# Methods examples

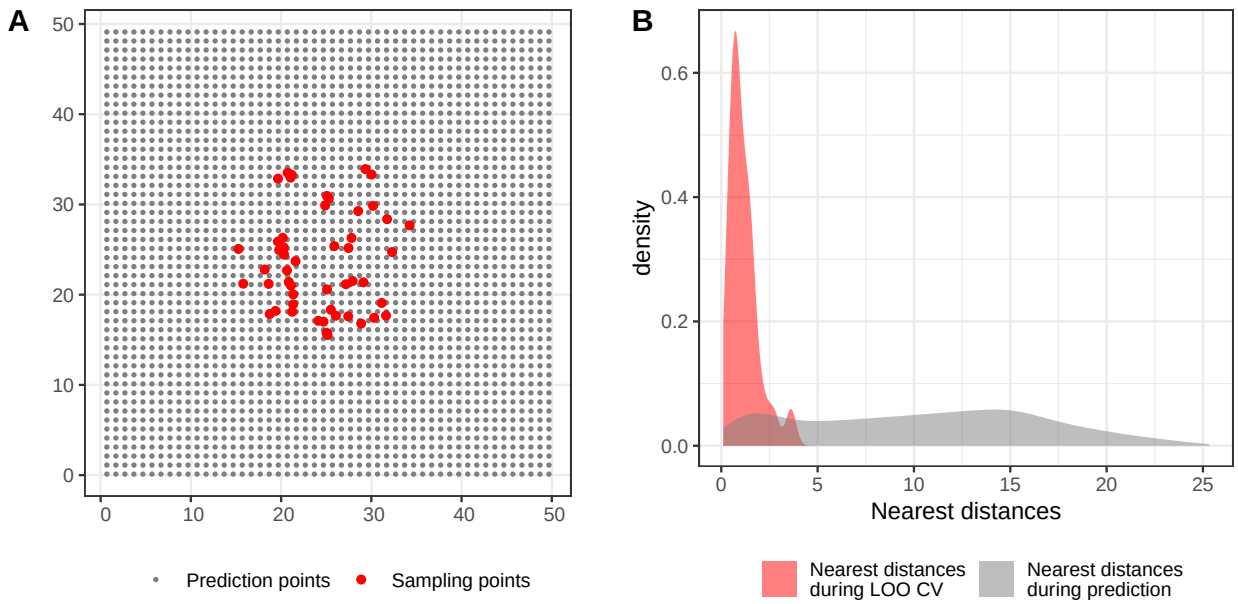
Carles Milà

2022-02-21

## 1. Introduction

This section contains the code to run the example of section *1. Introduction*, where we simulate one set of 50 clustered samples in a 50x50 target prediction grid (i.e. 2500 regular target points); from which we calculate the nearest neighbour distance for each training point, and the distance from each target point to the nearest sampling point, and compare their densities.

```
# Define study area and target points
set.seed(1234)
mat <- matrix(c(0,0,0,50,50,50,50,0,0,0), ncol=2,  byrow=TRUE)
target_window <- sf::st_polygon(list(mat))
target_points <- st_sample(target_window, 50*50, type="regular")
target_points <- st_sf(type="Prediction points", geom=target_points)
sample_window <- st_buffer(st_point(c(25,25)), 10)
sample_points <- st_sample(sample_window, 50, type="random")
sample_points <- st_sf(type="Sampling points", geom=sample_points)
all_points <- rbind(target_points, sample_points)

# Nearest distances between target and train points
target_train_dist <- st_distance(target_points, sample_points)
diag(target_train_dist) <- NA
target_train_dist <- apply(target_train_dist, 1, function(x) min(x, na.rm=T))
train_train_dist <- st_distance(sample_points)
diag(train_train_dist) <- NA
train_train_dist <- apply(train_train_dist, 1, function(x) min(x, na.rm=T))
all_dists <- rbind(data.frame(ndist = train_train_dist,
                              measure = "Nearest distances\nduring LOO CV"),
                   data.frame(ndist = target_train_dist,
                              measure = "Nearest distances\nduring prediction"))
```

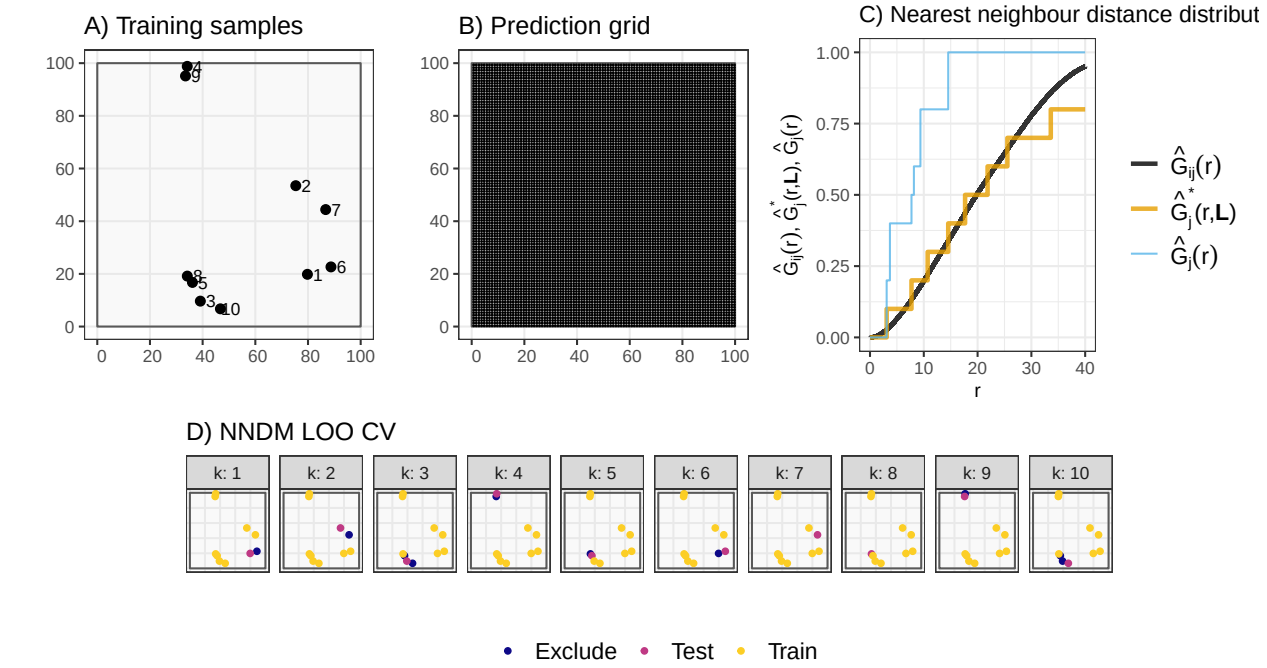



## 2.1 Nearest Distance Matching cross-validation for spatial prediction

This section contains the code to run the examples of section *2.1 Nearest Distance Matching cross-validation for spatial prediction*, where 1) we simulate a set of 10 points in a 100x100 target prediction grid, run the NNDM algorithm and show how LOO CV and NNDM LOO CV would be performed, and 2) we simulate three sets of points in a 100x100 target prediction grid and run the NNDM algorithm.

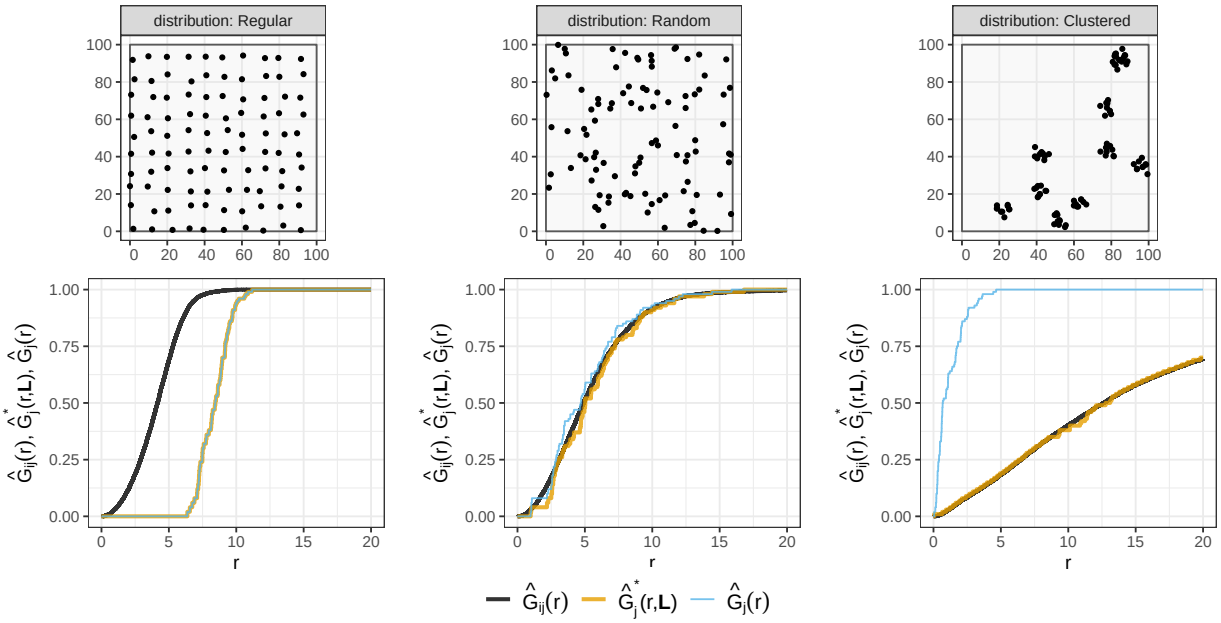
```
# Define study area and target points
mat <- matrix(c(0,0,0,100,100,100,100,0,0,0), ncol=2,  byrow=TRUE)
sampling_window <- sf::st_polygon(list(mat))
p_target <- st_sample(sampling_window, 100*100, type="regular")

# Sample plot
set.seed(123456)
p_clust <- clustered_sample(sampling_window, 10, 5, 15)
p_clust$ID <- 1:10
train_plot <- ggplot() +
  geom_sf(data=sampling_window, alpha=0.2) +
  geom_sf(data=p_clust, size=2) +
  geom_sf_text(data=p_clust, aes(label=ID), size = 3, nudge_x = 4, nudge_y = 0) +
  xlab("") + ylab("") + ggtitle("A) Training samples") +
  theme_bw()

# Prediction plot
pred_plot <- ggplot() +
  geom_sf(data=sampling_window, alpha=0.2) +
  geom_sf(data=p_target, size=0.05) +
  xlab("") + ylab("") + ggtitle("B) Prediction grid") +
  theme_bw()

# NDM plot
nndm_clust <- nndm(p_clust, p_target, 40, 0.5)
nndm_plot <- plot(nndm_clust)  +
```

```
  ggtitle("C) Nearest neighbour distance distributions") +
  theme(title = element_text(size=10))

# NNDM LOO CV plot
nndm_data <- st_sfc()
for(k in 1:10){
  nndm_it <- p_clust
  nndm_it$k <- k
  nndm_it$use <- ifelse(p_clust$ID == nndm_clust$indx_test[[k]], "Test",
                        ifelse(p_clust$ID %in% nndm_clust$indx_train[[k]], "Train", "Exclude"))
  nndm_data <- rbind(nndm_data, nndm_it)
}
nndmloo_plot <- ggplot() +
  geom_sf(data=sampling_window, alpha=0.2) +
  geom_sf(data=nndm_data, size=1.2, aes(colour = use)) +
  scale_colour_manual(values=c("#0D0887FF", "#BF3984FF", "#FCCE25FF"))+
  facet_wrap(~k, nrow=1, labeller = "label_both") +
  xlab("") + ylab("") + ggtitle("D) NNDM LOO CV") +
  theme_bw() +
  theme(legend.position = "bottom", axis.text = element_blank(),
        axis.ticks = element_blank(), legend.text = element_text(size=12)) +
  labs(colour="")

all_plot <- plot_grid(plot_grid(train_plot, pred_plot, nndm_plot, nrow = 1,
                                rel_widths = c(0.3, 0.3, 0.4)),
                      nndmloo_plot, nrow = 2, rel_heights = c(0.6, 0.4))
all_plot
```




```
# save_plot("figures/example_LOO_nndmCV.png", all_plot, base_height=6, base_asp = 1.8)
```

```r
# Define study area and target points
mat <- matrix(c(0,0,0,100,100,100,100,0,0,0), ncol=2,  byrow=TRUE)
sampling_window <- sf::st_polygon(list(mat))
p_target <- st_sample(sampling_window, 100*100, type="regular")

# Simulate 3 sets of 100 training points with different distributions
set.seed(1234)
p_reg <- sim1_samples(100, "sregular", sampling_window)
p_rand <- sim1_samples(100, "random", sampling_window)
p_clust <- sim1_samples(100, "sclust", sampling_window)

# Run NDM
ndm_reg <- nndm(p_reg, p_target, 20, 0.5)
ndm_rand <- nndm(p_rand, p_target, 20, 0.5)
ndm_clust <- nndm(p_clust, p_target, 20, 0.5)

p_11 <- ggplot() +
  geom_sf(data=sampling_window, alpha=0.2) +
  geom_sf(data=mutate(p_reg, distribution="Regular"), size=1) +
  facet_wrap(~distribution, labeller="label_both") +
  theme_bw()
p_21 <- ggplot() +
  geom_sf(data=sampling_window, alpha=0.2) +
  geom_sf(data=mutate(p_rand, distribution="Random"), size=1) +
  facet_wrap(~distribution, labeller="label_both") +
  theme_bw()
p_31 <- ggplot() +
  geom_sf(data=sampling_window, alpha=0.2) +
  geom_sf(data=mutate(p_clust, distribution="Clustered"), size=1) +
  facet_wrap(~distribution, labeller="label_both") +
  theme_bw()
p_legend <-  plot(ndm_reg) +
  theme(legend.position = "bottom")
p_legend <- get_legend(p_legend)
p_all <- plot_grid(p_11, p_21, p_31,
                   plot(ndm_reg) +
                     theme(legend.position = "none",
                           plot.margin = unit(c(0.1,0.6,0.1,0.1), "cm")),
                   plot(ndm_rand) + theme(legend.position = "none",
                           plot.margin = unit(c(0.1,0.6,0.1,0.1), "cm")),
                   plot(ndm_clust) + theme(legend.position = "none",
                           plot.margin = unit(c(0.1,0.6,0.1,0.1), "cm")),
                   NULL, p_legend, NULL,
                   ncol=3, nrow=3, rel_heights = c(1,1.1,0.15))
p_all
```




```
# save_plot("figures/example_nndmCV.png", p_all, base_height=5, base_asp=1.8)
```