

# **Soil Texture Classification and Spatial Mapping Using Field Samples and World Soil Grid Data**

Noman Ahmad

2025-12-02

## **Table of contents**

<b>1</b>	<b>Introduction</b>	<b>2</b>
1.1	Research Question . . . . .	2
1.2	Hypothesis . . . . .	2
1.3	Objective . . . . .	2
<b>2</b>	<b>Data Sources</b>	<b>3</b>
<b>3</b>	<b>Analysis</b>	<b>3</b>
3.1	Requirements . . . . .	3
3.2	Data Cleaning . . . . .	4
3.3	Soil Texture Classification . . . . .	4
3.4	Soil Texture Mapping . . . . .	6
3.4.1	Kriging Interpolation . . . . .	6
3.4.1.1	Data Processing . . . . .	6
3.4.1.2	Variogram Modelling . . . . .	8
3.4.1.3	USDA Soil Texture Classification . . . . .	10
3.4.2	IDW Interpolation . . . . .	12
3.4.3	SoilGrids (250-m Global Dataset) . . . . .	15
3.4.4	Patchwork of USDA Soil Texture Maps . . . . .	20
<b>4</b>	<b>Conclusion</b>	<b>21</b>
<b>5</b>	<b>Acknowledgements</b>	<b>21</b>
<b>6</b>	<b>Abbreviations</b>	<b>22</b>
	<b>References</b>	<b>23</b>

# **1 Introduction**

Soil texture, defined by the relative proportions of sand, silt, and clay, is a fundamental property that influences water retention, nutrient availability, erosion susceptibility, and overall soil fertility (Daniels 2016). Understanding the spatial distribution of soil texture is essential for applications in agriculture, land management, hydrology, and environmental modeling. Field-based soil sampling provides precise measurements at specific locations but is typically limited in spatial coverage due to time, labor, and cost constraints (McBratney, Mendonça Santos, and Minasny 2003). On the other hand, global soil datasets, such as the World Soil Grid (WSG), provide spatially continuous estimates of soil properties at continental to global scales (Hengl et al. 2017). While useful for broad-scale analyses, these datasets may not accurately capture fine-scale variability present in local landscapes.

Integrating field-measured soil data with global soil grids, combined with spatial interpolation techniques such as Inverse Distance Weighting (IDW) and Kriging, allows researchers to generate detailed maps of soil texture and identify patterns of spatial heterogeneity (Li et al. 2011). Such integration improves the reliability of soil property assessments and supports decision-making in precision agriculture and environmental management. The present study focuses on classifying soil texture from field samples using the USDA soil texture triangle, generating spatially continuous interpolated maps, and comparing these results with World Soil Grid estimates. The analysis aims to highlight areas of agreement and discrepancy between local measurements and global predictions, thereby providing a more comprehensive understanding of soil texture variability in the study area.

## **1.1 Research Question**

How do soil texture classes derived from field-measured soil samples differ from those estimated by the World Soil Grid, and what spatial patterns emerge when using interpolation techniques (Kriging & IDW) in R?

## **1.2 Hypothesis**

Field-derived soil texture classes will show finer spatial variation than World Soil Grid predictions, with higher disagreement in areas of micro-scale heterogeneity.

## **1.3 Objective**

To classify soil texture using the USDA texture triangle, map spatial variability using interpolation techniques, and compare results with World Soil Grid data.

## 2 Data Sources

### 1. Field Soil Data

- Sand (%), Silt (%), Clay (%)
- Coordinates (WGS84)
- Format: CSV / XLSX

### 2. World Soil Grid (ISRIC)

- Sand, Silt, Clay (5-15 cm depth)
- Resolution: 250 m
- Format: GeoTIFF

### 3. Study Area

- Jaranwala (Pakistan) area shapefile defining the project boundary

## 3 Analysis

### 3.1 Requirements

The following R packages are required to run the SCaM (Soil Classification and Mapping) workflow. Install them before executing any scripts.

```
library (tidyverse)      # Data manipulation and visualization
library (readxl)        # Excel input
library (soiltexture)    # USDA texture classification
library (sf)             # Simple features (spatial)
library (sp)              # Spatial classes
library (gstat)          # Variography and kriging
library (raster)          # Raster processing
library (terra)          # Modern raster + vector handling
library (geodata)         # Downloads SoilGrids layers
library (ggplot2)         # Plotting (explicitly used)
library (patchwork)       # Combine plots
library (ggspatial)       # Spatial map annotations
```

### 3.2 Data Cleaning

The data cleaning process begins by reading raw soil texture data from an Excel file. Column names are renamed to uppercase for clarity, and missing values in SAND, SILT, and CLAY are removed. The data is then validated to ensure the sum of SAND, SILT, and CLAY equals 100%, before saving the cleaned data for further analysis.

```
# 1. Read raw data
soil <- read_xlsx("Data/Raw/Soil_Water_Data.xlsx")
print(soil)

# 2. Columns rename for clarity
soil <- soil %>%
  rename(SAND = `Sand_%`, SILT = `Silt_%`, CLAY = `Clay_%`)

# 3. Missing values removal
soil <- soil %>%
  filter(!is.na(SAND) & !is.na(SILT) & !is.na(CLAY))

# 4. Validation of sum of soil texture components
soil <- soil %>%
  mutate(total = SAND + SILT + CLAY) %>%
  filter(total == 100 )

# 5. Save cleaned & separated soil texture data
ST.data <- soil %>% select(Address_code, X_Coordinates, Y_Coordinates,
                           SAND, SILT, CLAY)
write_csv(ST.data, "Data/Processed/Soil_Data_Cleaned.csv")
```

### 3.3 Soil Texture Classification

The soil texture classification is visualized using the USDA Soil Texture Triangle, which categorizes soil based on the proportions of sand, silt, and clay. After processing the cleaned data and assigning a texture class to each sample, the soil data is plotted on the triangle (Figure 1). The plot visually represents the classification of soil texture, with each sample color-coded based on its texture class. The resulting figure is saved as a high-resolution PNG for further considerations.

```
# 1. Read processed data
soil_data <- read.csv("Data/Processed/Soil_Data_Cleaned.csv")

# 2. Get ISSS/USDA/FAO texture matrix
```

```

tex_matrix <- TT.points.in.classes(tri.data = soil_data[
  c("SAND", "SILT", "CLAY")], class.sys = "USDA.TT")
print(tex_matrix)

# 3. Convert matrix (0/1) to a single texture class column
soil_data$Texture_Class <- apply(tex_matrix, 1, function(row)
  {classes <- names(row)[row == 1]; if (length(classes) == 0)
    return(NA); classes[1]})

SoilTC <- soil_data %>% select(Address_code, SAND, SILT, CLAY, Texture_Class)
print(SoilTC)
write.csv(SoilTC, "SCaM/Outputs/Tables/Soiltexc.csv")

# 4. Soil Texture Triangle

png("SCaM/Outputs/Figures/Soil_Texture_Triangle.png",
  width = 2000, height = 1800, res = 300)

par(family = "serif")

classes <- unique(soil_data$Texture_Class)
classes <- classes[!is.na(classes)]

palette_colors <- rainbow(length(classes))
names(palette_colors) <- classes

TT.plot(class.sys = "USDA.TT", main = "USDA Soil Texture Classification",
  cex.lab = 1.1, cex.axis = 1.0, cex.main = 1.25, frame.bg.col = "white")

TT.points(tri.data = soil_data, geo = TT.geo.get("USDA.TT"),
  col = palette_colors[soil_data$Texture_Class], pch = 19, cex = 1.2)

legend("topleft", inset = c(1.04, 0.06), legend = classes,
  col = palette_colors[classes], title = "Classes", title.cex = 1.0,
  title.font = 2, pch = 19, pt.cex = 1.0, cex = 0.9, bty = "o")

dev.off()

```

## USDA Soil Texture Classification

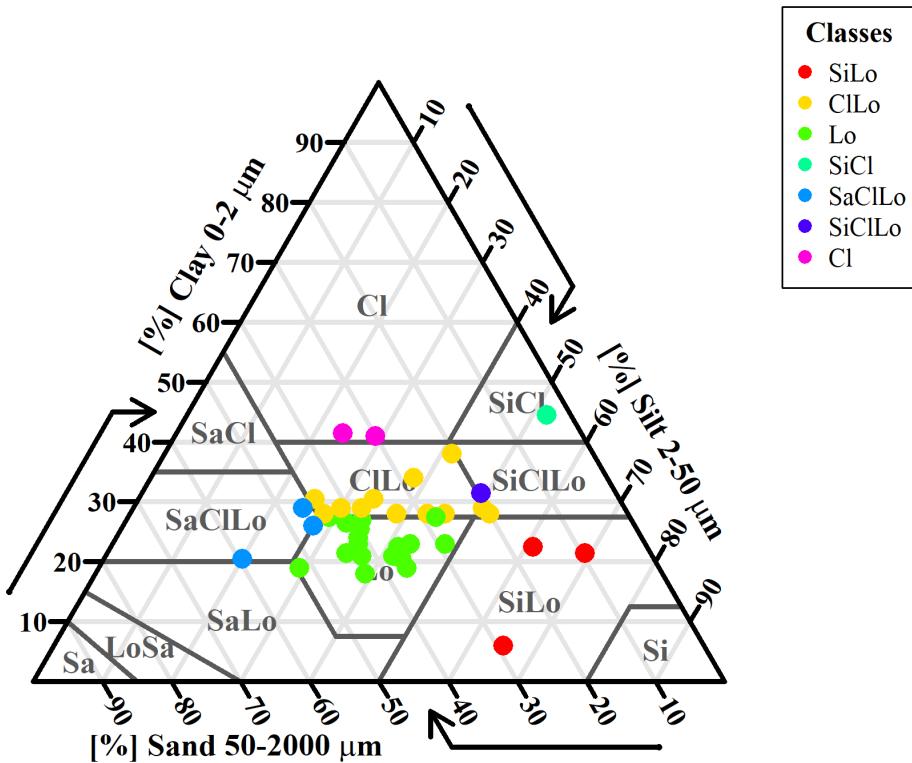


Figure 1: USDA Soil Texture Triangle showing the classification of soil texture based on the proportions of sand, silt, and clay.

### 3.4 Soil Texture Mapping

#### 3.4.1 Kriging Interpolation

##### 3.4.1.1 Data Processing

This section imports raw soil point data and the study-area boundary, assigns appropriate coordinate reference systems, and prepares spatial objects for geostatistical analysis. The dataset is projected from geographic (WGS84) to UTM to ensure distance-based geostatistics

are computed correctly. A prediction grid covering the area of interest is then generated and masked to the study boundary.

```
# 1. Paths & CRS
input_csv    <- "Data/Processed/Soil_Data_Cleaned.csv"
input_shp     <- "Data/Raw/study_area.shp"
out_fig_dir  <- "Outputs/Figures"
out_tbl_dir  <- "Outputs/Tables"
dir.create(out_fig_dir, showWarnings = FALSE, recursive = TRUE)
dir.create(out_tbl_dir, showWarnings = FALSE, recursive = TRUE)

epsg_input    <- 4326
epsg_kriging <- 32642
epsg_map      <- 4326

# 2. Load Data
soil_df <- read.csv(input_csv, stringsAsFactors = FALSE)
req_cols <- c("X_Coordinates","Y_Coordinates","SAND","SILT","CLAY")
stopifnot(all(req_cols %in% names(soil_df)))

soil_sf <- st_as_sf(soil_df, coords = c("X_Coordinates","Y_Coordinates"),
                      crs = epsg_input, remove = FALSE)

aoi_sf <- st_read(input_shp, quiet = TRUE)
if (is.na(st_crs(aoi_sf))) st_crs(aoi_sf) <- epsg_input

soil_utm <- st_transform(soil_sf, epsg_kriging)
aoi_utm   <- st_transform(aoi_sf,  epsg_kriging)
soil_sp   <- as(soil_utm, "Spatial")

# 3. Prediction Grid
res_m <- 250
bb <- st_bbox(aoi_utm)

grid_df <- expand.grid(
  x = seq(bb["xmin"], bb["xmax"], by = res_m),
  y = seq(bb["ymin"], bb["ymax"], by = res_m))

grid_sf <- st_as_sf(grid_df, coords = c("x","y"), crs = epsg_kriging)

inside_mat <- st_within(grid_sf, aoi_utm, sparse = FALSE)
inside <- apply(inside_mat, 1, any)
pred_pts_sp <- as(grid_sf[inside, ], "Spatial")
```

### 3.4.1.2 Variogram Modelling

The analysis derives empirical variograms for the sand, silt, and clay fractions and then fits suitable theoretical models to support subsequent kriging. Exponential or spherical models are fitted using robust initial estimates to stabilize optimisation. The final combined variogram figure 2 summarizes spatial dependence for all three soil fractions.

```
# 1. Variogram Calculation
compute_vario_params <- function(aoi_sf) { bb <- st_bbox(aoi_sf)
  diag_len <- sqrt((bb["xmax"] - bb["xmin"])^2 + (bb["ymax"] - bb["ymin"])^2)
  list(cutoff = diag_len / 2, width = diag_len / 30)}

vp <- compute_vario_params(aoi_utm)
fit_variogram <- function(spfd, z) {fmla <- as.formula(paste0(z, " ~ 1"))
  v <- variogram(fmla, locations = spfd, cutoff = vp$cutoff, width = vp$width)

  nugget_guess <- min(v$gamma, na.rm = TRUE)
  sill_guess <- max(v$gamma, na.rm = TRUE)
  psill_guess <- sill_guess - nugget_guess
  range_guess <- max(v$dist) / 3

  init <- vgm(psill = psill_guess, model = "Exp", range = range_guess,
               nugget = nugget_guess)
  fit <- fit.variogram(v, model = init)

  list(v = v, model = fit)}

var_sand <- fit_variogram(soil_sp, "SAND")
var_silt <- fit_variogram(soil_sp, "SILT")
var_clay <- fit_variogram(soil_sp, "CLAY")

# 2. ggplot variogram figure
vario_to_gg <- function(vario, title_prefix, var_name) {vdf <- vario$v
  mdf <- variogramLine(vario$model, maxdist = max(vdf$dist))
  ggplot(vdf, aes(x = dist, y = gamma)) + geom_point() +
    geom_line(data = mdf, aes(x = dist, y = gamma)) +
    theme_bw(base_size = 14, base_family = "serif") +
    labs(x = "Distance", y = "Semivariance",
         title = paste0(title_prefix, " - ", var_name))}
```

```

pA <- vario_to_gg(var_sand, "A", "SAND")
pB <- vario_to_gg(var_silt, "B", "SILT")
pC <- vario_to_gg(var_clay, "C", "CLAY")

combined_plot <- pA / pB / pC

ggsave(file.path(out_fig_dir, "variogram_ABC_patchwork.png"),
       plot = combined_plot, width = 8, height = 12, dpi = 300)

```

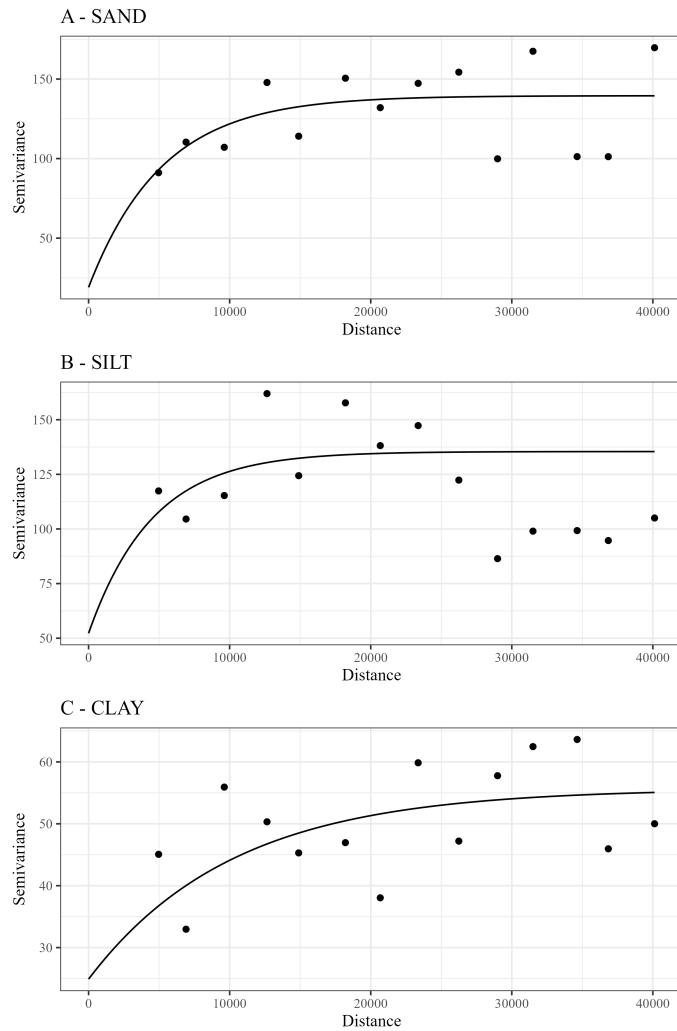


Figure 2: Semivariograms for soil texture fractions. The plots display experimental semivariance (dots) and fitted model curves (lines) for (A) Sand, (B) Silt, and (C) Clay over distance.

### 3.4.1.3 USDA Soil Texture Classification

In this section, ordinary kriging is applied to interpolate sand, silt, and clay across the study area using the previously fitted variogram models, producing continuous, spatially coherent estimates. The kriged soil fractions are normalized to ensure they sum to 100%, preserving their compositional nature before classification. USDA soil texture classes are then assigned using a ternary decision system, allowing transformation of continuous particle-size data into interpretable categorical texture groups. The final texture map (Figure 3) reveals spatial patterns in dominant soil types and highlights regions of textural transitions across the landscape.

```
# 1. Kriging soil texture data

krige_one <- function(z, model) {fmla <- as.formula(paste0(z, " ~ 1"))
gstat::krige(fmla, locations = soil_sp, newdata = pred_pts_sp, model = model)}

kr_sand <- krige_one("SAND", var_sand$model)
kr_silt <- krige_one("SILT", var_silt$model)
kr_clay <- krige_one("CLAY", var_clay$model)

coords <- coordinates(kr_sand)
kr_df <- data.frame(x_utm = coords[,1], y_utm = coords[,2],
                     SAND = kr_sand$var1.pred, SILT = kr_silt$var1.pred, CLAY = kr_clay$var1.pred)

# 2. Normalize SSC to 100%
ssc <- kr_df[c("CLAY", "SILT", "SAND")]
ssc[ssc < 0] <- 0
row_sum <- rowSums(ssc); row_sum[row_sum==0] <- 1
ssc_norm <- sweep(ssc, 1, row_sum, "/") * 100
kr_df[c("CLAY", "SILT", "SAND")] <- ssc_norm

# 3. USDA Texture Classes
usda <- TT.points.in.classes(kr_df[c("CLAY", "SILT", "SAND")],
                               class.sys = "USDA.TT")
kr_df$USDA_texture <- as.character(usda)

# 4. Project and map
pred_sf_utm <- st_as_sf(kr_df, coords = c("x_utm", "y_utm"),
                          crs = epsg_kriging, remove = FALSE)
pred_sf_wgs84 <- st_transform(pred_sf_utm, epsg_map)
coords_ll <- st_coordinates(pred_sf_wgs84)
pred_sf_wgs84$lon <- coords_ll[,1]
pred_sf_wgs84$lat <- coords_ll[,2]
```

```

# 5. Color palette (subset automatically)
palette_usda <- c(
  "Sand"="#F4D06F", "Loamy sand"="#F7E1A0", "Sandy loam"="#FFE699",
  "Loam"="#8BC34A", "Silt loam"="#AEDFF7", "Silt"="#6EC1E4",
  "Sandy clay loam"="#F5A65B", "Clay loam"="#C58C5E",
  "Silty clay loam"="#A38BD4", "Sandy clay"="#E07B39",
  "Silty clay"="#7A4E9A", "Clay"="#D9534F")

used <- sort(unique(pred_sf_wgs84$USDA_texture))
palette_used <- palette_usda[used]

aoi_ll <- st_transform(aoi_sf, epsg_map)

mean_lat <- mean(pred_sf_wgs84$lat, na.rm = TRUE)
lon_step <- 250 / (111320 * cos(mean_lat * pi/180))
lat_step <- 250 / 110540

p_kriging <- ggplot() +
  geom_sf(data = aoi_ll, fill = NA, color = "white", linewidth = 1.5) +
  geom_tile(data = st_drop_geometry(pred_sf_wgs84),
            aes(x = lon, y = lat, fill = USDA_texture),
            width = lon_step, height = lat_step) +
  scale_fill_manual(values = palette_used) +
  theme_bw(base_family="serif") +
  theme(legend.position="bottom", panel.grid=element_blank()) +
  annotation_north_arrow(location = "tr",
                          style = north_arrow_fancy_orienteering(text_family="serif")) +
  annotation_scale(location = "br", text_family="serif") +
  ggtitle("USDA Soil Texture Classes (Normalized & Kriged)")

ggsave(file.path(out_fig_dir, "USDA_soil_texture_classes_Kriging_map.png"),
       plot = p_kriging, width = 10, height = 8, dpi = 300)

```

USDA Soil Texture Classes (Normalized & Kriged)

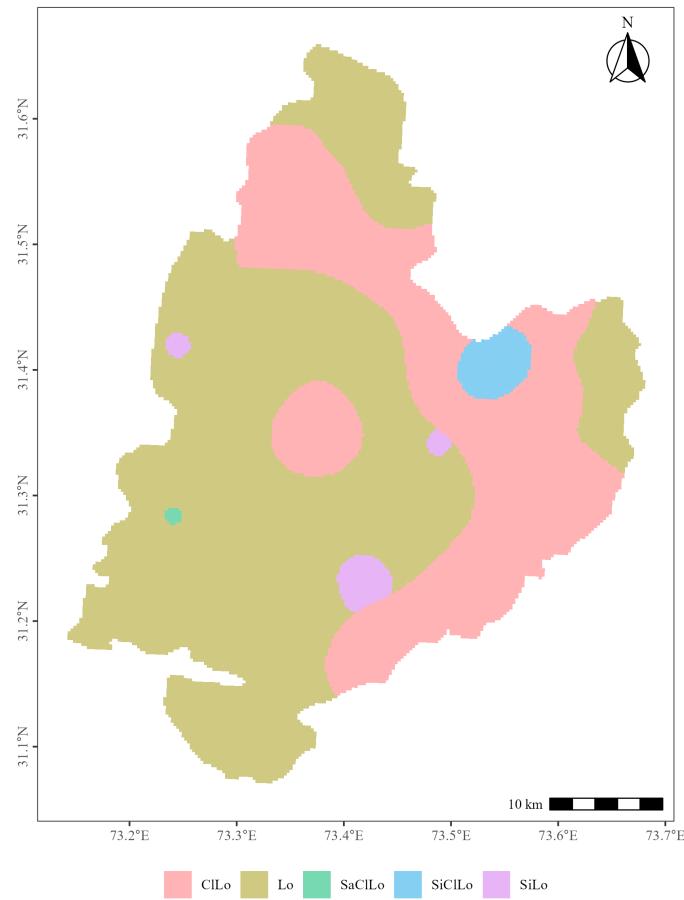


Figure 3: Spatial distribution of USDA soil texture classes generated via Kriging interpolation.

#### 3.4.2 IDW Interpolation

Inverse Distance Weighting (IDW) was implemented as a deterministic alternative to kriging, providing a proximity-based estimate of SAND, SILT, and CLAY across the study area. After interpolation, SSC values were normalized to maintain their compositional structure, and USDA texture classes were assigned using the same classification scheme applied to the kriged results. The resulting thematic map (Figure 4) illustrates the spatial patterns derived from a purely distance-weighted method.

```
# 1. IDW function (uses soil_sp & pred_pts_sp from kriging section)
run_idw <- function(var, idp = 2, nmax = 20)
  {fmla <- as.formula(paste0(var, " ~ 1"))
  pred <- gstat::idw(formula = fmla, locations = soil_sp,
```

```

            newdata = pred_pts_sp,
idp = idp, nmax = nmax) pred$var1.pred}

# 2. Run IDW for SAND-SILT-CLAY
idw_df <- data.frame(x_utm = pred_pts_sp@coords[,1],
                      y_utm = pred_pts_sp@coords[,2], SAND = run_idw("SAND"),
                      SILT = run_idw("SILT"), CLAY = run_idw("CLAY"))

# 3. Normalize SSC to 100%
ssc <- idw_df[, c("CLAY", "SILT", "SAND")]
ssc[ssc < 0] <- 0
row_sum <- rowSums(ssc)
row_sum[row_sum == 0] <- 1
ssc_norm <- sweep(ssc, 1, row_sum, "/") * 100
idw_df[, c("CLAY", "SILT", "SAND")] <- ssc_norm

# 4. USDA classification (same system as kriging)
usda <- TT.points.in.classes(idw_df[c("CLAY", "SILT", "SAND")],
                               class.sys = "USDA.TT")
if (is.factor(usda)) {idw_df$USDA_texture <- as.character(usda)} else {
  m <- as.matrix(usda)
  idx <- max.col(m, ties.method = "first")
  idw_df$USDA_texture <- colnames(m)[idx]}

# 5. Convert to sf and project to WGS84 (same as kriging)
idw_sf_utm <- st_as_sf(idw_df, coords = c("x_utm", "y_utm"),
                        crs = epsg_kriging, remove = FALSE)
idw_sf_wgs84 <- st_transform(idw_sf_utm, epsg_map)
coords_idw <- st_coordinates(idw_sf_wgs84)
idw_sf_wgs84$lon <- coords_idw[,1]
idw_sf_wgs84$lat <- coords_idw[,2]

# 6. Build IDW-specific legend (critical fix)
used_idw <- sort(unique(idw_sf_wgs84$USDA_texture))

idw_sf_wgs84$USDA_texture <- factor(idw_sf_wgs84$USDA_texture,
                                       levels = used_idw)

palette_used_idw <- palette_usda[used_idw]

missing_idw <- is.na(palette_used_idw)
if (any(missing_idw)) {palette_used_idw[missing_idw] <-

```

```

    hcl.colors(sum(missing_idw), "Set 3")}

names(palette_used_idw) <- used_idw

# 7. Plot IDW map (identical style to kriging map)
p_idw <- ggplot() +
  geom_sf(data = aoi_ll, fill = NA, color = "white", linewidth = 1.5) +
  geom_tile(data = st_drop_geometry(idw_sf_wgs84),
    aes(x = lon, y = lat, fill = USDA_texture),
    width = lon_step, height = lat_step) +
  scale_fill_manual(values = palette_used_idw, drop = FALSE) +
  coord_sf(expand = TRUE) + labs(x = NULL, y = NULL) +
  theme_bw(base_family = "serif") +
  theme(legend.position = "bottom", legend.title = element_blank(),
    panel.grid = element_blank()) +
  annotation_north_arrow(location = "tr",
    style = north_arrow_fancy_orienteering(text_family = "serif")) +
  annotation_scale(location = "br", text_family = "serif") +
  ggtitle("USDA Soil Texture Classes (Normalized & IDW)") +
  theme(plot.title = element_text(face = "bold", size = 16),
    axis.text.y = element_text(angle = 90, hjust = 0.5))

print(p_idw)

ggsave(file.path(out_fig_dir, "USDA_soil_texture_classes_IDW_map.png"),
       plot = p_idw, width = 10, height = 8, dpi = 300)

```

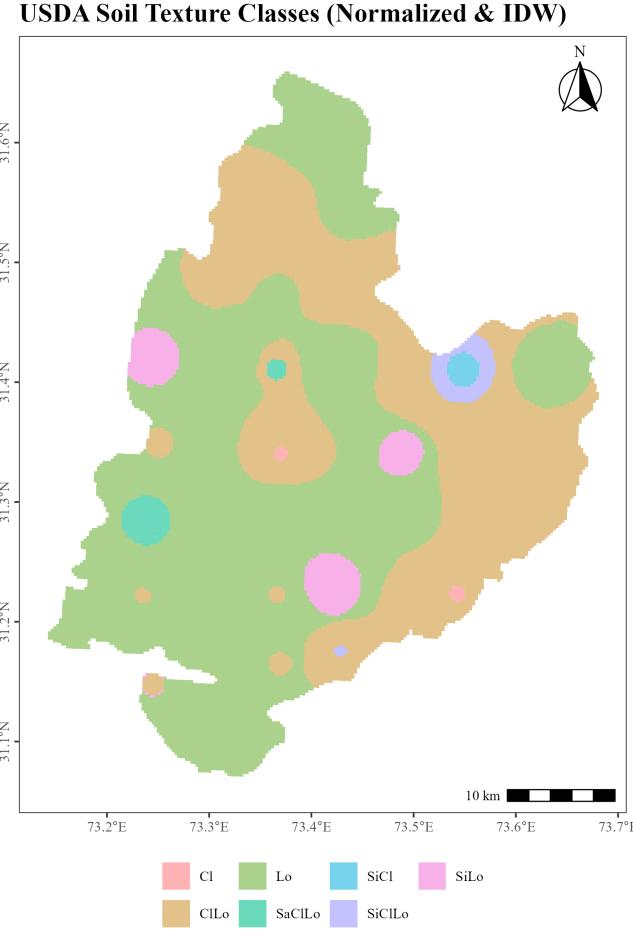


Figure 4: Spatial distribution of USDA soil texture classes generated via IDW interpolation.

### 3.4.3 SoilGrids (250-m Global Dataset)

Soil texture information from SoilGrids (250 m resolution) was incorporated to provide a global, remote-sensing-derived comparison against field-based kriging and IDW predictions. Raster layers for sand, silt, and clay (5-15 cm depth) were downloaded and clipped to the study area before being reprojected and converted to point data for analysis. The SoilGrids SSC values were normalized, used to derive silt, and classified into USDA soil texture groups using the same ternary-based classification applied throughout this study. The resulting map (Figure 5) provides a spatially continuous representation of SoilGrids-derived USDA texture classes for the study area, serving as an independent, globally modeled baseline for subsequent visualization and integration.

```

# 1. Load study area (ensure WGS84)
study_area <- st_read("SCaM/Data/Raw/study_area.shp")
study_area_sf <- st_transform(study_area, 4326)
study_area <- vect(study_area_sf) # sf -> terra

# 2. Download SoilGrids raster datasets (5-15 cm layer = depth = 15)
path <- "SCaM/Data/Raw/soilgrids_data"

sand <- soil_world("sand", depth = 15, stat = "mean", path = path)
silt <- soil_world("silt", depth = 15, stat = "mean", path = path)
clay <- soil_world("clay", depth = 15, stat = "mean", path = path)

# 3. Crop + mask study area -> View -> Save .tif

sand_c <- mask(crop(sand, study_area), study_area)
silt_c <- mask(crop(silt, study_area), study_area)
clay_c <- mask(crop(clay, study_area), study_area)

plot(sand_c, main = "Sand (Clipped)")
plot(silt_c, main = "Silt (Clipped)")
plot(clay_c, main = "Clay (Clipped)")

writeRaster(sand_c, "SCaM/Data/Processed/sand_clipped.tif")
writeRaster(silt_c, "SCaM/Data/Processed/silt_clipped.tif")
writeRaster(clay_c, "SCaM/Data/Processed/clay_clipped.tif")

# 4. Read and project rasters
SAND.raw <- raster("Data/Processed/clay_clipped.tif")
CLAY.raw <- raster("Data/Processed/clay_clipped.tif")

# Read projection system
crs(CLAY.raw)
projection(CLAY.raw)

# Create projection system (correct syntax)
utm <- CRS("+proj=longlat +datum=WGS84 +no_defs")

# Assign/Project to new projection system
CLAY <- projectRaster(from = CLAY.raw, crs = utm, method = "ngb")
crs(CLAY)
projection(CLAY)

```

```

SAND <- projectRaster(from = SAND.raw, crs = utm, method = "ngb")

# 5. Create & join point from rasters
point1 <- rasterToPoints(CLAY, fun = function(x) {x>0}, spatial = TRUE)
plot(CLAY)
names(point1)

# Extract sand values
x <- extract(SAND, point1)

# Join the extracted values to point shape file
point2 <- cbind(point1, x)
names(point2)
names(point2) <- c("CLAY", "SAND")
names(point2)

# 6. Convert to data frame
point3 <- as.data.frame(point2)
names(point3)
head(point3)

# 7. Calculate silt value and create soil texture classes
point3$SILT <- 100-point3$CLAY-point3$SAND
head(point3)

# Create soil texture class
point3$Texture <- TT.points.in.classes(tri.data = point3,
                                         class.sys = "USDA.TT", PiC.type = "t")
head(point3)
str(point3)

point3$TextureClass = factor(point3$Texture)
point3$i.texclass = as.numeric(point3$TextureClass)
r = rasterFromXYZ(point3[, c("x", "y", "i.texclass")], crs = utm)
print(r)

r[] = as.numeric(factor(levels(point3$TextureClass))[r[]])

# Define the custom color palette for soil texture classes
manual_palette <- c(
  "#b2182b", "#ffd8b1", "#f4a582", "#92c5de",
  "#fddbc7", "#1b7837", "#2166ac", "#67a9cf",

```

```

"#d1e5f0", "#f7f7f7", "#F4D06F", "#F5A65B")

# Now plot the raster with the manual color palette
plot(r, main = "Soil Texture Classes (Rasterized)", col = manual_palette)

# Get the resolution (cell size) of the raster
lon_step <- res(r)[1] # Longitude resolution
lat_step <- res(r)[2] # Latitude resolution

# 8. Build ggplot map with USDA Soil Texture classes
p_grid <- ggplot() +
  geom_sf(data = study_area_sf, fill = NA, color = "white", linewidth = 1.5) +
  geom_tile(data = point3, aes(x = x, y = y, fill = Texture),
            width = lon_step, height = lat_step) +
  scale_fill_manual(values = manual_palette, drop = FALSE) +
  coord_sf(expand = TRUE) +
  labs(x = NULL, y = NULL) +
  theme_bw(base_family = "serif") +
  theme(legend.position = "bottom", legend.title = element_blank(),
        panel.grid = element_blank()) +
  annotation_north_arrow(location = "tr",
                          style = north_arrow_fancy_orienteering
                          (text_family = "serif")) +
  annotation_scale(location = "br", text_family = "serif") +
  ggtitle("SoilGrids Data (250 m) USDA Texture Classes") +
  theme(plot.title = element_text(face = "bold", size = 16),
        axis.text.y = element_text(angle = 90, hjust = 0.5))

print(p_grid)

# 9. Save the map as PNG

out_fig_dir <- "SCaM/Outputs/Figures"
ggsave(file.path(out_fig_dir, "SoilGrid_Data_Texture_Classification_map.png"),
       plot = p_grid, width = 10, height = 8, dpi = 300)

```

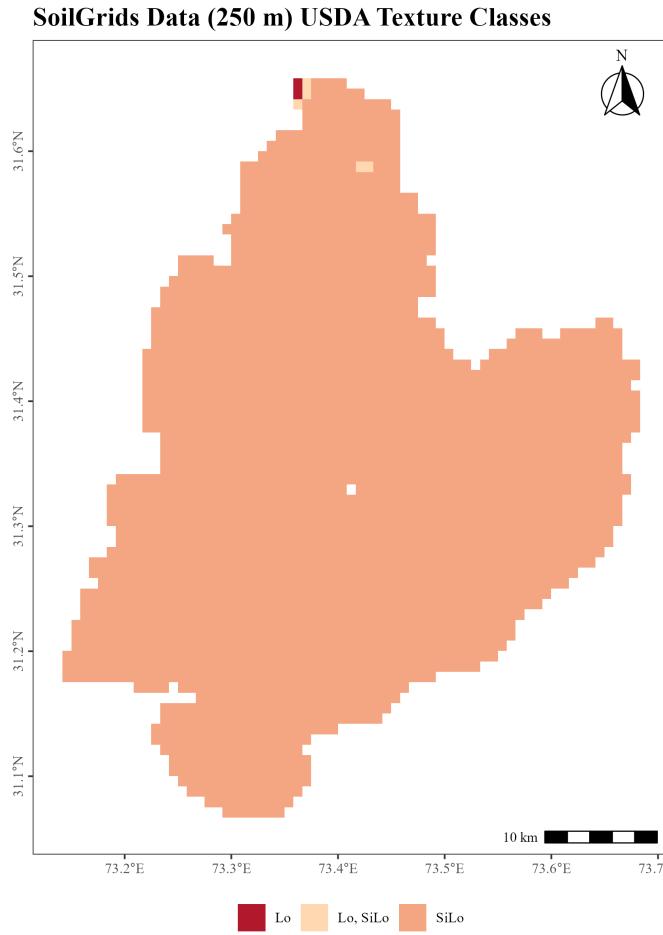


Figure 5: Soil texture classes derived from SoilGrids (250 m) sand, silt, and clay predictions for the 5-15 cm depth interval, clipped to the study area and reclassified using the USDA textural triangle

Table 1: SoilGrids-derived point coordinates and USDA texture classes extracted from the 250 m sand–silt–clay rasters (full table available in Outputs/Tables/SoilGridss\_table.csv).

X	Y	Texture
73.36241	31.65416	Lo
73.37074	31.65416	Lo, SiLo
73.37907	31.65416	SiLo
73.38740	31.65416	SiLo
Continue.....		

### 3.4.4 Patchwork of USDA Soil Texture Maps

The three USDA soil texture maps, derived respectively from kriging, IDW, and SoilGrids, were combined into a single patchwork layout to allow clear side-by-side visualization of their spatial patterns. This arrangement places all interpolation and modeling outputs on a shared geographic frame, enabling direct visual inspection of similarities, contrasts, and structural differences among the methods. The patchwork figure (Figure 6) provides an integrated overview of the three approaches before proceeding to a more detailed comparison in the subsequent section.

```
#Patch Work
A <- p_kriging + ggtitle("A: USDA Soil Texture Map (Kriging)")
B <- p_idw    + ggtitle("B: USDA Soil Texture Map (IDW)")
C <- p_grid   + ggtitle("C: USDA Soil Texture Map (SoilGrids)")

combined_plot <- (A | B | C)
plot(combined_plot)

ggsave(file.path(out_fig_dir, "Kriging_IDW_SoilGrids_comparison.png"),
       plot = combined_plot, width = 12, height = 6, dpi = 300)
```

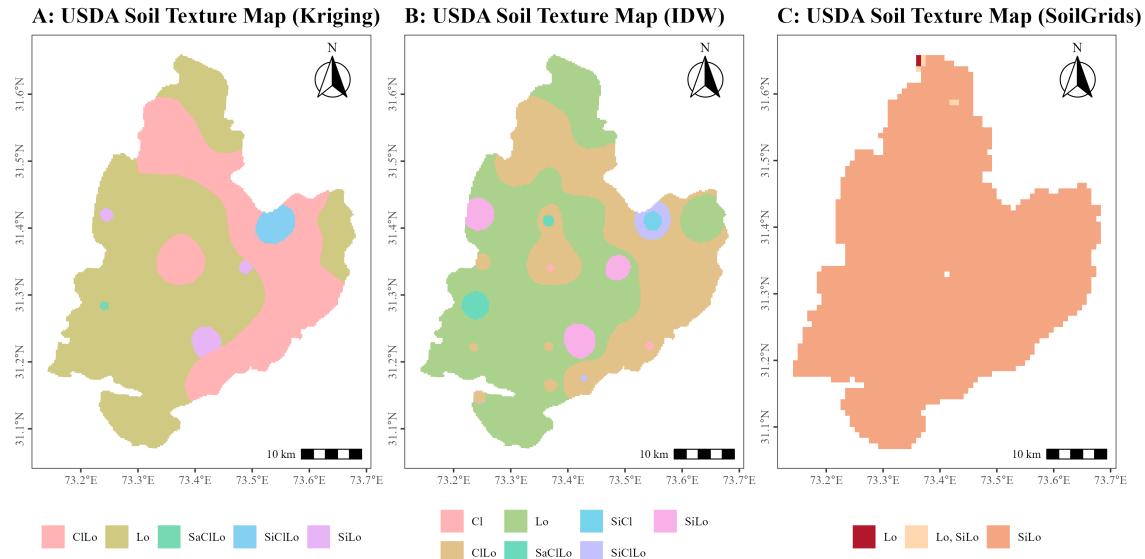


Figure 6: Patchwork visualization of USDA soil texture classes generated using kriging, IDW, and SoilGrids (250 m) datasets, displayed on a common spatial framework for visual comparison.

## 4 Conclusion

This project demonstrates that interpolated soil texture maps generated from field samples (Kriging and IDW, 250 m) reveal finer spatial variability than the global SoilGrids product. While SoilGrids offers a consistent regional baseline, the interpolation methods capture local heterogeneity reflected in measured texture class differences. This underscores the value of field data and suitable spatial interpolation when high-resolution or site-specific soil information is needed.

## 5 Acknowledgements

This project was developed with valuable support and inspiration from several open-source tools and data providers. The core code flow builds upon the excellent [soiltexture](#) package by [Julien Moeys](#). Additional inspiration was drawn from the [ggsoiltexture](#) project of [Sara Acevedo](#). I also acknowledge all other R packages that are used in building this workflow. Special thanks to the [global SoilGrids \(250 m\)](#) dataset for providing essential soil information used in this project. I also thank AI-assisted tools for helping me overcome challenges and deepen my understanding throughout this project.

**Important note:** All the tables and figures generated in this project are available in *SCaM/Outputs* folder.

## 6 Abbreviations

Abbreviations and full names of USDA soil texture classes used in this study.

Abbreviation	Full Name
S	Sand
LS	Loamy Sand
SL	Sandy Loam
L	Loam
SiL	Silt Loam
Si	Silt
SCL	Sandy Clay Loam
CL	Clay Loam
SiCL	Silky Clay Loam
SC	Sandy Clay
SiC	Silky Clay
C	Clay

## References

- Daniels, W. Lee. 2016. "The Nature and Properties of Soils, 15th Edition" Ray R. Weil and Nyle C. Brady. Pearson Press, Upper Saddle River NJ, 2017. 1086 p. \$164.80. ISBN-10: 0-13-325448-8; ISBN-13: 978-0-13-325448-8. Also Available as eText for \$67.99." *Soil Science Society of America Journal* 80 (5): 1428–28. <https://doi.org/10.2136/sssaj2016.0005br>.
- Hengl, Tomislav, Jorge Mendes de Jesus, Gerard B. M. Heuvelink, Maria Ruiperez Gonzalez, Milan Kilibarda, Aleksandar Blagotić, Wei Shangguan, et al. 2017. "SoilGrids250m: Global Gridded Soil Information Based on Machine Learning." Edited by Ben Bond-Lamberty. *PLOS ONE* 12 (2): e0169748. <https://doi.org/10.1371/journal.pone.0169748>.
- Li, Jin, Andrew D. Heap, Anna Potter, and James J. Daniell. 2011. "Application of Machine Learning Methods to Spatial Interpolation of Environmental Variables." *Environmental Modelling & Software* 26 (12): 1647–59. <https://doi.org/10.1016/j.envsoft.2011.07.004>.
- McBratney, A. B, M. L Mendonça Santos, and B Minasny. 2003. "On Digital Soil Mapping." *Geoderma* 117 (1-2): 3–52. [https://doi.org/10.1016/s0016-7061\(03\)00223-4](https://doi.org/10.1016/s0016-7061(03)00223-4).