Assignment 2

Md Kamrul Hasan Khan

Due 5PM on Friday February 14, 2020

Problem 1

Make a map of the total annual precipitation (The sum of annual precipitation from columns 6-17).

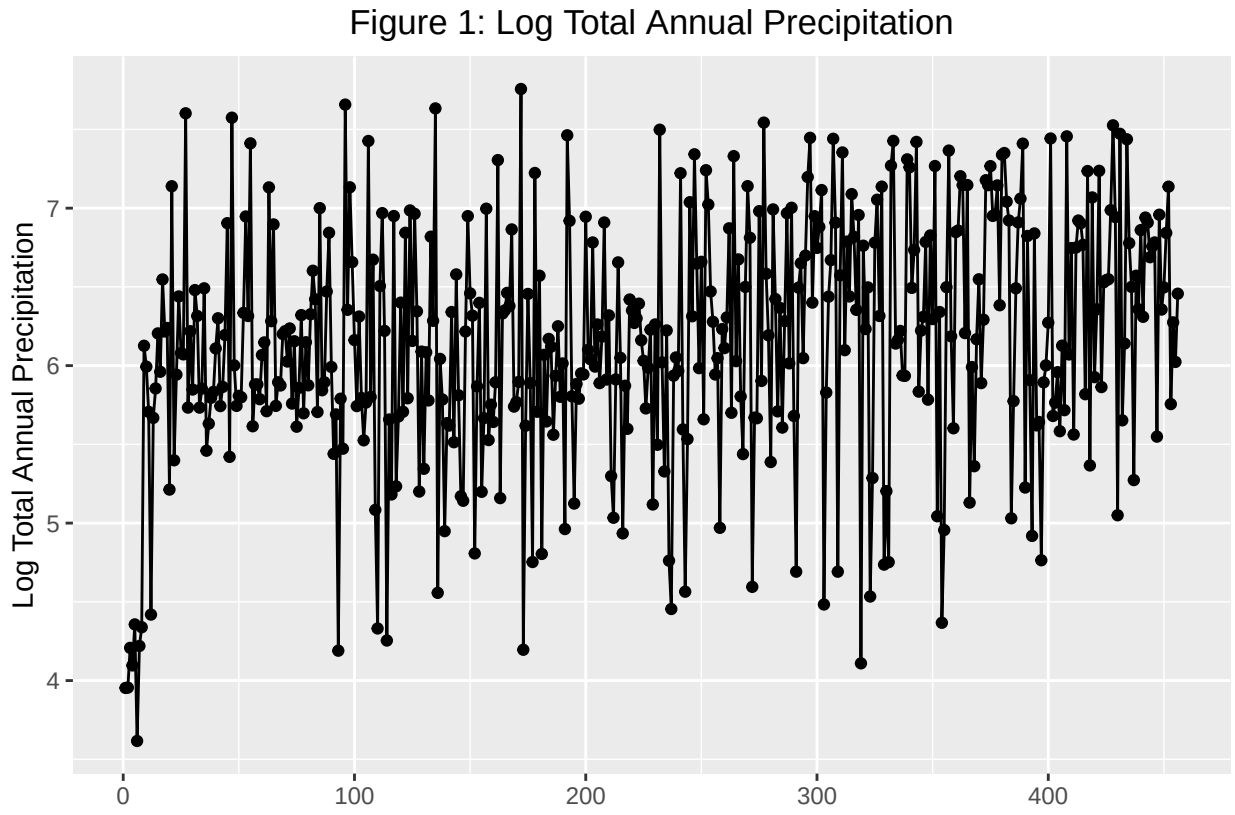
Answer:

```
if (!require("devtools")) install.packages('devtools')
#if (!require("rspatial")) devtools::install_github('rspatial/rspatial')
library(tidyverse)
library(fields)
```

```
dat <- readRDS(here::here("data","precipitation.Rds"))
dat <- dat %>%
            mutate(annual_total = rowSums(dat[ , 6:17])) %>%
            mutate(log_annual_total = log(annual_total))
glimpse(dat)
```

```
## Rows: 456
## Columns: 19
## $ ID               <chr> "ID741", "ID743", "ID744", "ID753", "ID754", "ID758",~
## $ NAME             <chr> "DEATH VALLEY", "THERMAL/FAA AIRPORT", "BRAWLEY 2SW",~
## $ LAT              <dbl> 36.47, 33.63, 32.96, 32.83, 33.28, 32.82, 32.76, 33.7~
## $ LONG             <dbl> -116.87, -116.17, -115.55, -115.57, -115.51, -115.67,~
## $ ALT              <int> -59, -34, -31, -18, -18, -13, -9, -6, 2, 2, 3, 3, 3, ~
## $ JAN              <dbl> 7.4, 9.2, 11.3, 10.6, 9.0, 9.8, 9.0, 16.7, 106.3, 89.~
## $ FEB              <dbl> 9.5, 6.9, 8.3, 7.0, 8.0, 1.6, 7.0, 12.1, 107.1, 88.3,~
## $ MAR              <dbl> 7.5, 7.9, 7.6, 6.1, 9.0, 3.7, 5.0, 9.2, 72.9, 72.4, 4~
## $ APR              <dbl> 3.4, 1.8, 2.0, 2.5, 3.0, 3.0, 1.0, 2.2, 32.1, 30.1, 2~
## $ MAY              <dbl> 1.7, 1.6, 0.8, 0.2, 0.0, 0.4, 1.0, 1.3, 7.6, 2.0, 4.3~
## $ JUN              <dbl> 1.0, 0.4, 0.1, 0.0, 1.0, 0.0, 0.0, 0.2, 2.2, 1.1, 1.3~
## $ JUL              <dbl> 3.7, 1.9, 1.9, 2.4, 8.0, 3.0, 2.0, 3.5, 0.6, 0.6, 0.4~
## $ AUG              <dbl> 2.8, 3.4, 9.2, 2.6, 9.0, 10.8, 9.0, 6.5, 0.6, 0.5, 1.~
## $ SEP              <dbl> 4.3, 5.3, 6.5, 8.3, 7.0, 0.2, 8.0, 6.4, 5.3, 1.4, 6.2~
## $ OCT              <dbl> 2.2, 2.0, 5.0, 5.4, 8.0, 0.0, 8.0, 3.8, 11.4, 11.6, 6~
## $ NOV              <dbl> 4.7, 6.3, 4.8, 7.7, 7.0, 3.3, 7.0, 7.3, 47.8, 45.3, 3~
## $ DEC              <dbl> 3.9, 5.5, 9.7, 7.3, 9.0, 1.4, 11.0, 7.4, 63.7, 58.0, ~
## $ annual_total     <dbl> 52.1, 52.2, 67.2, 60.1, 78.0, 37.2, 68.0, 76.6, 457.6~
## $ log_annual_total <dbl> 3.953165, 3.955082, 4.207673, 4.096010, 4.356709, 3.6~
```

```r
dat %>%
    ggplot(aes(x = 1:length(annual_total), y = log(annual_total) )) +
    geom_line()+
    geom_point() +
    xlab("")+
    ylab("Log Total Annual Precipitation")+
    ggtitle("Figure 1: Log Total Annual Precipitation") +
    theme(plot.title = element_text(hjust = 0.5))
```




```r
quilt.plot(dat$LONG, dat$LAT, dat$log_annual_total, nx = 30, ny = 30,
           xlab = "Longitude", ylab = "Latitude",
           main = "Figure 2: Log Total Annual Precipitation")
points(dat$LONG, dat$LAT, cex = .3)
```

Figure 2: Log Total Annual Precipitation

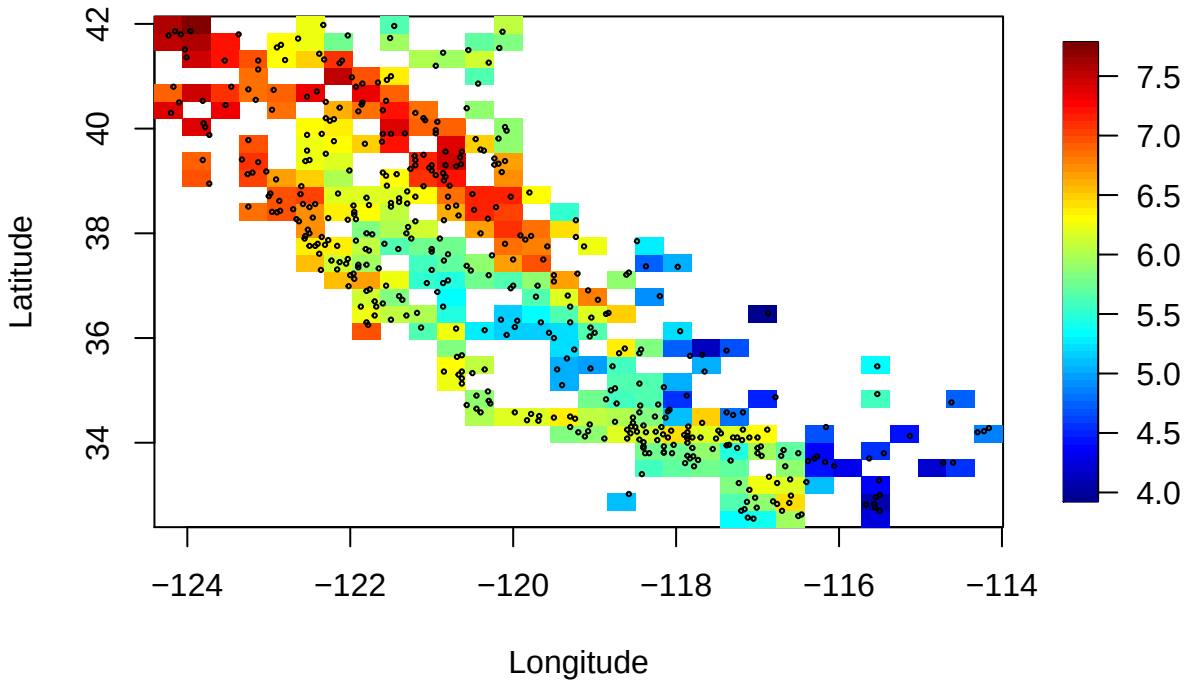


Figure 1 shows that trace plot and Figure 2 depicts the spatial representation of log total annual precipitation in California. Figure 2 shows that the north western part has more precipitation compared to the south eastern part.

Problem 2

Using the total annual rainfall, fit 8 different spatial models using the *geoR* package. 4 models with the spatial covariates (lat, long, and altitude) and 4 models without the covariates (mean is intercept-only). For each of the 4 models, fit two models with a nugget and two models without a nugget. Finally, for each model, consider a Matern covariance and an exponential covariance. Make sure you report the AIC and BIC for each of the models in a table. The 8 models are given in the table below.

```
knitr::kable(
    data.frame(
        Model      = 1:8,
        Covariates = rep(c("Spatial Coefficients", "Intercept Only"), each = 4),
        Covariance = rep(rep(c("Matern", "Exponential"), each = 2), times = 2),
        Nugget     = rep(rep(c("No Nugget", "Nugget"), times = 2), times = 2)

    )
)
```

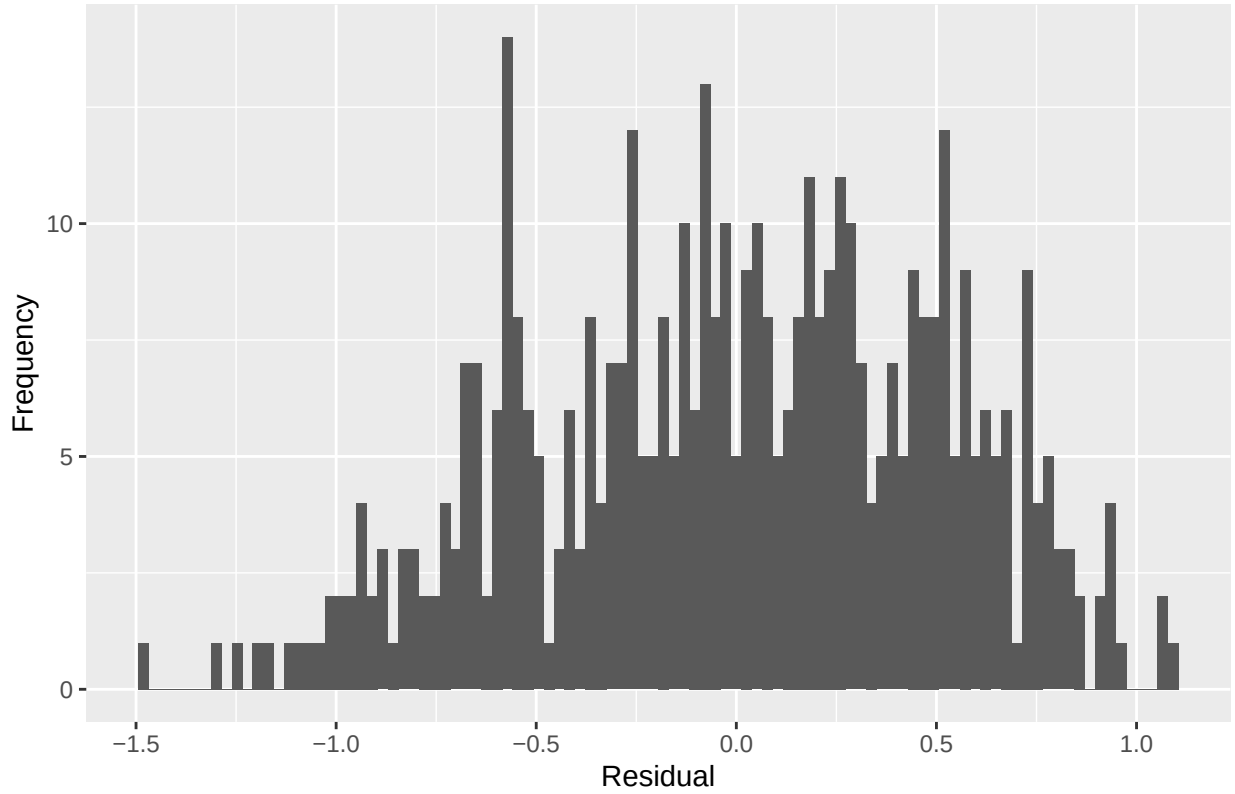
Model	Covariates	Covariance	Nugget
1	Spatial Coefficients	Matern	No Nugget
2	Spatial Coefficients	Matern	Nugget
3	Spatial Coefficients	Exponential	No Nugget
4	Spatial Coefficients	Exponential	Nugget
5	Intercept Only	Matern	No Nugget
6	Intercept Only	Matern	Nugget
7	Intercept Only	Exponential	No Nugget
8	Intercept Only	Exponential	Nugget

Answer:

Precipitation cannot take any negative values. Hence, total annual precipitation is modeled in log scale. Before fitting the data using any spatial model, I first fit the log total annual precipitation using a multiple linear regression where the predictors are latitude, longitude and altitude and check the distribution of the residuals. Figure 3 shows the histogram of the residuals. The histogram depicts that the residuals follow approximately normal distribution.

```
dat %>%
    mutate(residual = resid(lm(log_annual_total ~ LAT + LONG + ALT, data = .))) %>%
    ggplot(aes(residual)) +
    geom_histogram(bins = 100) +
    xlab("Residual") +
    ylab("Frequency") +
    ggtitle("Figure 3: Histogram of Residuals")+
    theme(plot.title = element_text(hjust = 0.5))
```

Figure 3: Histogram of Residuals




```r
#if (!require("groR")) install.packages("geoR")

library(geoR)

fit_model_1 <- likfit(
    data         = dat$log_annual_total,
    trend        = ~ dat$LAT + dat$LONG + dat$ALT,
    coords       = cbind(dat$LAT, dat$LONG),
    fix.kappa = FALSE, kappa = 1,
    cov.model    = "matern",
    ini.cov.pars = c(var(dat$log_annual_total), 1),
    fix.nugget = TRUE, nugget = 0,
    message = FALSE
)
#summary(fit_model_1)

fit_model_2 <- likfit(
    data         = dat$log_annual_total,
    trend        = ~ dat$LAT + dat$LONG + dat$ALT,
    coords       = cbind(dat$LAT, dat$LONG),
    fix.kappa = FALSE, kappa = 1,
    cov.model    = "matern",
    ini.cov.pars = c(var(dat$log_annual_total), 1),
    message = FALSE
)
#summary(fit_model_2)
```

```r
fit_model_3 <- likfit(
    data         = dat$log_annual_total,
    trend        = ~ dat$LAT + dat$LONG + dat$ALT,
    coords       = cbind(dat$LAT, dat$LONG),
    cov.model    = "exponential",
    ini.cov.pars = c(var(dat$log_annual_total), 1),
    fix.nugget = TRUE, nugget = 0,
    message = FALSE
)
#summary(fit_model_3)

fit_model_4 <- likfit(
    data         = dat$log_annual_total,
    trend        = ~ dat$LAT + dat$LONG + dat$ALT,
    coords       = cbind(dat$LAT, dat$LONG),
    cov.model    = "exponential",
    ini.cov.pars = c(var(dat$log_annual_total), 1),
    message = FALSE
)
#summary(fit_model_4)

fit_model_5 <- likfit(
    data         = dat$log_annual_total,
    coords       = cbind(dat$LAT, dat$LONG),
    fix.kappa = FALSE, kappa = 1,
    cov.model    = "matern",
    ini.cov.pars = c(var(dat$log_annual_total), 1),
    fix.nugget = TRUE, nugget = 0,
    message = FALSE
)
#summary(fit_model_5)

fit_model_6 <- likfit(
    data         = dat$log_annual_total,
    coords       = cbind(dat$LAT, dat$LONG),
    fix.kappa = FALSE, kappa = 1,
    cov.model    = "matern",
    ini.cov.pars = c(var(dat$log_annual_total), 1),
    message = FALSE
)
#summary(fit_model_6)

fit_model_7 <- likfit(
    data         = dat$log_annual_total,
    coords       = cbind(dat$LAT, dat$LONG),
    cov.model    = "exponential",
    ini.cov.pars = c(var(dat$log_annual_total), 1),
    fix.nugget = TRUE, nugget = 0,
    message = FALSE
)
#summary(fit_model_7)

fit_model_8 <- likfit(
```

```
    data         = dat$log_annual_total,
    coords       = cbind(dat$LAT, dat$LONG),
    cov.model    = "exponential",
    ini.cov.pars = c(var(dat$log_annual_total), 1),
    message = FALSE
)
#summary(fit_model_8)
```

```
AIC <- rep(0, times = 8)
BIC <- rep(0, times = 8)
log_likelihood <- rep(0, times = 8)

for(i in 1:8)
    {
    log_likelihood[i] <- round(eval(parse(text=paste0("fit_model_", i, "$loglik"))) , 3)
    AIC[i] <- round(eval(parse(text=paste0("fit_model_", i, "$AIC"))) , 3)
    BIC[i] <- round(eval(parse(text=paste0("fit_model_", i, "$BIC"))) , 3)
    }

knitr::kable(
    data.frame(
        Model      = 1:8,
        Covariates = rep(c("Spatial Coefficients", "Intercept Only"), each = 4),
        Covariance = rep(rep(c("Matern", "Exponential"), each = 2), times = 2),
        Nugget     = rep(rep(c("No Nugget", "Nugget"), times = 2), times = 2),
        Log_likelihood = log_likelihood,
        AIC = AIC,
        BIC = BIC
    ),
    caption = "Model comparison using information based criterion"
)
```

Table 2: Model comparison using information based criterion

Model	Covariates	Covariance	Nugget	Log_likelihood	AIC	BIC
1	Spatial Coefficients	Matern	No Nugget	-5.783	25.565	54.423
2	Spatial Coefficients	Matern	Nugget	1.618	12.765	45.745
3	Spatial Coefficients	Exponential	No Nugget	-7.432	26.865	51.600
4	Spatial Coefficients	Exponential	Nugget	-7.129	28.259	57.116
5	Intercept Only	Matern	No Nugget	-79.987	167.973	184.463
6	Intercept Only	Matern	Nugget	-76.913	163.826	184.439
7	Intercept Only	Exponential	No Nugget	-81.185	168.370	180.737
8	Intercept Only	Exponential	Nugget	-81.133	170.266	186.756

There are 8 candidate models to fit the log total annual precipitation. After fitting the data the models are compared using log likelihood as well as several information based criteria like AIC and BIC. Model comparison outputs are shown in Table 2. All three criteria show that adding spatial covariates (latitude, longitude and altitude) in the model improves the outputs significantly and Model 2 (model with spatial coefficients, Matern covariance and nugget) performs significantly better compared to the others because the log likelihood is maximum, and AIC as well as BIC are minimum.

Problem 3

Plot the fitted correlation function for each of the 8 models. Is there evidence of residual spatial autocorrelation? Does the inclusion of the covariates (lat, long, and altitude) have an effect on the correlation structure? **Write a few sentences to interpret the results.** Make sure you write out the answers in **clear, English sentences. Communication matters!**

Answer:

```
beta_0 <- rep(0, times = 8)
partial_sill <- rep(0, times = 8)
phi <- rep(0, times = 8)


for(i in 1:8) {
    beta_0[i] <- round(eval(parse(text=paste0("fit_model_", i, "$beta")))[1] , 3)
    partial_sill[i] <- round(eval(parse(text=paste0("fit_model_", i, "$sigmasq"))), 3)
    phi[i] <- round(eval(parse(text=paste0("fit_model_", i, "$phi"))) , 3)
    }

beta_1 <- rep(0, times = 8)

beta_2 <- rep(0, times = 8)

beta_3 <- rep(0, times = 8)

for(i in 1:4)
    {
    beta_1[i] <- round(eval(parse(text=paste0("fit_model_", i, "$beta")))[2] , 3)
    beta_2[i] <- round(eval(parse(text=paste0("fit_model_", i, "$beta")))[3] , 3)
    beta_3[i] <- round(eval(parse(text=paste0("fit_model_", i, "$beta")))[4] , 4)
    }

nugget <- rep(0, times = 8)

for(i in c(2,4,6,8)) {
    nugget[i] <- round(eval(parse(text=paste0("fit_model_", i, "$nugget"))) , 4)}

kappa <- rep(0.5, times = 8)

for(i in c(1, 2, 5,6)) {
    kappa[i] <- round(eval(parse(text=paste0("fit_model_", i, "$kappa"))) , 3)}


d   <- seq(0, 30, length.out = 200)
cor <- matrix(-99, nrow = length(d), 8)
for (i in 1:8)
    {
    cov <- c(nugget[i], rep(0, times = (length(d) - 1) ) ) + partial_sill[i] *
            matern(d, phi = phi[i], kappa = kappa[i])
    cor[ , i] <- cov / cov[1]
    }

data.frame(
```

```
    d   = rep(d[-1], times = 8),
    cor = c(cor[-1, ]),
    model = rep(paste0("Model_", 1:8), each = (length(d)-1) )
) %>%

    ggplot(aes(x = d, y = cor)) +
        geom_line() +
        geom_point(data = data.frame(d_nugget = d[1], cor_nugget = cor[1]),
                aes(x = d_nugget, y = cor_nugget), inherit.aes = FALSE) +
        ggtitle("Figure 4: Estimated Correlation Function") +
        theme(plot.title = element_text(hjust = 0.5)) +
        xlab("Distance") +
        ylab("Correlation") +
        ylim(c(0, max(cor)))
```

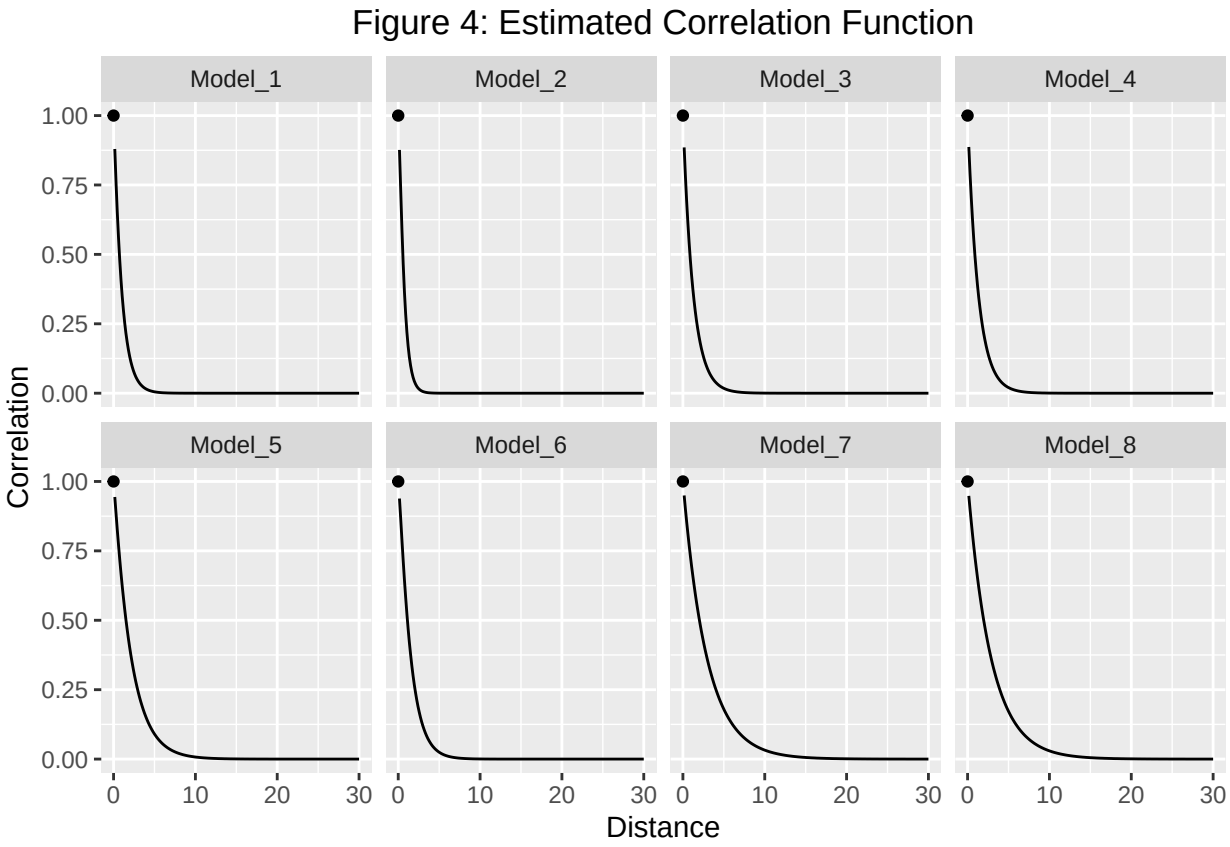


Figure 4 depicts the fitted correlation function for each of the 8 candidate models. Models in the first row have spatial covariates and models in the second row have intercept only. The figure shows that after adding the spatial covariates the correlation drops faster compared to the models without spatial covariates. And in Model 2 (the best model) the decay is the fastest.

Problem 4

Fit a non-spatial regression (e.g., use the *lm* function). Look at the regression coefficient estimates. Is there evidence that the inclusion of the spatial model is changing the covariate estimates? **Write a few**

sentences to interpret the results. Make sure you write out the answers in **clear, English sentences. Communication matters!**

Answer:

```
knitr::kable(
    t(data.frame(
        Model      = 1:8,
        Covariates = rep(c("Spat_Coef", "Inpt_only"), each = 4),
        Covariance = rep(rep(c("Matern", "Exp"), each = 2), times = 2),
        Nugget     = nugget,
        beta_0     = beta_0,
        beta_1     = beta_1,
        beta_2     = beta_2,
        beta_3     = beta_3,
        part_sill     = partial_sill,
        phi     = phi,
        kappa     = kappa
    ) ),
    caption = "Estimated values of the parameters for all models (Here zero represents
    the corresponding parameter is fixed at zero in the corresponding model)"
)
```

Table 3: Estimated values of the parameters for all models (Here zero represents the corresponding parameter is fixed at zero in the corresponding model)

Model	1	2	3	4	5	6	7	8
Covariates	Spat_Coef	Spat_Coef	Spat_Coef	Spat_Coef	Inpt_only	Inpt_only	Inpt_only	Inpt_only
Covariance	Matern	Matern	Exp	Exp	Matern	Matern	Exp	Exp
Nugget	0.0000	0.0064	0.0000	0.0008	0.0000	0.0050	0.0000	0.0002
beta_0	-35.283	-34.843	-35.419	-35.423	5.810	5.823	5.805	5.806
beta_1	-0.102	-0.094	-0.105	-0.105	0.000	0.000	0.000	0.000
beta_2	-0.373	-0.367	-0.375	-0.375	0.000	0.000	0.000	0.000
beta_3	5e-04	5e-04	5e-04	5e-04	0e+00	0e+00	0e+00	0e+00
part_sill	0.308	0.276	0.323	0.326	0.911	0.803	1.009	0.980
phi	0.909	0.496	1.237	1.283	1.974	1.197	2.912	2.841
kappa	0.569	0.901	0.500	0.500	0.557	0.699	0.500	0.500

```
dat %>%
    do(broom::tidy(lm(log_annual_total ~ LAT + LONG + ALT, .)))
```

```
## # A tibble: 4 x 5
##   term          estimate std.error statistic  p.value
##   <chr>            <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept) -28.9      2.17         -13.3  1.81e-34
## 2 LAT          -0.0658   0.0203        -3.24 1.30e- 3
## 3 LONG         -0.310    0.0233       -13.3  1.75e-34
## 4 ALT           0.000535 0.0000525     10.2  4.06e-22
```

Table 3 shows the estimated values of the parameters of all candidate models. Table 2 and the non-spatial regression outputs in the above clearly shows that inclusion of spatial model decrease all the regression

coefficients like the intercept drops by about 6 units form around 29, the coefficients of latitude, longitude becomes around -0.10 and -0.37 from -0.06 and -0.31 after including spatial model. The coefficient of altitude remains similar in both cases.

Problem 5

For the model with the smallest BIC, fit separate models for sites in the northern region vs. the southern region (North of 37.20 degrees latitude (inclusive) vs. south of 37.20 degrees latitude (exclusive)). Based on this fit, do you think the residual covariance is stationary? **Write a few sentences to interpret the results.** Make sure you write out the answers in **clear, English sentences. Communication matters!**

Answer:

```
dat_north <- dat %>%
                subset(LAT >= 37.2)

dat_south <- dat %>%
                subset(LAT < 37.2)

fit_model_2_north <- likfit(
    data         = dat_north$log_annual_total,
    trend        = ~ dat_north$LAT + dat_north$LONG + dat_north$ALT,
    coords       = cbind(dat_north$LAT, dat_north$LONG),
    fix.kappa = FALSE, kappa = 1,
    cov.model    = "matern",
    ini.cov.pars = c(var(dat_north$log_annual_total), .5),
    message = FALSE
)

#summary(fit_model_2_north)

fit_model_2_south <- likfit(
    data         = dat_south$log_annual_total,
    trend        = ~ dat_south$LAT + dat_south$LONG + dat_south$ALT,
    coords       = cbind(dat_south$LAT, dat_south$LONG),
    fix.kappa = FALSE, kappa = 1,
    cov.model    = "matern",
    ini.cov.pars = c(var(dat_south$log_annual_total), .5),
    message = FALSE
)

nugget_new <- round(c(fit_model_2_north$nugget, fit_model_2_south$nugget), 4)
partial_sill_new <- round( c(fit_model_2_north$sigmasq, fit_model_2_south$sigmasq),3 )
phi_new <- round( c(fit_model_2_north$phi, fit_model_2_south$phi),3 )
kappa_new <- round(c(fit_model_2_north$kappa, fit_model_2_south$kappa), 3)

knitr::kable(
    data.frame(
        Model      = c("Model_2", "Model_2_North", "Model_2_South" ),
        Nugget     = c(nugget[2], nugget_new),
        Partial_sill  = c(partial_sill[2],  partial_sill_new),
        Phi        = c(phi[2], phi_new),
        Kappa      = c(kappa[2], kappa_new)
```

```
    ),
    caption = "Estimated values of the covariance parameters"
)
```

Table 4: Estimated values of the covariance parameters

Model	Nugget	Partial_sill	Phi	Kappa
Model_2	0.0064	0.276	0.496	0.901
Model_2_North	0.0073	0.235	0.482	1.133
Model_2_South	0.0078	0.248	0.324	1.035

```
d   <- seq(0, 5, length.out = 200)
cov <- matrix(-99, nrow = length(d), 2)
for (i in 1:2)
    {
    cov[ , i] <- c(nugget_new[i], rep(0, times = (length(d) - 1) ) ) +
        partial_sill_new[i]*matern(d, phi = phi_new[i], kappa = kappa_new[i])
    }

cov = cbind( c(nugget[2], rep(0, times = (length(d) - 1) ) ) +
                partial_sill[2]*matern(d, phi = phi[2], kappa = kappa[2]), cov )

data.frame(
    d   = rep(d[-1], times = 3),
    cov = c(cov[-1, ]),
    Model = rep(c("Model_2", "Model_2_North", "Model_2_South" ), each = (length(d)-1) )
    ) %>%
    ggplot(aes(x = d, y = cov, col = Model)) +
        geom_line() +
        geom_point(data = data.frame(d_nugget = rep(d[1], 3), cov_nugget = cov[1, ],
                           Model = c("Model_2", "Model_2_North", "Model_2_South" ) ),
                aes(x = d_nugget, y = cov_nugget, col = Model), inherit.aes = FALSE) +
        ylim(c(0, max(cov))) +
        xlab("Distance") +
        ylab("Covariance") +
        theme(plot.title = element_text(hjust = 0.5))
```

Figure 5: Estimated Covariance Function

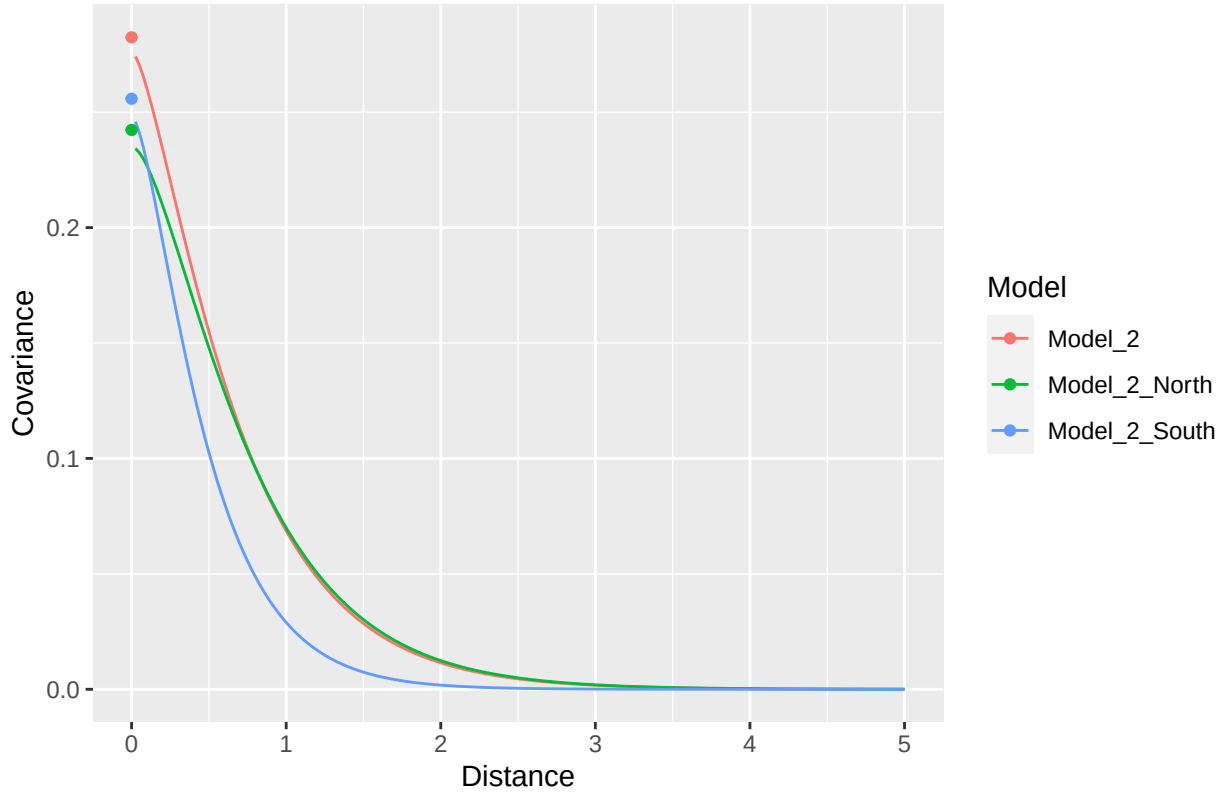


Table 4 shows that the nugget and partial sill are similar in both north as well as in south part, and these values are also close the corresponding value in the full model. However, in the south part the phi and kappa takes slightly lower values compared to the corresponding estimation in the north. The covariance function plot in Figure 5 also depicts that the patterns are close in both part. Therefore, we can interpret that the residual covariance are almost stationary.