TMA4250 Spatial Statistics
Project 1 - Random Fields and Gaussian Random Fields

Christian Moen, Jim Totland

2/3/2022

Problem 1

a)

The positive semi-definite (PSD) property of the correlation function can be stated as follows. $\forall m \in \mathbb{Z}_+$, $\forall a_1, \ldots, a_m \in \mathbb{R}$ and $\forall \boldsymbol{s}_1, \ldots, \boldsymbol{s}_m \in \mathcal{D}$, we have

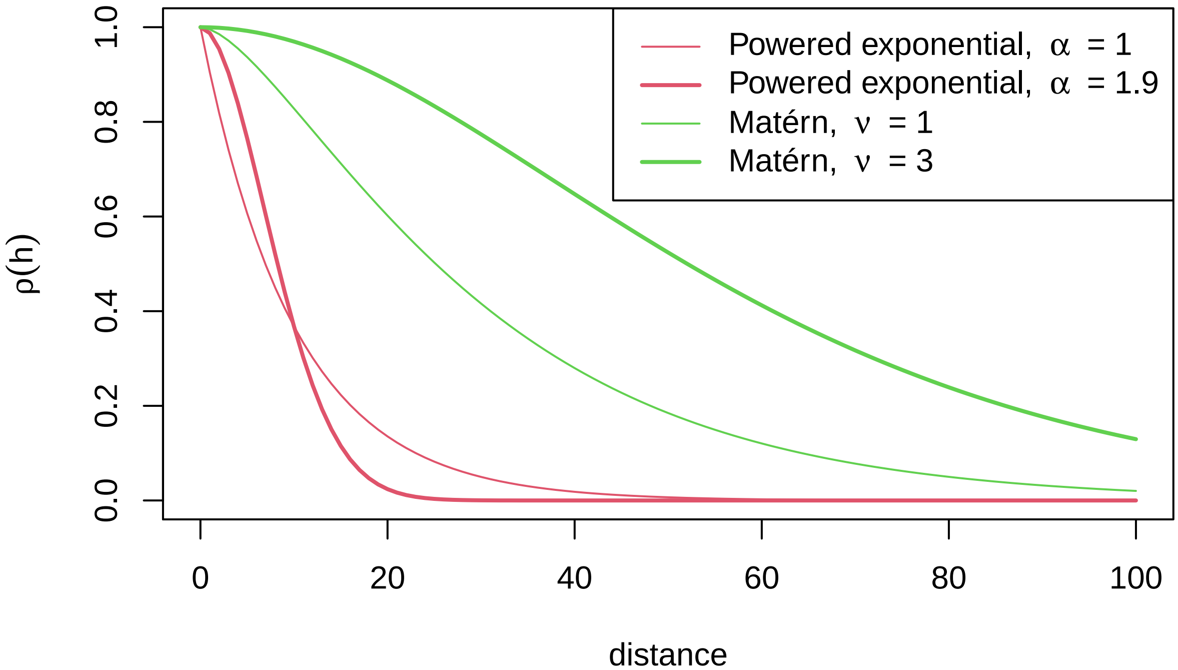
$$\sum_{i=1}^{m} \sum_{j=1}^{m} a_i a_j \rho(\boldsymbol{s}_i, \boldsymbol{s}_j) \geq 0.$$

To explain why this requirement is necessary, we observe that (in this case) $\rho(\boldsymbol{s}_i, \boldsymbol{s}_j) = \sigma^{-2} c(\boldsymbol{s}_i, \boldsymbol{s}_j)$, where c is the covariance function. Consequently,

$$\begin{aligned} \sum_{i=1}^{m} \sum_{j=1}^{m} a_i a_j \rho(\boldsymbol{s}_i, \boldsymbol{s}_j) &= \sigma^{-2} \sum_{i=1}^{m} \sum_{j=1}^{m} a_i a_j c(\boldsymbol{s}_i, \boldsymbol{s}_j) \\ &= \sigma^{-2} \mathrm{Var}\left[\sum_{i=1}^{m} a_i X(\boldsymbol{s_i})\right]. \end{aligned}$$

Since the variance must be non-negative, it is clear that the PSD property above must be satisfied. Below, the different correlation functions are illustrated. ingen grunn til to plots, fordi ulik varians gir samme korrelasjon?!

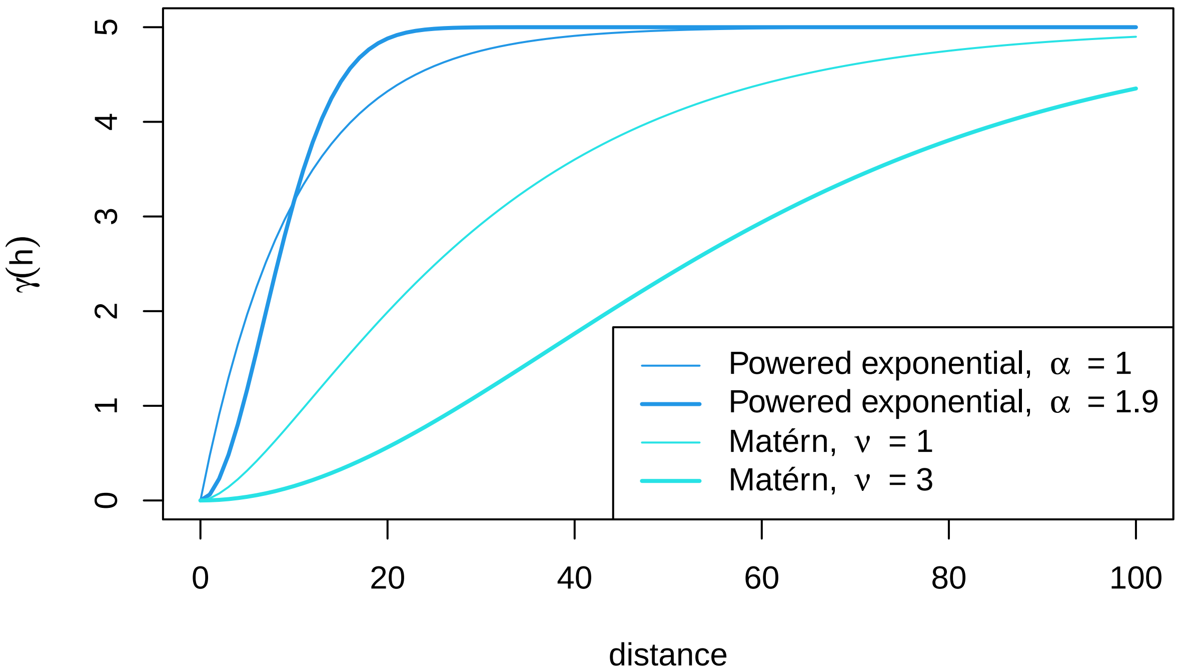
```
curve(1/5 * cov.spatial(x, cov.mod = "powered.exponential", cov.pars = c(5, 10), kappa = 1),
      from = 0, to = 100, col = 2, xlab = "distance", ylab = expression(rho(h)))
curve(1/5 *cov.spatial(x, cov.mod = "powered.exponential", cov.pars = c(5, 10), kappa = 1.9),
      from = 0, to = 100, col = 2, lwd = 2, add = TRUE)
curve(1/5 * cov.spatial(x, cov.mod = "matern", cov.pars = c(5, 20), kappa = 1),
      from = 0, to = 100, col = 3, add = TRUE)
curve(1/5 *cov.spatial(x, cov.mod = "matern", cov.pars = c(5, 20), kappa = 3),
      from = 0, to = 100, col = 3, lwd = 2, add = TRUE)
legend("topright", c(expression("Powered exponential, " ~alpha~ " = 1"),
                     expression("Powered exponential, " ~alpha~ " = 1.9"),
                     expression("Matérn, " ~nu~ " = 1"), expression("Matérn, " ~nu~ " = 3")),
       col = c(2,2,3,3), lwd = c(1,2,1,2))
```



Next, we plot the semi-variograms. to figurer her kanskje? litt unødvendig? spør om dette.

```r
semi.variogram <- function(x, ...){
  return(cov.spatial(0, ...) - cov.spatial(x, ...))
}

curve(semi.variogram(x, cov.mod = "powered.exponential", cov.pars = c(5, 10), kappa = 1),
      from = 0, to = 100, col = 4, xlab = "distance", ylab = expression(gamma(h)))
curve(semi.variogram(x, cov.mod = "powered.exponential", cov.pars = c(5, 10), kappa = 1.9),
      from = 0, to = 100, col = 4, lwd = 2, add = TRUE)
curve(semi.variogram(x, cov.mod = "matern", cov.pars = c(5, 20), kappa = 1),
      from = 0, to = 100, col = 5, add = TRUE)
curve(semi.variogram(x, cov.mod = "matern", cov.pars = c(5, 20), kappa = 3),
      from = 0, to = 100, col = 5, lwd = 2, add = TRUE)
legend("bottomright", c(expression("Powered exponential, " ~alpha~ " = 1"),
                        expression("Powered exponential, " ~alpha~ " = 1.9"),
                        expression("Matérn, " ~nu~ " = 1"),
                        expression("Matérn, " ~nu~ " = 3")),
       col = c(4,4,5,5), lwd = c(1,2,1,2))
```



b)

By the definition of a GRF, $\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$. The parameters are calculated from the mean- and covariance function of the GRF, such that $\boldsymbol{\mu} = \mathbf{0}$ and $\Sigma_{ij} = \sigma^2 \rho(\|i - j\|)$. First, we create grids which span all the parameter combinations and summarize them in two tables.

```
# Parameters
sigma2 <- c(1, 5)
alpha <- c(1, 1.9)
nu <- c(1, 3)
a.exp <- 10
a.matern <- 20

params.exp <- expand.grid(sigma2, alpha, a.exp)
params.exp <- cbind(params.exp, 1:4)
colnames(params.exp) <- c("sigma2", "alpha", "a.exp", "combination")

params.matern <- expand.grid(sigma2, alpha, a.matern)
params.matern <- cbind(params.matern, 1:4)
colnames(params.matern) <- c("sigma2", "nu", "a.matern", "combination")

knitr::kable(params.exp)
```

sigma2	alpha	a.exp	combination
1	1.0	10	1
5	1.0	10	2
1	1.9	10	3
5	1.9	10	4

```r
knitr::kable(params.matern)
```

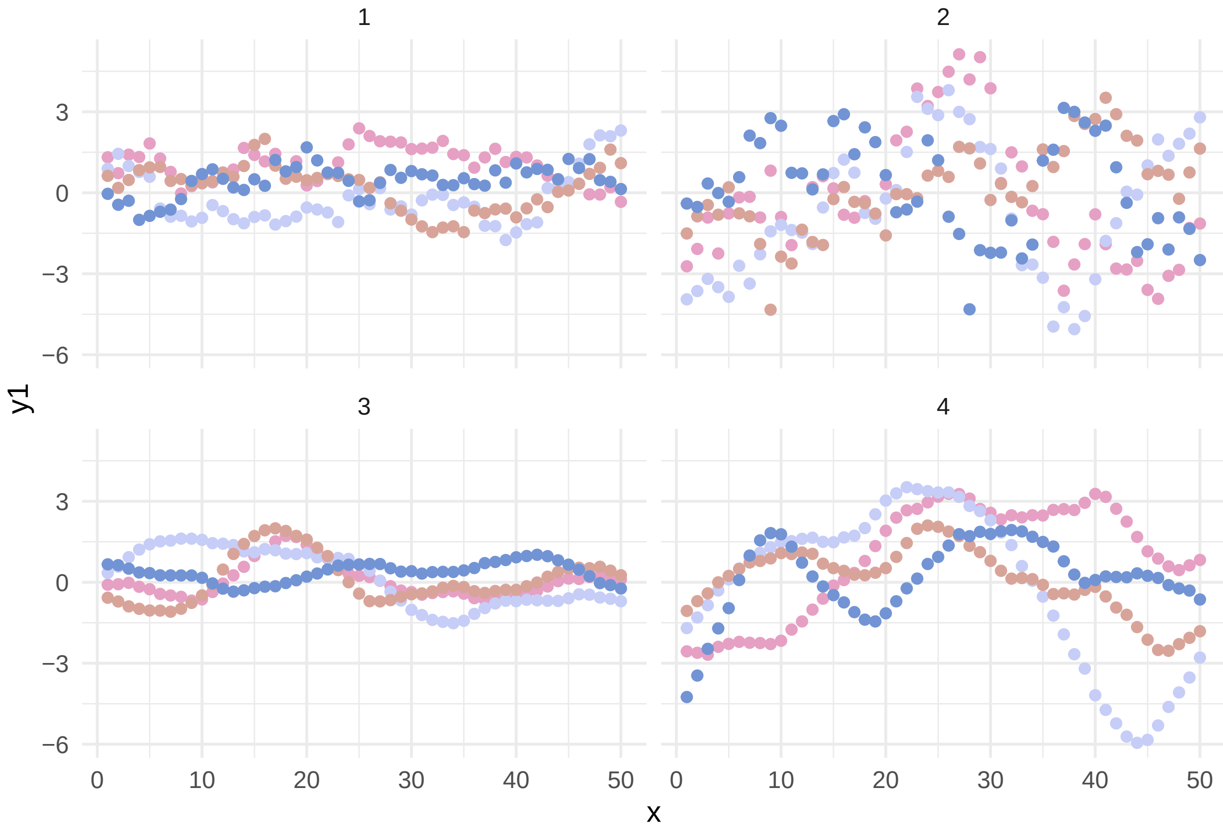
sigma2	nu	a.matern	combination
1	1.0	20	1
5	1.0	20	2
1	1.9	20	3
5	1.9	20	4

Then, we simulate the 4 realizations with the powered exponential covariance function for all the parameter combinations.

```r
n <- 50 # Number of grid points
D.tilde <- 1:n # Grid

df.exp <- data.frame(x = rep(D.tilde, 4),
                     y1 = rep(NA, 4), y2 = rep(NA, 4), y3 = rep(NA, 4), y4 = rep(NA, 4),
                     combination = rep(NA, 4 * n))
for(i in 1:nrow(params.exp)){
  mu <- rep(0, n)
  Sigma <- cov.spatial(as.matrix(dist(expand.grid(D.tilde))),
                     cov.mod = "powered.exponential",
                     cov.pars = c(params.exp$sigma2[i], params.exp$a.exp[i]),
                     kappa = params.exp$alpha[i])
  X <- mvrnorm(4, mu, Sigma)
  df.exp$y1[((i-1)*n + 1):(i*50)] = X[1, ]
  df.exp$y2[((i-1)*n + 1):(i*50)] = X[2, ]
  df.exp$y3[((i-1)*n + 1):(i*50)] = X[3, ]
  df.exp$y4[((i-1)*n + 1):(i*50)] = X[4, ]
  df.exp$combination[((i-1)*n + 1):(i*50)] = rep(i, 50)
}

palette <- wes_palette("GrandBudapest2", n = 4)
df <- data.frame(t(X), D = D.tilde)
ggplot(df.exp, aes(x = x)) + geom_point(aes(y = y1), color = palette[1]) +
  geom_point(aes(y = y2), color = palette[2]) +
  geom_point(aes(y = y3), color = palette[3]) +
  geom_point(aes( y = y4), color = palette[4]) +
  facet_wrap( ~combination, nrow = 2) + theme_minimal()
```

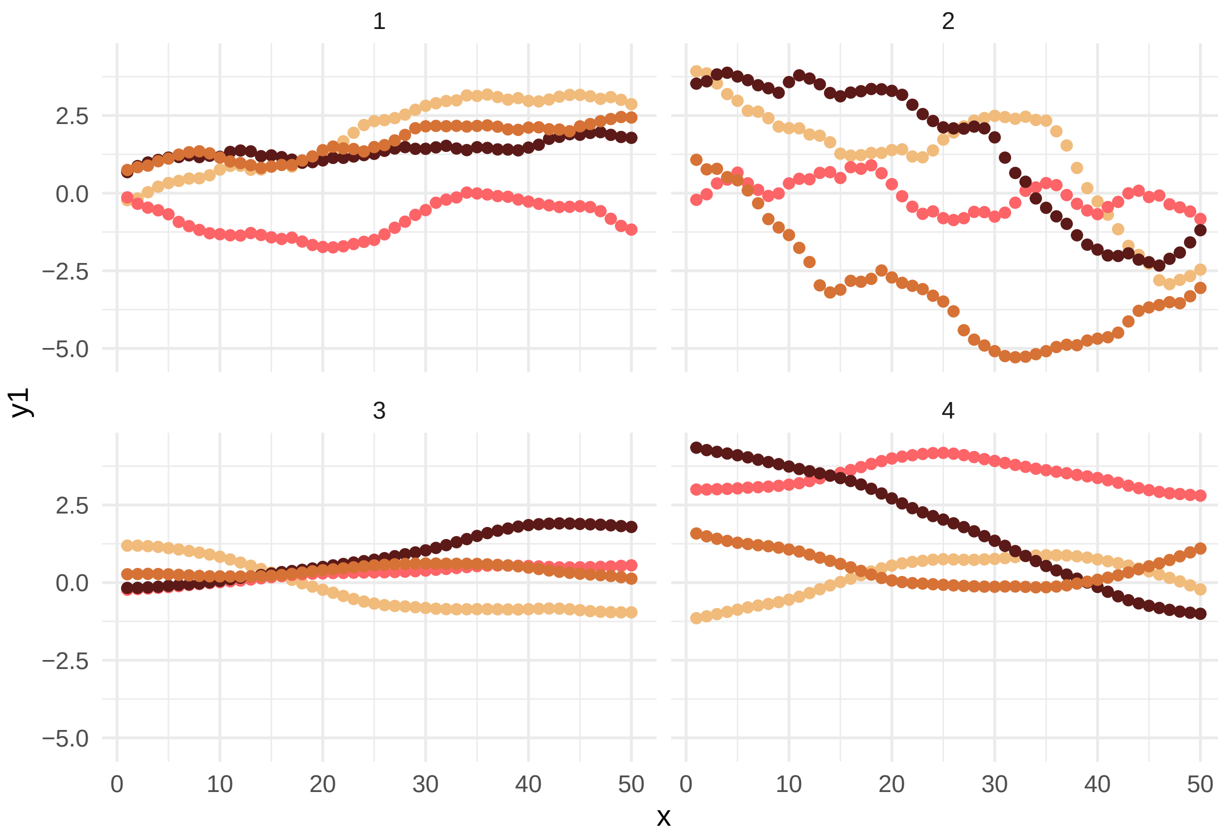


Unsurprisingly, we see that higher variance (facet 2 and 4) leads to more variability in the simulations. When the power parameter $\alpha = 1$, the realizations are much more jagged and less smooth compared to when $\alpha = 1.9$ (facet 3 and 4). We follow the same procedure to simulate with a Matérn covariance function.

```
df.matern <- data.frame(x = rep(D.tilde, 4),
                        y1 = rep(NA, 4), y2 = rep(NA, 4), y3 = rep(NA, 4), y4 = rep(NA, 4),
                        combination = rep(NA, 4 * n))

for(i in 1:nrow(params.matern)){
  mu <- rep(0, n)
  Sigma <- cov.spatial(as.matrix(dist(expand.grid(D.tilde))),
                       cov.mod = "matern",
                       cov.pars = c(params.matern$sigma2[i], params.matern$a.matern[i]),
                       kappa = params.matern$nu[i])
  X <- mvrnorm(4, mu, Sigma)
  df.matern$y1[((i-1)*n + 1):(i*50)] = X[1, ]
  df.matern$y2[((i-1)*n + 1):(i*50)] = X[2, ]
  df.matern$y3[((i-1)*n + 1):(i*50)] = X[3, ]
  df.matern$y4[((i-1)*n + 1):(i*50)] = X[4, ]
  df.matern$combination[((i-1)*n + 1):(i*50)] = rep(i, 50)
}

palette <- wes_palette("GrandBudapest1", n = 4)
ggplot(df.matern, aes(x = x)) + geom_point(aes(y = y1), color = palette[1]) +
  geom_point(aes(y = y2), color = palette[2]) +
  geom_point(aes(y = y3), color = palette[3]) +
  geom_point(aes( y = y4), color = palette[4]) +
  facet_wrap( ~combination, nrow = 2) + theme_minimal()
```



We also see here that higher variance (facet 2 and 4) leads to more variability in the simulations. A higher smoothness parameter, ν, also leads to smoother realizations, as expected.