapping salt-affected soils are: methods based on interpretation of soil-type maps, remote sensing applications, and methods based on soil indicators of salt problems. This Lesson targets the method using soil indicators of salt problems. It focuses on the classification of major types and problem intensity of salt-affected soils using input maps of indicators of salt problems. The Lesson targets national experts with knowledge of and access to indicators of salt-affected soils in their countries. Its outputs are expected to form the database and information for building national soil information of salt-affected soils.

## 1.2 Objective

The overall objective of this Lesson is to establish national spatial maps of salt-affected soils in each member country.

## 1.3 Expected outcomes

By the end of this Lesson, the participants are expected to:

- i. Understand the classification of salt-affected soils
- ii. Produce spatial maps of salt-affected soils
- iii. Locate sample points for future monitoring or update of the map of salt-affected soils

# 2 Activities

## 2.1 Classifying salt-affected soils

The three major types of salt-affected soils are: saline, sodic, and saline-sodic soils. Table 1 gives popularly used classification schemes for classifying these major types of salt-affected soils.

Table1: Characteristics of salt-affected soils

Soil property	Unit/Symbol	Type of salt-affected soil		
		Saline	Saline-sodic	Sodic
Electrical conductivity	EC <sub>e</sub> (dS/m)	> 4	> 4	< 4
Exchangeable sodium Percent	ESP	< 15	> 15	> 15
pH	-	< 8.5	< 8.5	> 8.5
Sodium adsorption ratio	SAR	< 13	> 13	> 13

The *soilassessment* package contains the functions *saltClass* for implementing the classification scheme in Table 1. The input data for this function are the three soil indicators, EC, pH, and ESP. The data be point-data in a spreadsheet dataframe or raster maps.



```

> library(soilassessment)
> predictors$salty=saltClass(predictors$ECse,predictors$PH,predictors$ESP,"FAO")
> summary(predictors$salty)

  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
1.000  1.000   1.000   1.151  1.000   5.000

> predictors$saltiness=classCode(predictors$salty,"saltclass")
> spplot(predictors["salty"])
> spplot(predictors["saltiness"]) # Figure 1

```

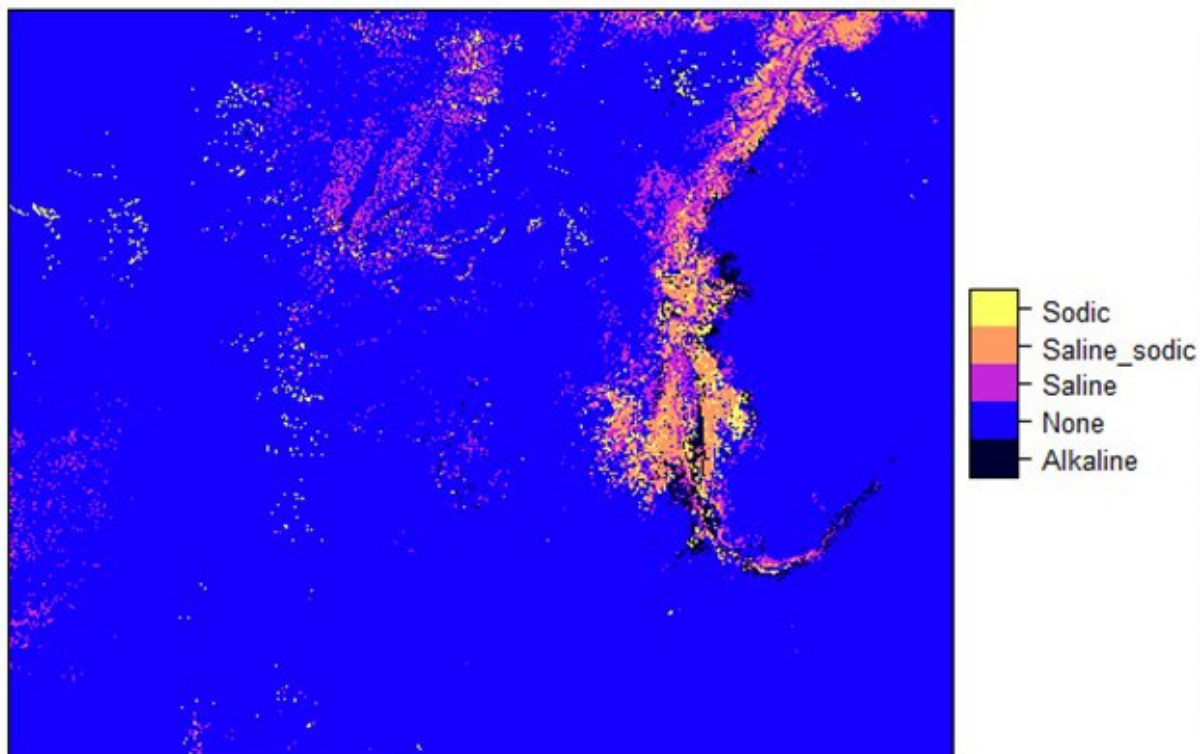


Figure 1: Maps of topsoil (0-30 cm) salt-affected soils

Classification of the intensity of salt problems in the soil is expressed using levels of electrical conductivity and exchangeable sodium ions (Table 2).

Table 2: Identifying intensity of salt problems in soil

Salinity ( $EC_e$ dS/m)			Sodicity (ESP)		
Intensity	FAO	USDA	Intensity	FAO	Amrhein (1996) <sup>1</sup>
None	< 0.75	0 - 2	None	< 15	< 6
slight	0.75 - 2	2 - 4	Slight	15 - 30	6 - 10
Moderate	2 - 4	4 - 8	Moderate	30 - 50	10 - 15
Strong	4 - 8	8 - 16	High/Strong	50 - 70	15 - 25
Very Strong	8 - 15	> 16	Extreme/V. Strong	> 70	> 25
Extreme	> 15				

<sup>1</sup> Amrhein C. 1996. Australian sodic soils: Distribution, properties, and management. Soil Science 161. pp412

```
> predictors$Salt_affected=saltseverity(predictors$ECse,predictors$PH,predictors$ES
P,"FAO")
> predictors$saltaffectedness=classCode(predictors$Salt_affected,"saltseverity")
> spplot(predictors["saltaffectedness"]) # Figure 2
```

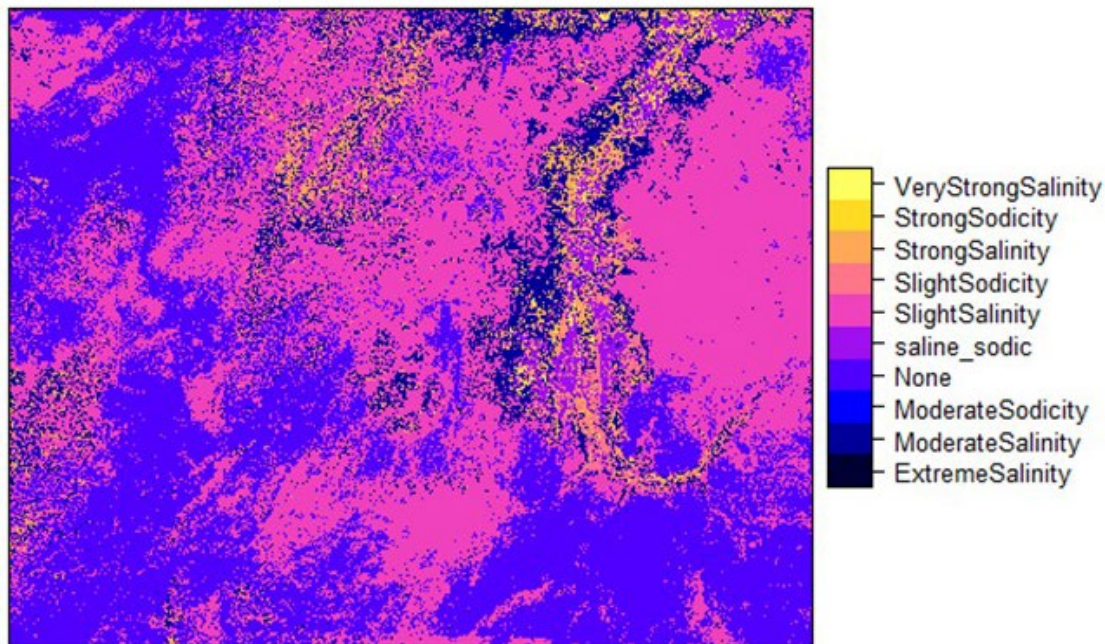


Figure 2: Topsoil (0-30 cm) map of intensity of salt problems

The final map in Figure 2 is exported as a GIS file format for data sharing. Since the export function does not work with factors /character values, the salt classes in the maps are first converted into numeric map-values and then exported. A look-up table (LUT) is necessary to help identify the classes and the unique numeric codes generated for each map-value. The LUT is exported as a text-file.

```
> predictors$Saltclass=as.numeric(predictors$saltaffectedness)
> salinity_LUT30=classLUT(predictors["saltaffectedness"],"saltseverity")

|=====| 100%

> writeGDAL(predictors["Saltclass"],drivername = "GTiff","Top0_30saltaffected.tif")
> write.table(salinity_LUT30,file = "saltaffected_LUT30.txt",row.names = FALSE)
```

## 2.2 Accuracy assessment

Accuracy of classified salt-affected map is assessed using confusion matrix. In this strategy holdout samples are independently classified in terms of types and severity of salt problems in the soil. Classification of the holdout samples should follow the same procedure of harmonization as other input data (that is, harmonization of input indicators and depths 0-30 and 30-100 cm). These classes are then compared to the pixel-extracted classes from the classified maps. The Kappa index is a suitable indicator for reporting the accuracy.

### *#Import and classify validation dataset*

```
> soilv=readOGR(".", "validation_harmonized")
> soilv=subset(soilv,soilv$Horizon==1)
> soilv$Salt_affected1=saltSeverity(soilv$EC,soilv$pH,soilv$ESP,"FAO")
> summary(soilv$Salt_affected1)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
   3.0    6.0    8.0    8.5   11.5   15.0
```



```
> soilv$saltaffectedness1=classCode(soilv$salt_affected1,"saltseverity")
> summary(soilv$saltaffectedness1)
```

ExtremeSalinity	ModerateSalinity	ModerateSodicity	None
3	10	1	18
saline_sodic	SlightSalinity	SlightSodicity	StrongSalinity
12	13	16	5

After classifying the validation dataset, the dataset is used to extract pixel values of the classified map and compared with the classified validation.

*# Extract the salt classes from the map using the validation samples*

```
> soilv=subset(soilv,!is.na(soilv$saltaffectedness1))
> predictors.ovv=over(soilv, predictors)
> soilv$salt_affected=predictors.ovv$salt_affected
> soilv$saltaffectedness=predictors.ovv$saltaffectedness
```

*# Check the summary of extracted classified pixels*

```
> summary(soilv$salt_affected)
  Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 3.000  6.000   8.000  7.808 10.000  14.000
> summary(soilv$saltaffectedness)
  ExtremeSalinity  ModerateSalinity  ModerateSodicity      None
                1                7                0          14
  saline_sodic    SlightSalinity    SlightSodicity  StrongSalinity
                18                15                11          10
  StrongSodicity  VeryStrongSalinity
                0                  2
```

A visual comparison shows that the validation datasets had points classified as extremely saline soil, but the map reported only one pixel. Similarly, three points in the validation datasets had strong salinity class while the map had 10 pixels with strong salinity classes. A graphical plot of the comparison (confusion matrix) gives a clear picture of the accuracy (Figure 3).

```
> agreementplot(confusion(soilv$salt_affected, soilv$salt_affected1),main = "Accuracy assessment",xlab = "Class codes in holdout samples", ylab = "Class codes in map")
> Kappa(confusion(soilv$salt_affected, soilv$salt_affected1))
```

	value	ASE	z	Pr(> z )
Unweighted	0.4211	0.06384	6.597	4.208e-11
weighted	0.4780	0.07792	6.135	8.525e-10

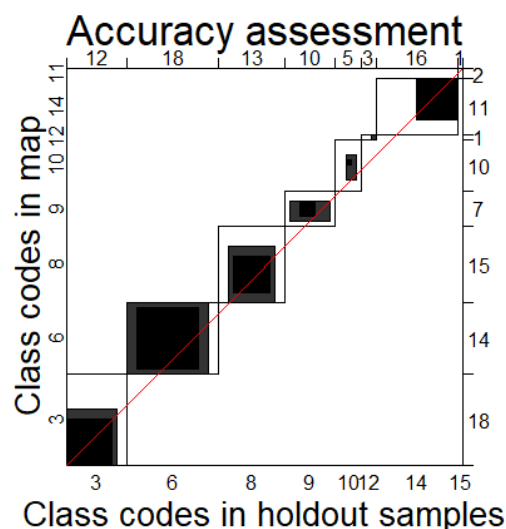


Figure 3: Classification accuracy for salt-affected soils

## 2.3 Uncertainty assessment

Uncertainty assessment in salt-affected modelling is conceived as estimating uncertainties contributed by input data modelling and uncertainties from the salt-classification model. The Monte Carlo uncertainty propagation approach is used to model both input data uncertainty and classification model uncertainty. This is a three-step approach involving input parameter specification, development of marginal and joint distributions, and simulations for uncertainty propagation. Input parameter specifications comprise definition of the salt-classification model and spatial distribution of mean and variance of the input variables. Spatial distribution of mean and variance are used to train the MC simulations at a set number of simulations/realizations. Usually, MC simulations are more accurate with a higher number of realizations. However, this may cost the analysis computing time for large datasets. A trade-off is necessary to safeguard suitable accuracy while at the same time incurring moderate computing time. A value of 100 is suggested for modelling salt-affected soils. Development of marginal density functions uses statistical distribution parameters and correlation models (crm). Examples of statistical distribution parameters, which depend on the type of distribution, are the mean ( $\bar{x}$ ) and standard variation ( $s$ ) for normal distribution or the scale ( $\lambda$ ) and rate ( $\alpha$ ) for gamma distribution.

The following steps describes the process for preparing the input indicator maps into raster file format, since the modules for uncertainty assessment were developed for the raster file format. The input maps are further converted into spatialPixelsDataFrame to facilitate mathematical operations on dataframes. It is important to check the probability distributions of the input data using the histogram function. Although MC simulations in Figure 4 are not strict on the type of the distribution, normal distribution is easy to sample. It's therefore recommended that normalized distributions be established from the input maps.

### *# Convert the input layers into raster files*

```
> EC=raster(predictors["ECse"]);names(EC)=c("EC"); EC1=as(EC,"SpatialPixelsDataFram
e")
> PH=raster(predictors["PH"]); names(PH)=c("PH"); PH1=as(PH,"SpatialPixelsDataFrame
")
> ESP=raster(predictors["ESP"]);names(EC)=c("ESP");ESP1=as(ESP,"SpatialPixelsDataFr
ame")
>
> ECte=raster(predictors["ECte"]);ECsd=pred_uncerta$pred_sd; names(ECsd)=c("ECsd")
> PHde=raster(predictors["PHt"]);PHsd=pred_uncertb$pred_sd; names(PHsd)=c("PHsd")
> ESPt=raster(predictors["ESPt"]);ESPsd=pred_uncertc$pred_sd; names(ESPsd)=c("ESPsd
")
```

### *# Obtain sample spatial autocorrelation (Figure 4)*

```
> b=nrow(EC1)
> c=trunc(0.01*b)
> jj=EC1[sample(b,c),]
> vrm=autofitVariogram(EC~1,jj)

> plot(vrm)#Note the spatial correlation model and the value of Range parameter
> acf((EC1$EC)) ##Also not the acf0 (at lag 0)
> EC_crm <- makeCRM(acf0 = 0.85, range = 20000, model = "Sph")
> plot(EC_crm, main = "EC correlogram")
```

The above correlation functions are repeated for all input soil indicators for 0-30 cm and 30-100 cm.

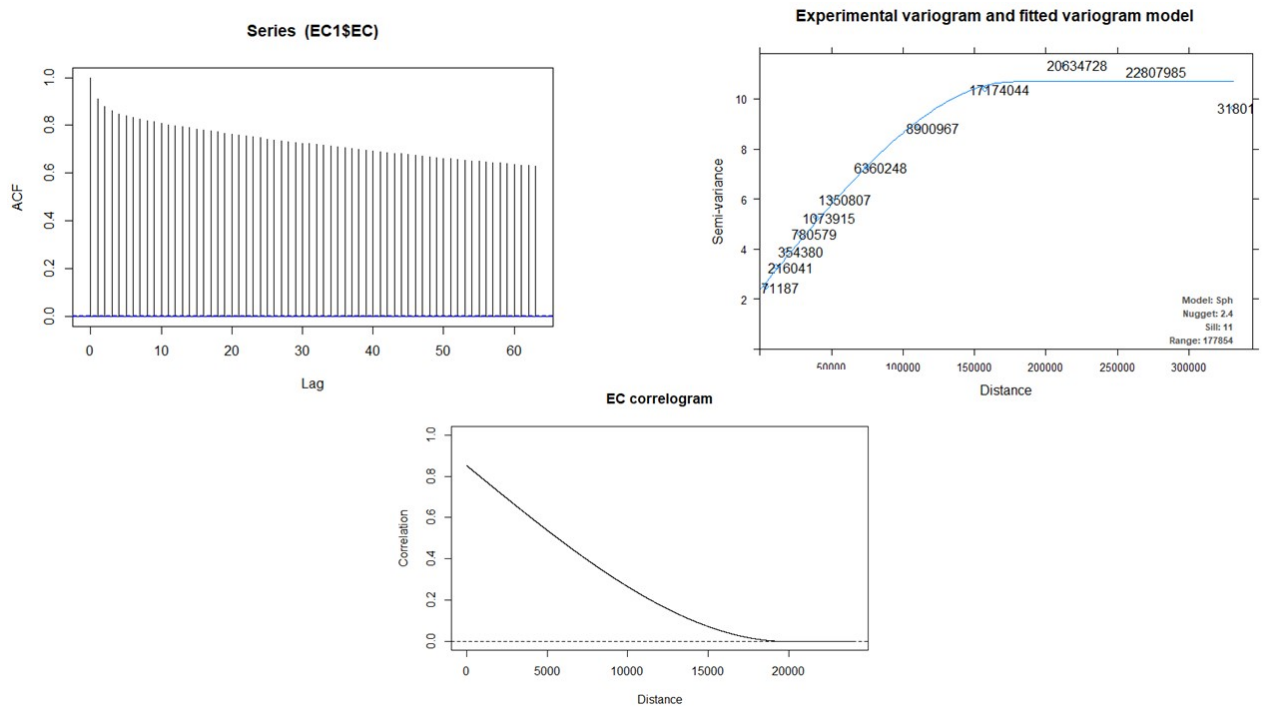


Figure 4: Correlation functions for EC (0-30 cm)

*# Develop input marginal and joint multivariate uncertainty models for defining MC models*

```
> EC_UM=defineUM(distribution="norm",distr_param = c(ECte,ECsd),crm =EC_crm,id = "E
C")
> PH_UM=defineUM(distribution = "norm",distr_param =c(PHde,PHsd),crm =PH_crm,id = "P
H")
> ESP_UM=defineUM(distribution="norm",distr_param=c(ESPt,ESPsd),crm=ESP_crm,id= "ES
P")
> class(EC_UM);class(PH_UM);class(ESP_UM)
[1] "MarginalNumericSpatial"
[1] "MarginalNumericSpatial"
[1] "MarginalNumericSpatial"
```

*#Get the correlation values and use them in defining the Monte Carlo Uncertainty Mode (MUM)*

```
> cor(values(ECte),values(PHde)); cor(values(ECte),values(ESPt)); cor(values(PHde),
values(ESPt))
[1] 0.5511048
[1] 0.3204495
[1] 0.2859129

> salinityMUM = defineMUM(UMlist = list(EC_UM, PH_UM, ESP_UM), cormatrix =
matrix(c(1, cor(values(ECte),values(PHde)), cor(values(ECte),values(ESPt)),
cor(values(ECte), values(PHde)), 1, cor(values(PHde), values(ESPt)),
cor(values(ECte), values(ESPt)), cor(values(PHde), values(ESPt)),1), nrow = 3, ncol
= 3))

> class(salinityMUM)
[1] "JointNumericSpatial"
```

Possible realizations are now developed after setting the Monte Carlo Uncertainty models (MUM). 100 level is set for simulating the MC simulations.



### *# Create MC realizations from the distributions*

```
> MC <- 100
> input_sample = genSample(UMobject = salinityMUM, n = MC, samplmethod = "ugs", n
max = 20, asList = FALSE)
Linear Model of Coregionalization found. Good.
[using unconditional Gaussian cosimulation]
```

### *# Compute input sample statistics*

```
> EC_sample = input_sample[[1:MC]]
> PH_sample = input_sample[[MC+1):(2*MC]]
> ESP_sample = input_sample[[2*MC+1):(3*MC]]
> EC_sample_mean <- mean(EC_sample)
> PH_sample_mean <- mean(PH_sample)
> ESP_sample_mean <- mean(ESP_sample)

> EC_sample_sd <- calc(EC_sample, fun = sd)
> PH_sample_sd <- calc(PH_sample, fun = sd)
> ESP_sample_sd <- calc(ESP_sample, fun = sd)
```

### *#Plot the realizations*

```
> par(mfrow=c(2,2),mar = c(1, 1, 2, 2), mgp = c(1.7, 0.5, 0), oma = c(0, 0, 0, 1),
+ las = 1, cex.main = 1, tcl = -0.2, cex.axis = 0.8, cex.lab = 0.8)
> plot(EC_sample_mean, main = "Mean of ECt realizations", xaxt = "n", yaxt = "n")
> plot(PH_sample_mean, main = "Mean of PHt realizations", xaxt = "n", yaxt = "n")
> plot(ESP_sample_mean, main = "Mean of ESpt realizations", xaxt = "n", yaxt = "n")
```

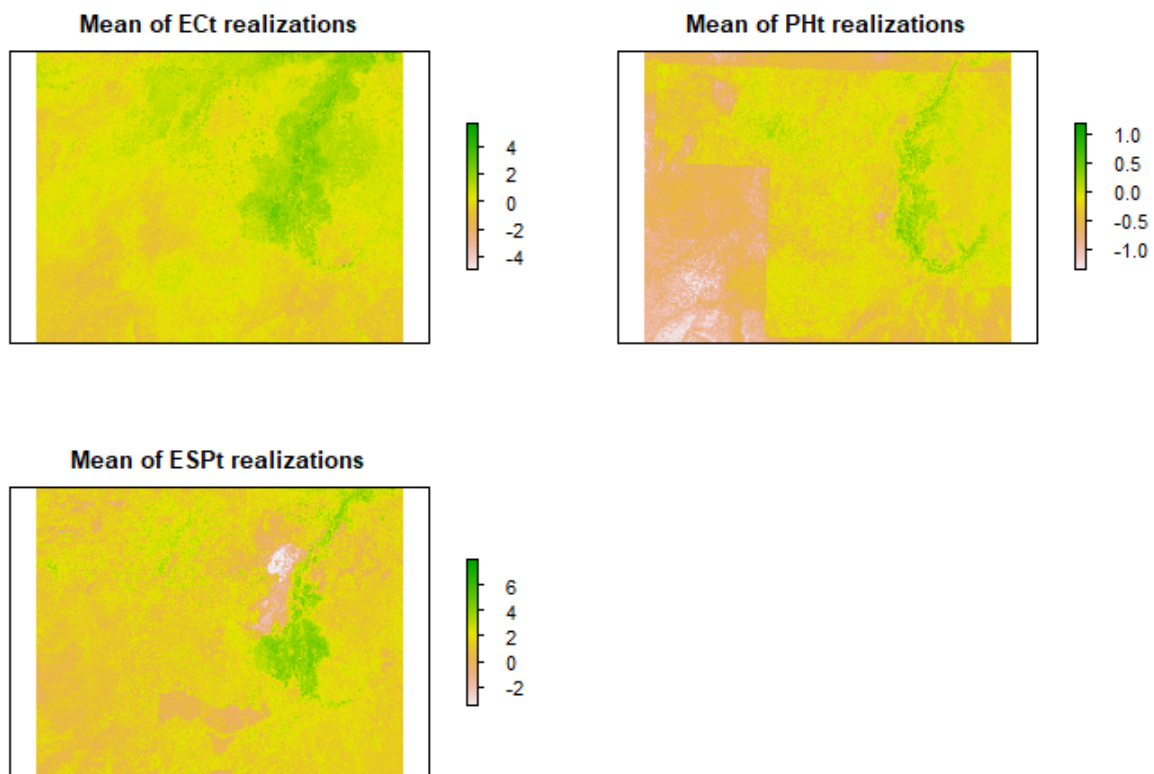


Figure 5: Simulated realizations of normalized input soil indicators

Note that the MC simulations were done on the normalized input soil indicators. It's possible to sample from non-normalized data. However, a harmonized statistical distribution is preferred.

### *# Uncertainty propagation through the classification model*

The uncertainty propagation model is used to determine how uncertainties in input soil indicators propagate through the classification model into the final classified map.

```
> Salinity_model_raster <- function (EC1,PH1,ESP1){
+   ww=EC1
+   ww=raster(ww)
+   ww$salt=saltSeverity(values(EC1),values(PH1),values(ESP1),"FAO")
+   ww=ww$salt; names(ww)=c("salt")
+   ww
+ }

> v <- list()
> v[[1]] = map(1:100, function(x){input_sample[[x]]})
> v[[2]] = map(101:200, function(x){input_sample[[x]]})
> v[[3]] = map(201:300, function(x){input_sample[[x]]})
> input_sample=v
> salinity_sample=propagate(realizations=input_sample,model=Salinity_model_raster,n
=MC)
```

### *#Determine the uncertainty in final classified map*

```
> samplelist <- list()
> samplelist [[1]] = map(1:100, function(x){input_sample[[x]]})
> samplelist [[2]] = map(101:200, function(x){input_sample[[x]]})
> samplelist [[3]] = map(201:300, function(x){input_sample[[x]]})
> input_sample= samplelist
> salinity_sample = propagate(realizations = input_sample, model =
Salinity_model_raster, n = MC)

> salinity_sample <- raster::stack(salinity_sample)
> names(salinity_sample) <- paste("salt.", c(1:nlayers(salinity_sample)), sep = "")
> salinity_freq = modal(salinity_sample, freq=TRUE)
> salinity_prop = salinity_freq/100
> salinity_SErr = sqrt(salinity_prop*(1-salinity_prop)/100)
> CL=0.95
> z_star=round(qnorm((1-CL)/2,lower.tail=F),digits = 2)
> salinity_MErr=z_star*salinity_SErr
```

The final output is exported to a GIS file format for data sharing.

```
> writeRaster(salinity_MErr,filename="Salinity_ME.tif",format="GTiff")
```

## 3 Resources

The following resources are useful for implementing the activities during data collection:

- References
  - Technical guidelines and cookbook for mapping salt-affected soils ([GSP-Secretariat@fao.org](mailto:GSP-Secretariat@fao.org))
  - Country guidelines and specifications for global mapping of salt-affected soils
- Data sources: Input maps of soil indicators of salt problems
- R Software and its contributed packages

## 4 Outputs

Each participant is expected to produce the following at the end of this lesson:

1. Spatial maps (0-30 and 30-100 cm) of salt-affected soils
2. Uncertainty maps (0-30 and 30-100 cm) of salt-affected soils
3. Map accuracy for 0-30 and 30-100 cm maps of salt-affected soils



Thanks to the financial support of



Ministry of Finance of the  
Russian Federation



European  
Commission



Rural Development  
Administration

