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. $\forall m \in \mathbb{Z}_+$, $\forall a_1, \dots, a_m \in \mathbb{R}$ and $\forall \boldsymbol{s}_1, \dots, \boldsymbol{s}_m \in \mathcal{D}$, we have

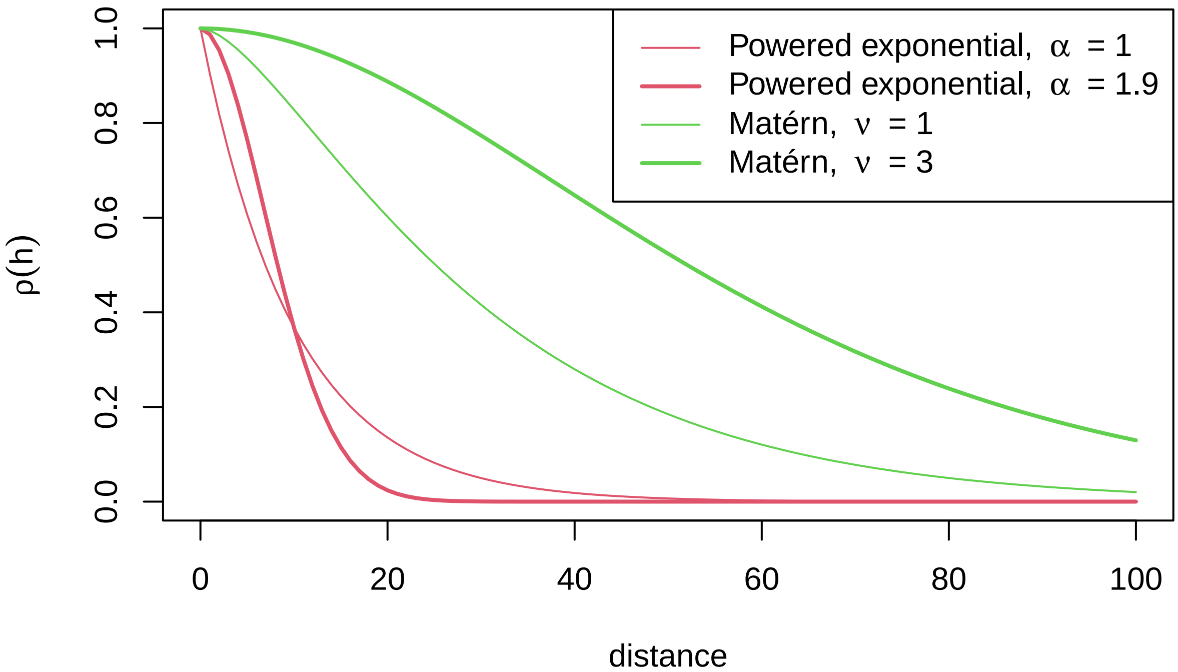
$$\sum_{i=1}^{m} \sum_{j=1}^{m} a_i a_j \rho(\boldsymbol{s}_i, \boldsymbol{s}_j) \geq 0.$$

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?!

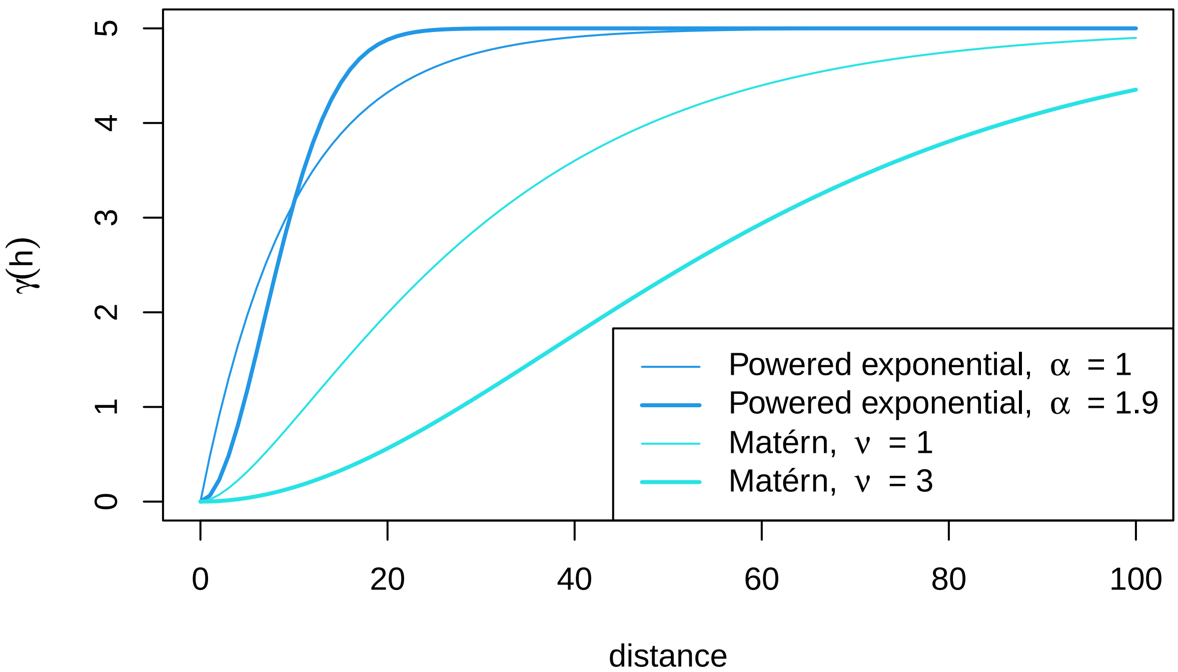
```
curve(1/5 * cov.spatial(x, cov.mod = "powered.exponential", cov.pars = c(5, 10), kappa = 1),
      from = 0, to = 100, col = 2, xlab = "distance", ylab = expression(rho(h)))
curve(1/5 *cov.spatial(x, cov.mod = "powered.exponential", cov.pars = c(5, 10), kappa = 1.9),
      from = 0, to = 100, col = 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.

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

```r
# 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, nu, 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

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

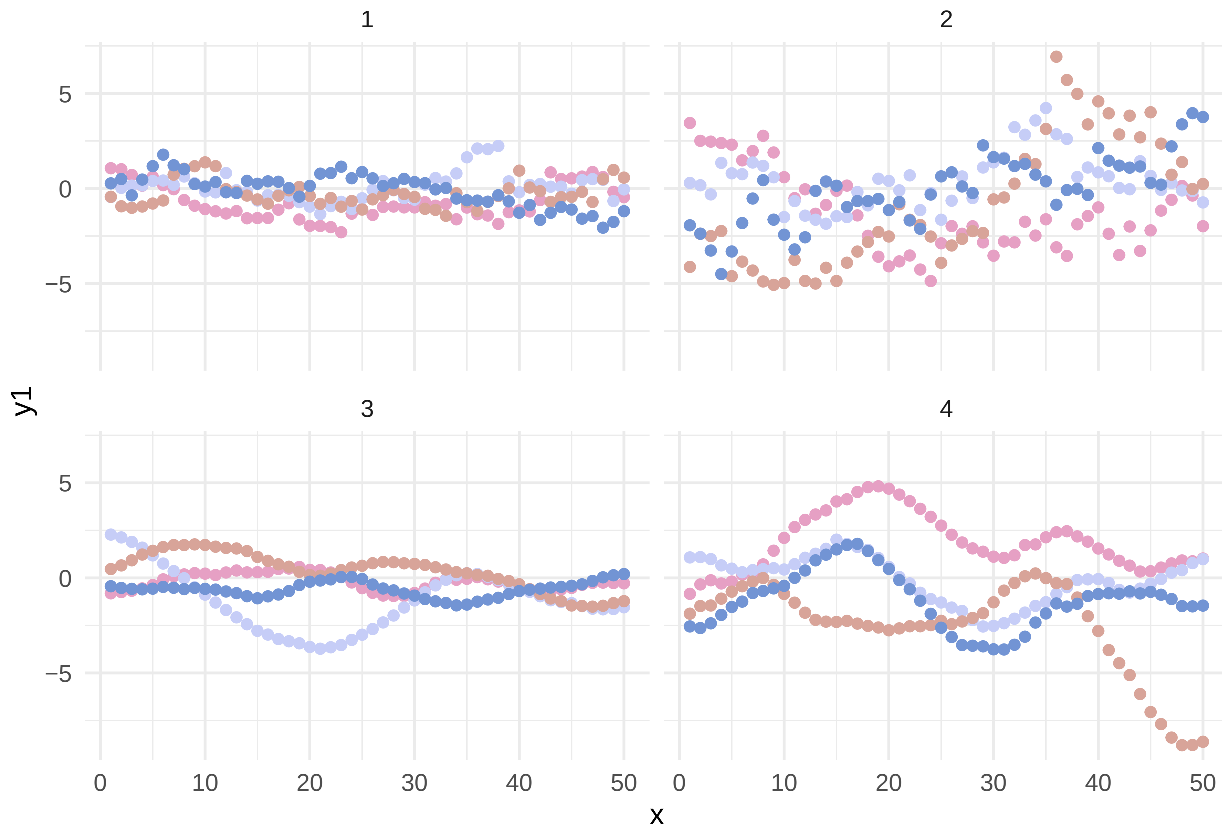
sigma2	nu	a.matern	combination
1	1	20	1
5	1	20	2
1	3	20	3
5	3	20	4

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

```
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()
```



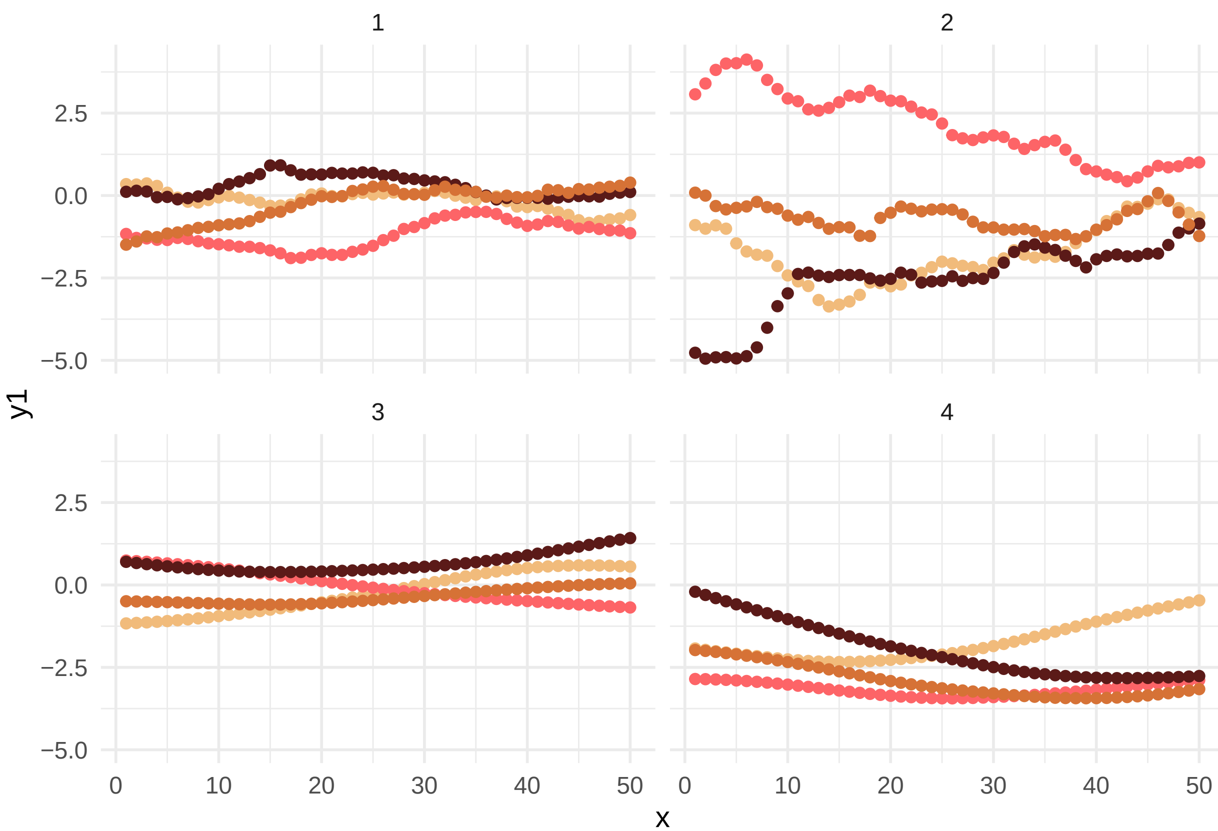
Unsurprisingly, we see that higher variance (facet 2 and 4) leads to more variability in the simulations. When the power parameter $\alpha = 1$, the realizations are much more jagged and less smooth compared to when $\alpha = 1.9$ (facet 3 and 4). We follow the same procedure to simulate with a Matérn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

c)

We plan to observe Y_1, Y_2 and Y_3 at locations $s_1 = 10$, $s_2 = 25$ and $s_3 = 30$, respectively. The observation model is given by

$$Y_i = X(s_i) + \varepsilon_i, \quad i = 1, 2, 3,$$

where $\varepsilon_1, \varepsilon_2, \varepsilon_3 \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_N^2)$ and independent of X. We let $\boldsymbol{Y} = (Y_1, Y_2, Y_3)^T$. Since $\boldsymbol{Y}$ is a linear combination of multivariate normal distributions, it is multivariate normally distributed itself, with mean

$$\mathrm{E}[\boldsymbol{Y}] = \mathbf{0},$$

and not right

$$\mathrm{Cov}[\boldsymbol{Y}] = \mathrm{diag}(\sigma_N^2) + \Sigma_X.$$

We consider the Matérn covariance model with $\sigma^2 = 5$ and $\nu = 3$ and pick one realization from b). When $\sigma_N^2 = 0$, $Y_i = X(s_i)$, such that the realization $\boldsymbol{Y} = \boldsymbol{y}$ becomes

```r
s = c(10, 25, 30) # Positions of interest
realization = filter(df.matern, combination == 4)[, 2] # First realization over entire grid
(y = realization[s])
```

```
## [1] -2.258018 -2.110430 -1.854015
```

When $\sigma_N^2 = 0.25$, the realization $\boldsymbol{y}$ becomes

```r
y + rnorm(3, mean = 0, sd = sqrt(0.25))
```

```
## [1] -2.057358 -2.800639 -1.319296
```

d)

We now consider $\boldsymbol{X}|\boldsymbol{Y} = \boldsymbol{y}$. To find its distribution and corresponding parameters, we consider

$$\begin{pmatrix} \boldsymbol{X} \\ \boldsymbol{Y} \end{pmatrix} \sim \mathcal{N}_{50+3}(\boldsymbol{0}, \Sigma),$$

where

$$\Sigma = \begin{pmatrix} \Sigma_{XX} & \Sigma_{XY} \\ \Sigma_{YX} & \Sigma_{YY} \end{pmatrix}$$

Then we know that $\boldsymbol{X}|\boldsymbol{Y} = \boldsymbol{y} \sim \mathcal{N}(\boldsymbol{\mu}_{X|Y}, \Sigma_{X|Y})$, where

$$\begin{aligned} \Sigma_{X|Y} &= \Sigma_{XX} - \Sigma_{XY}\Sigma_{YY}^{-1}\Sigma_{YX}, \\ \boldsymbol{\mu}_{X|Y} &= \mathrm{E}[\boldsymbol{X}] + \Sigma_{XY}\Sigma_{YY}^{-1}\left(\boldsymbol{y} - \mathrm{E}[\boldsymbol{Y}]\right) = \Sigma_{XY}\Sigma_{YY}^{-1}\boldsymbol{y} \end{aligned}$$

Let $\widetilde{\boldsymbol{X}} = (X(s_1), X(s_2), X(s_3))^T$ and $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \varepsilon_3)^T$. We note that

$$\Sigma_{YY} = \mathrm{Cov}(\boldsymbol{Y}) = \mathrm{Cov}(\widetilde{\boldsymbol{X}} + \boldsymbol{\varepsilon}) = \mathrm{Cov}(\widetilde{\boldsymbol{X}}) + \sigma_N^2 I_3.$$

For prediction we use the BLUP, which is given by

$$\hat{\boldsymbol{X}} = \boldsymbol{\mu}_{X|Y},$$

and we use the pointwise prediction variances, $\mathrm{diag}\left(\Sigma_{X|Y}\right)$ to construct the confidence intervals.

```r
D.tilde <- 1:50
Sigma_XX <- cov.spatial(as.matrix(dist(expand.grid(D.tilde))),
                        cov.mod = "matern",
                        cov.pars = c(5, 20),
                        kappa = 3)

Sigma_XY <- Sigma_XX[ , s]
# Construct Sigma_YY
H <- matrix(rep(0, 3*50), nrow = 50, ncol = 3)
H[10, 1] = 1
H[25, 2] = 1
H[30, 3] = 1
dim(H)
```

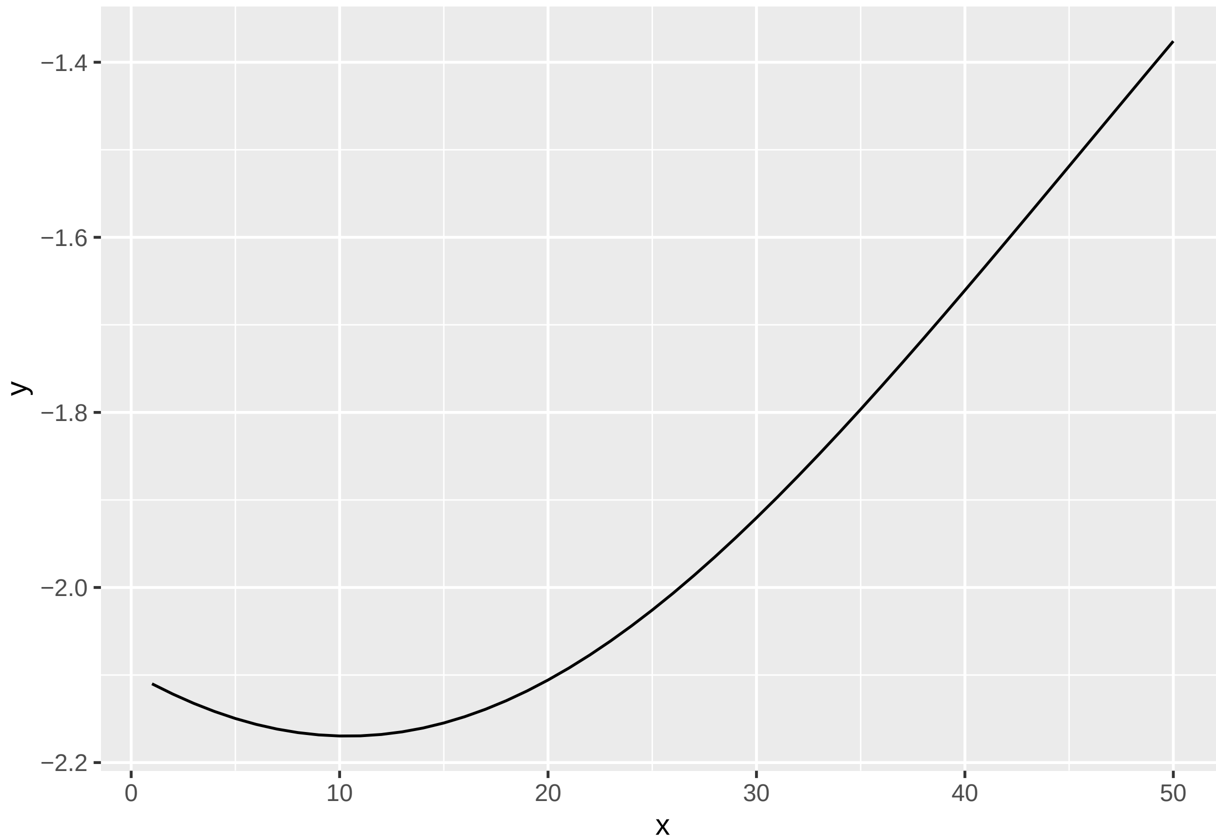
```
## [1] 50  3
```

```r
dim(Sigma_XX)
```

```
## [1] 50 50
```

```r
Sigma_YY <- t(H) %*% Sigma_XX %*% H + 0.25*diag(3) # sigma_N^2 = 0.25
BLUP <- Sigma_XY %*% solve(Sigma_YY) %*% y
var.BLUP <- 2

point.variance <- diag(Sigma_XX)
ggplot(data.frame(x = D.tilde, y = BLUP), aes(x)) + geom_line(aes(y = y))
```



e)

Before coding up the simulation, we note that $\boldsymbol{Y}|\boldsymbol{X} = \boldsymbol{x} \sim \mathcal{N}_3(\mu_Y, \sigma_N^2 I_3)$, where ...

Problem 3: Parameter Estimation

We consider the stationary GRF $\{X(\boldsymbol{s});\ \boldsymbol{s} \in \mathcal{D} = [1, 30]^2 \subset \mathbb{R}\}$ with

$$\mathrm{E}[X(\boldsymbol{s})] = \mu = 0, \quad \boldsymbol{s} \in \mathcal{D},$$
$$\mathrm{Var}[X(\boldsymbol{s})] = \sigma^2, \quad \boldsymbol{s} \in \mathcal{D},$$
$$\mathrm{Corr}[X(\boldsymbol{s}), X(\boldsymbol{s}')] = \exp(-\|\boldsymbol{s} - \boldsymbol{s}'\|/a), \quad \boldsymbol{s}, \boldsymbol{s}' \in \mathcal{D}$$

We let $\tilde{\mathcal{D}} = \{1, 2, \ldots, 30\}^2$ be a regular grid of $\mathcal{D}$, and set the marginal variance $\sigma^2 = 2$ and the spatial scale $a = 3$.

```
sigma2 <- 2
a <- 3

D.tilde <- expand.grid(1:30,1:30)
H <- as.matrix(dist(D.tilde))
Sigma <- cov.spatial(H,
                     cov.mod = "exponential",
                     cov.pars = c(sigma2, a))
L <- chol(Sigma)
Z <- rnorm(30*30)
X <- L %*% Sigma # mu = 0.
```