Assignment\_2\_STA5001\_Michael\_Le

2024-04-08

#Part a. Produce a quick numeric summary of the data. #What are (round the answer with 3 decimal places)

library(geoR)

## Warning: package 'geoR' was built under R version 4.3.3

## --------------------------------------------------------------  
## Analysis of Geostatistical Data  
## For an Introduction to geoR go to http://www.leg.ufpr.br/geoR  
## geoR version 1.9-4 (built on 2024-02-14) is now loaded  
## --------------------------------------------------------------

sic.100

## $coords  
## V2 V3  
## 13 29.52739 80.71854  
## 14 33.77939 99.52954  
## 22 46.80639 102.58454  
## 23 48.71439 121.45354  
## 24 49.31639 113.65554  
## 29 53.21039 79.09954  
## 30 54.51039 106.87954  
## 35 60.31039 132.35054  
## 36 60.79839 75.62054  
## 37 60.81839 76.73254  
## 52 76.45039 97.59954  
## 55 77.70339 177.65154  
## 66 85.25839 179.76254  
## 71 86.30039 93.00554  
## 73 86.20739 30.73054  
## 84 94.39639 129.59654  
## 95 101.28339 68.35054  
## 102 106.13639 19.36854  
## 105 107.31339 131.66954  
## 126 123.54739 73.69754  
## 130 124.59039 110.38554  
## 136 127.39639 182.64354  
## 138 127.47739 193.76254  
## 159 135.85839 84.73554  
## 168 140.10639 169.22154  
## 172 141.56539 159.20654  
## 178 143.23239 193.66954  
## 181 144.67939 181.43154  
## 185 145.27339 143.62254  
## 188 146.01239 139.17254  
## 192 148.43039 179.19254  
## 198 150.38039 85.78154  
## 202 152.77239 120.24554  
## 203 152.80439 131.36454  
## 207 155.96239 183.61754  
## 208 156.42339 54.63054  
## 218 163.46639 173.59754  
## 224 170.24439 110.21254  
## 226 171.78039 60.17554  
## 235 176.31239 128.00754  
## 245 184.01239 79.09454  
## 246 183.96139 101.33354  
## 247 186.16839 130.24954  
## 254 191.40439 150.28154  
## 261 194.35439 168.08454  
## 263 195.72839 198.11254  
## 274 200.35939 172.56054  
## 275 201.25339 145.87754  
## 276 201.20639 154.77354  
## 277 202.29139 93.62054  
## 281 203.14039 78.05754  
## 283 204.13539 36.92054  
## 287 203.86439 217.05654  
## 292 207.62239 213.74454  
## 293 209.26739 76.98254  
## 300 211.37039 104.79654  
## 302 212.15639 207.10354  
## 305 214.78339 157.08454  
## 314 215.97839 197.12454  
## 317 217.71839 72.59554  
## 320 219.37639 57.04154  
## 322 219.44239 141.55354  
## 335 224.21539 198.30654  
## 336 226.55339 28.19254  
## 341 225.85239 183.86654  
## 342 225.84139 184.97854  
## 344 229.14839 78.25754  
## 357 233.57539 20.47754  
## 362 233.16239 202.84454  
## 368 235.70839 176.18354  
## 369 235.44639 199.53354  
## 372 236.24639 195.09454  
## 373 238.42839 136.18154  
## 377 241.21639 92.84454  
## 378 240.77439 192.92254  
## 381 241.91839 160.68754  
## 384 243.30239 170.71354  
## 392 248.28639 77.36354  
## 399 251.84939 151.92054  
## 400 251.95039 198.62854  
## 401 254.68439 111.92554  
## 406 255.53039 158.64454  
## 408 257.97139 95.29154  
## 409 256.87539 169.78554  
## 421 264.89139 186.59054  
## 423 266.71039 121.00454  
## 424 266.45739 136.57054  
## 425 268.17739 77.65554  
## 436 275.17839 71.10054  
## 442 277.06039 94.48954  
## 449 283.14239 95.71354  
## 450 281.62639 175.76254  
## 451 284.53639 62.37454  
## 455 290.88139 90.30354  
## 456 292.86039 68.09854  
## 458 294.99739 112.63254  
## 460 298.08639 110.47354  
## 466 312.06739 66.29654  
## 468 315.29239 59.69954  
## 471 320.91139 49.82554  
##   
## $data  
## 13 14 22 23 24 29 30 35 36 37 52 55 66 71 73 84 95 102 105 126   
## 151 255 79 191 194 334 107 296 394 394 324 105 135 585 114 334 131 78 398 141   
## 130 136 138 159 168 172 178 181 185 188 192 198 202 203 207 208 218 224 226 235   
## 192 151 107 145 334 327 213 331 400 327 380 94 185 239 330 30 254 53 78 71   
## 245 246 247 254 261 263 274 275 276 277 281 283 287 292 293 300 302 305 314 317   
## 62 71 59 60 124 153 75 137 86 129 345 441 184 121 346 270 100 45 107 359   
## 320 322 335 336 341 342 344 357 362 368 369 372 373 377 378 381 384 392 399 400   
## 278 72 131 90 141 131 452 16 136 130 118 109 145 254 140 152 60 283 184 127   
## 401 406 408 409 421 423 424 425 436 442 449 450 451 455 456 458 460 466 468 471   
## 220 178 218 137 144 230 282 129 65 190 170 156 131 10 99 92 67 18 20 55   
##   
## $covariate  
## altitude  
## 1 682  
## 2 813  
## 3 436  
## 4 833  
## 5 579  
## 6 525  
## 7 433  
## 8 1169  
## 9 548  
## 10 784  
## 11 785  
## 12 598  
## 13 434  
## 14 753  
## 15 466  
## 16 435  
## 17 1097  
## 18 2054  
## 19 698  
## 20 1285  
## 21 539  
## 22 334  
## 23 248  
## 24 1001  
## 25 799  
## 26 468  
## 27 300  
## 28 413  
## 29 654  
## 30 824  
## 31 596  
## 32 1217  
## 33 946  
## 34 1223  
## 35 573  
## 36 725  
## 37 388  
## 38 690  
## 39 1576  
## 40 1770  
## 41 1379  
## 42 1220  
## 43 763  
## 44 436  
## 45 565  
## 46 407  
## 47 589  
## 48 750  
## 49 609  
## 50 1091  
## 51 1312  
## 52 806  
## 53 756  
## 54 491  
## 55 996  
## 56 1079  
## 57 423  
## 58 406  
## 59 463  
## 60 943  
## 61 1094  
## 62 904  
## 63 466  
## 64 828  
## 65 546  
## 66 532  
## 67 936  
## 68 770  
## 69 624  
## 70 765  
## 71 454  
## 72 442  
## 73 657  
## 74 1606  
## 75 433  
## 76 647  
## 77 843  
## 78 1916  
## 79 1012  
## 80 474  
## 81 1085  
## 82 1328  
## 83 1869  
## 84 956  
## 85 459  
## 86 1275  
## 87 465  
## 88 1529  
## 89 2040  
## 90 1076  
## 91 1202  
## 92 398  
## 93 1667  
## 94 1433  
## 95 1989  
## 96 2418  
## 97 1720  
## 98 2332  
## 99 1058  
## 100 731  
##   
## attr(,"class")  
## [1] "geodata"

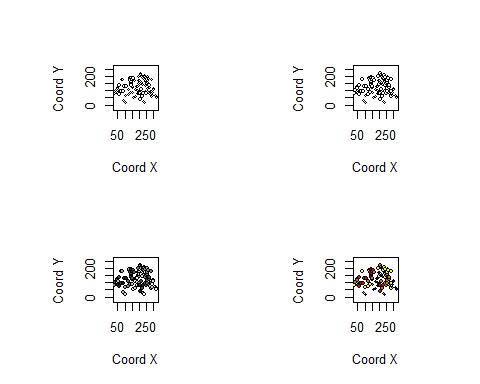
summary(sic.100)

## Number of data points: 100   
##   
## Coordinates summary  
## V2 V3  
## min 29.52739 19.36854  
## max 320.91139 217.05654  
##   
## Distance summary  
## min max   
## 1.112054 293.017086   
##   
## Data summary  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 10.00 97.75 141.00 180.15 254.25 585.00   
##   
## Covariates summary  
## altitude   
## Min. : 248.0   
## 1st Qu.: 516.5   
## Median : 764.0   
## Mean : 891.5   
## 3rd Qu.:1094.8   
## Max. :2418.0

#Minimum distance between data points = 1.112  
  
#min\_dis   
#Maximum distance between data points = 293.017

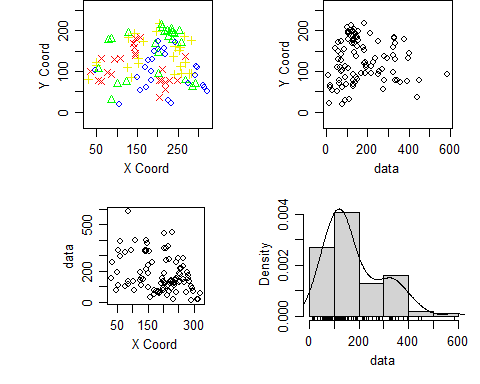
#Part b. #Visualize the locations (using 4 plots) of the data in the x and y planes.

par(mfrow=c(2,2))  
points(sic.100, xlab = "Coord X", ylab = "Coord Y",cex.max=0.7)  
points(sic.100, xlab = "Coord X", ylab = "Coord Y", pt.divide = "rank.prop",cex.max= 0.7)  
points(sic.100, xlab = "Coord X", ylab = "Coord Y", cex.max = 0.7, col = gray(seq(1, 0.1, l=100)), pt.divide = "equal")  
points(sic.100, pt.divide = "quintile", xlab = "Coord X", ylab = "Coord Y",cex.max=0.7)



#Part c.

plot(sic.100)



#What is the approximate y coordinate of a point with the largest data value?   
#By observation  
  
#(i) 0  
#0  
  
#(ii) 100  
#207.10354  
  
#(iii) 400  
#143.62254  
  
#(iv) 600  
#0  
  
#NOTE: They are rough estimates

# Reading data and coords from observation

sic.100$data

## 13 14 22 23 24 29 30 35 36 37 52 55 66 71 73 84 95 102 105 126   
## 151 255 79 191 194 334 107 296 394 394 324 105 135 585 114 334 131 78 398 141   
## 130 136 138 159 168 172 178 181 185 188 192 198 202 203 207 208 218 224 226 235   
## 192 151 107 145 334 327 213 331 400 327 380 94 185 239 330 30 254 53 78 71   
## 245 246 247 254 261 263 274 275 276 277 281 283 287 292 293 300 302 305 314 317   
## 62 71 59 60 124 153 75 137 86 129 345 441 184 121 346 270 100 45 107 359   
## 320 322 335 336 341 342 344 357 362 368 369 372 373 377 378 381 384 392 399 400   
## 278 72 131 90 141 131 452 16 136 130 118 109 145 254 140 152 60 283 184 127   
## 401 406 408 409 421 423 424 425 436 442 449 450 451 455 456 458 460 466 468 471   
## 220 178 218 137 144 230 282 129 65 190 170 156 131 10 99 92 67 18 20 55

sic.100$coords

## V2 V3  
## 13 29.52739 80.71854  
## 14 33.77939 99.52954  
## 22 46.80639 102.58454  
## 23 48.71439 121.45354  
## 24 49.31639 113.65554  
## 29 53.21039 79.09954  
## 30 54.51039 106.87954  
## 35 60.31039 132.35054  
## 36 60.79839 75.62054  
## 37 60.81839 76.73254  
## 52 76.45039 97.59954  
## 55 77.70339 177.65154  
## 66 85.25839 179.76254  
## 71 86.30039 93.00554  
## 73 86.20739 30.73054  
## 84 94.39639 129.59654  
## 95 101.28339 68.35054  
## 102 106.13639 19.36854  
## 105 107.31339 131.66954  
## 126 123.54739 73.69754  
## 130 124.59039 110.38554  
## 136 127.39639 182.64354  
## 138 127.47739 193.76254  
## 159 135.85839 84.73554  
## 168 140.10639 169.22154  
## 172 141.56539 159.20654  
## 178 143.23239 193.66954  
## 181 144.67939 181.43154  
## 185 145.27339 143.62254  
## 188 146.01239 139.17254  
## 192 148.43039 179.19254  
## 198 150.38039 85.78154  
## 202 152.77239 120.24554  
## 203 152.80439 131.36454  
## 207 155.96239 183.61754  
## 208 156.42339 54.63054  
## 218 163.46639 173.59754  
## 224 170.24439 110.21254  
## 226 171.78039 60.17554  
## 235 176.31239 128.00754  
## 245 184.01239 79.09454  
## 246 183.96139 101.33354  
## 247 186.16839 130.24954  
## 254 191.40439 150.28154  
## 261 194.35439 168.08454  
## 263 195.72839 198.11254  
## 274 200.35939 172.56054  
## 275 201.25339 145.87754  
## 276 201.20639 154.77354  
## 277 202.29139 93.62054  
## 281 203.14039 78.05754  
## 283 204.13539 36.92054  
## 287 203.86439 217.05654  
## 292 207.62239 213.74454  
## 293 209.26739 76.98254  
## 300 211.37039 104.79654  
## 302 212.15639 207.10354  
## 305 214.78339 157.08454  
## 314 215.97839 197.12454  
## 317 217.71839 72.59554  
## 320 219.37639 57.04154  
## 322 219.44239 141.55354  
## 335 224.21539 198.30654  
## 336 226.55339 28.19254  
## 341 225.85239 183.86654  
## 342 225.84139 184.97854  
## 344 229.14839 78.25754  
## 357 233.57539 20.47754  
## 362 233.16239 202.84454  
## 368 235.70839 176.18354  
## 369 235.44639 199.53354  
## 372 236.24639 195.09454  
## 373 238.42839 136.18154  
## 377 241.21639 92.84454  
## 378 240.77439 192.92254  
## 381 241.91839 160.68754  
## 384 243.30239 170.71354  
## 392 248.28639 77.36354  
## 399 251.84939 151.92054  
## 400 251.95039 198.62854  
## 401 254.68439 111.92554  
## 406 255.53039 158.64454  
## 408 257.97139 95.29154  
## 409 256.87539 169.78554  
## 421 264.89139 186.59054  
## 423 266.71039 121.00454  
## 424 266.45739 136.57054  
## 425 268.17739 77.65554  
## 436 275.17839 71.10054  
## 442 277.06039 94.48954  
## 449 283.14239 95.71354  
## 450 281.62639 175.76254  
## 451 284.53639 62.37454  
## 455 290.88139 90.30354  
## 456 292.86039 68.09854  
## 458 294.99739 112.63254  
## 460 298.08639 110.47354  
## 466 312.06739 66.29654  
## 468 315.29239 59.69954  
## 471 320.91139 49.82554

#Part d.Plot a sample variogram on the interval [0,300] using 11 bins.(0.5 mark)

library(geoR)  
data(SIC)  
#NOTE: Ensure to change max.dist 1 to 300.  
cloud1<-variog(sic.100,option="cloud",max.dist=300)

## variog: computing omnidirectional variogram

cloud2<-variog(sic.100,option="cloud",estimator.type= "modulus",max.dist=300)

## variog: computing omnidirectional variogram

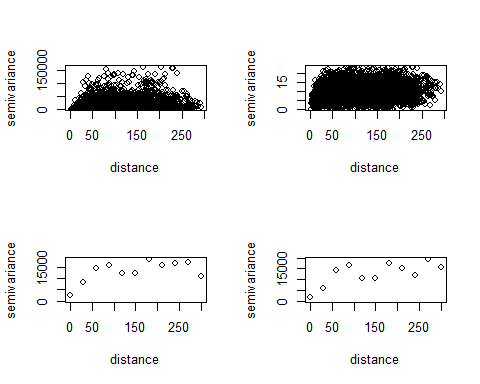
bin1<-variog(sic.100,uvec=seq(0,300,l=11))

## variog: computing omnidirectional variogram

bin2 <-variog(sic.100,uvec=seq(0,300,l=11),estimator.type="modulus")

## variog: computing omnidirectional variogram

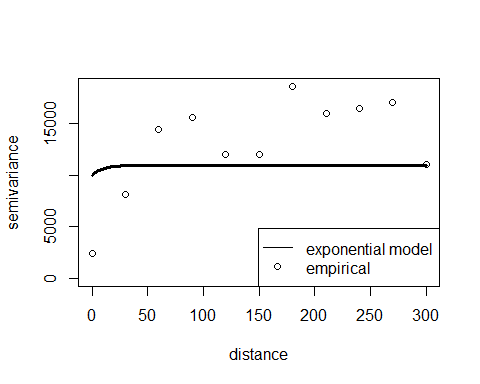
par(mfrow = c(2,2))  
plot(cloud1)  
plot(cloud2)  
plot(bin1)  
plot(bin2)



#Part e. # Fit the exponential variogram to the sample variogram from # (d) by using ordinary least squares and maximum likelihood methods. # Use the initial values (1000,10) and nugget = 10000.

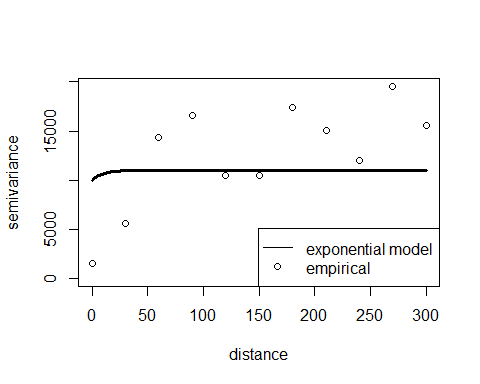
#For bin1

plot(bin1)  
lines.variomodel(cov.model = "exp", cov.pars = c(1000,10),nugget = 10000, max.dist = 300, lwd = 3)  
legend("bottomright", c("exponential model", "empirical"),lty=c(1,0), lwd = c(1,1), pch=c(NA,1))



#Simarly for bin2

plot(bin2)  
lines.variomodel(cov.model = "exp", cov.pars = c(1000,10),nugget = 10000, max.dist = 300, lwd = 3)  
legend("bottomright", c("exponential model", "empirical"),lty=c(1,0), lwd = c(1,1), pch=c(NA,1))



#Part f. #Plot the fitted variograms against the sample variogram. Compare the fitted ordinary least squares and maximum likelihood curves. (0.5 mark)

ml.n <- likfit(sic.100, ini = c(1000,10), nug = 1000,cov.model = "exponential")

## kappa not used for the exponential correlation function  
## ---------------------------------------------------------------  
## likfit: likelihood maximisation using the function optim.  
## likfit: Use control() to pass additional  
## arguments for the maximisation function.  
## For further details see documentation for optim.  
## likfit: It is highly advisable to run this function several  
## times with different initial values for the parameters.  
## likfit: WARNING: This step can be time demanding!  
## ---------------------------------------------------------------  
## likfit: end of numerical maximisation.

ml1.n <- likfit(sic.100, ini = c(1000,10), nug = 1000,cov.model = "wave")

## kappa not used for the wave correlation function  
## ---------------------------------------------------------------  
## likfit: likelihood maximisation using the function optim.  
## likfit: Use control() to pass additional  
## arguments for the maximisation function.  
## For further details see documentation for optim.  
## likfit: It is highly advisable to run this function several  
## times with different initial values for the parameters.  
## likfit: WARNING: This step can be time demanding!  
## ---------------------------------------------------------------  
## likfit: end of numerical maximisation.

bin1 <- variog(sic.100, uvec=seq(0,300,l=11))

## variog: computing omnidirectional variogram

ols.n <- variofit(bin1, ini = c(1000,10),nugget=1000,weights="equal", cov.model = "exponential")

## variofit: covariance model used is exponential   
## variofit: weights used: equal   
## variofit: minimisation function used: optim

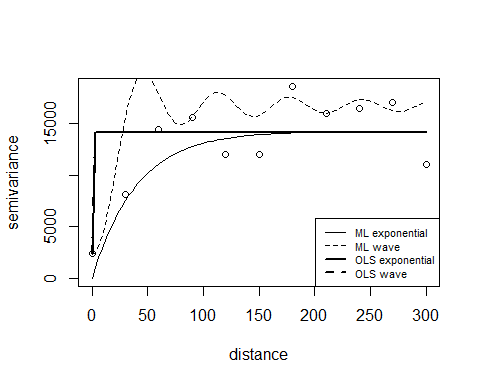
## Warning in variofit(bin1, ini = c(1000, 10), nugget = 1000, weights = "equal",  
## : unreasonable initial value for sigmasq + nugget (too low)

ols1.n <- variofit(bin1,ini = c(1000,10),nugget=1000,weights="equal", cov.model = "wave")

## variofit: covariance model used is wave   
## variofit: weights used: equal   
## variofit: minimisation function used: optim

## Warning in variofit(bin1, ini = c(1000, 10), nugget = 1000, weights = "equal",  
## : unreasonable initial value for sigmasq + nugget (too low)

plot(bin1)  
lines(ml.n, max.dist = 300)  
lines(ml1.n,lty = 2, max.dist = 300)  
lines(ols.n, lwd = 2, max.dist = 300)  
lines(ols1.n, lty = 2, lwd = 2, max.dist = 300)  
legend("bottomright", legend=c("ML exponential","ML wave","OLS exponential","OLS wave"),lty=c(1,2,1,2),lwd=c(1,1,2,2), cex=0.7)



ols.n

## variofit: model parameters estimated by OLS (ordinary least squares):  
## covariance model is: exponential  
## parameter estimates:  
## tausq sigmasq phi   
## 2432.639 11735.453 0.000   
## Practical Range with cor=0.05 for asymptotic range: 0.0001159668  
##   
## variofit: minimised sum of squares = 94076369

summary(ml.n)

## Summary of the parameter estimation  
## -----------------------------------  
## Estimation method: maximum likelihood   
##   
## Parameters of the mean component (trend):  
## beta   
## 154.8629   
##   
## Parameters of the spatial component:  
## correlation function: exponential  
## (estimated) variance parameter sigmasq (partial sill) = 14282  
## (estimated) cor. fct. parameter phi (range parameter) = 39.96  
## anisotropy parameters:  
## (fixed) anisotropy angle = 0 ( 0 degrees )  
## (fixed) anisotropy ratio = 1  
##   
## Parameter of the error component:  
## (estimated) nugget = 0  
##   
## Transformation parameter:  
## (fixed) Box-Cox parameter = 1 (no transformation)  
##   
## Practical Range with cor=0.05 for asymptotic range: 119.7063  
##   
## Maximised Likelihood:  
## log.L n.params AIC BIC   
## "-576.2" "4" "1160" "1171"   
##   
## non spatial model:  
## log.L n.params AIC BIC   
## "-617.3" "2" "1239" "1244"   
##   
## Call:  
## likfit(geodata = sic.100, ini.cov.pars = c(1000, 10), nugget = 1000,   
## cov.model = "exponential")

summary(ols.n)

## $pmethod  
## [1] "OLS (ordinary least squares)"  
##   
## $cov.model  
## [1] "exponential"  
##   
## $spatial.component  
## sigmasq phi   
## 11735.45 0.00   
##   
## $spatial.component.extra  
## kappa   
## 0.5   
##   
## $nugget.component  
## tausq   
## 2432.639   
##   
## $fix.nugget  
## [1] FALSE  
##   
## $fix.kappa  
## [1] TRUE  
##   
## $practicalRange  
## [1] 0.0001159668  
##   
## $sum.of.squares  
## value   
## 94076369   
##   
## $estimated.pars  
## tausq sigmasq phi   
## 2432.639 11735.453 0.000   
##   
## $weights  
## [1] "equal"  
##   
## $call  
## variofit(vario = bin1, ini.cov.pars = c(1000, 10), cov.model = "exponential",   
## nugget = 1000, weights = "equal")  
##   
## attr(,"class")  
## [1] "summary.variomodel"

summary(ml.n)

## Summary of the parameter estimation  
## -----------------------------------  
## Estimation method: maximum likelihood   
##   
## Parameters of the mean component (trend):  
## beta   
## 154.8629   
##   
## Parameters of the spatial component:  
## correlation function: exponential  
## (estimated) variance parameter sigmasq (partial sill) = 14282  
## (estimated) cor. fct. parameter phi (range parameter) = 39.96  
## anisotropy parameters:  
## (fixed) anisotropy angle = 0 ( 0 degrees )  
## (fixed) anisotropy ratio = 1  
##   
## Parameter of the error component:  
## (estimated) nugget = 0  
##   
## Transformation parameter:  
## (fixed) Box-Cox parameter = 1 (no transformation)  
##   
## Practical Range with cor=0.05 for asymptotic range: 119.7063  
##   
## Maximised Likelihood:  
## log.L n.params AIC BIC   
## "-576.2" "4" "1160" "1171"   
##   
## non spatial model:  
## log.L n.params AIC BIC   
## "-617.3" "2" "1239" "1244"   
##   
## Call:  
## likfit(geodata = sic.100, ini.cov.pars = c(1000, 10), nugget = 1000,   
## cov.model = "exponential")

summary(ols.n)

## $pmethod  
## [1] "OLS (ordinary least squares)"  
##   
## $cov.model  
## [1] "exponential"  
##   
## $spatial.component  
## sigmasq phi   
## 11735.45 0.00   
##   
## $spatial.component.extra  
## kappa   
## 0.5   
##   
## $nugget.component  
## tausq   
## 2432.639   
##   
## $fix.nugget  
## [1] FALSE  
##   
## $fix.kappa  
## [1] TRUE  
##   
## $practicalRange  
## [1] 0.0001159668  
##   
## $sum.of.squares  
## value   
## 94076369   
##   
## $estimated.pars  
## tausq sigmasq phi   
## 2432.639 11735.453 0.000   
##   
## $weights  
## [1] "equal"  
##   
## $call  
## variofit(vario = bin1, ini.cov.pars = c(1000, 10), cov.model = "exponential",   
## nugget = 1000, weights = "equal")  
##   
## attr(,"class")  
## [1] "summary.variomodel"

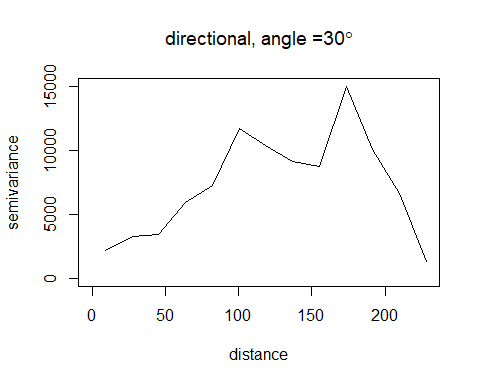
#Main question: #Which of these curves better fits points of the empirical variogram? # The Maximum Likelihoog wave, because,…………

#Part g. Plot the empirical directional variogram for 30 degrees with a tolerance angle 20 degrees. #Use the line plot. (0.5 mark)

vario30 <- variog(sic.100, max.dist = 300, direction=pi/6, tolerance = pi/9)

## variog: computing variogram for direction = 30 degrees (0.524 radians)  
## tolerance angle = 20 degrees (0.349 radians)

plot(vario30,type="l")  
 title(main=expression(paste("directional, angle =",30\*degree)))

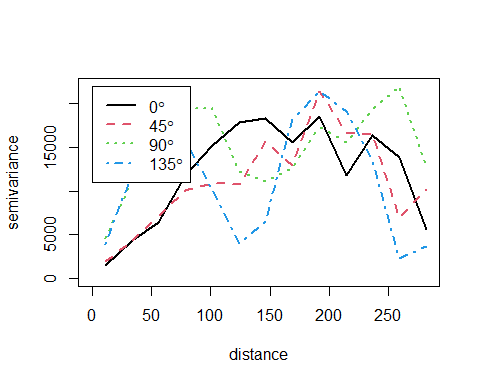


#Part h.Plot a multidirectional empirical variogram (computed for 0, 45, 90, and 135 degrees)

vario.4 <- variog4(sic.100, max.dist = 300)

## variog: computing variogram for direction = 0 degrees (0 radians)  
## tolerance angle = 22.5 degrees (0.393 radians)  
## variog: computing variogram for direction = 45 degrees (0.785 radians)  
## tolerance angle = 22.5 degrees (0.393 radians)  
## variog: computing variogram for direction = 90 degrees (1.571 radians)  
## tolerance angle = 22.5 degrees (0.393 radians)  
## variog: computing variogram for direction = 135 degrees (2.356 radians)  
## tolerance angle = 22.5 degrees (0.393 radians)  
## variog: computing omnidirectional variogram

plot(vario.4, lwd=2)



#Part i. Create a data frame from sic.100 using the command #Create a SpatialPointsDataFrame from df

library(sp)

## Warning: package 'sp' was built under R version 4.3.3

llCRS <- CRS("+proj=longlat +ellps=WGS84")   
df <- data.frame(sic.100$coords, data = sic.100$data)  
coords <- coordinates(df[,c("V2","V3")])  
row.names(coords) <- 1:nrow(coords)  
row.names(df) <- 1:nrow(df)  
df\_spdf <- SpatialPointsDataFrame(coords,df,proj4string = llCRS, match.ID = TRUE)  
summary(df\_spdf)

## Object of class SpatialPointsDataFrame  
## Coordinates:  
## min max  
## V2 29.52739 320.9114  
## V3 19.36854 217.0565  
## Is projected: FALSE   
## proj4string : [+proj=longlat +ellps=WGS84 +no\_defs]  
## Number of points: 100  
## Data attributes:  
## V2 V3 data   
## Min. : 29.53 Min. : 19.37 Min. : 10.00   
## 1st Qu.:141.20 1st Qu.: 79.10 1st Qu.: 97.75   
## Median :202.72 Median :121.23 Median :141.00   
## Mean :187.78 Mean :124.92 Mean :180.15   
## 3rd Qu.:241.39 3rd Qu.:172.82 3rd Qu.:254.25   
## Max. :320.91 Max. :217.06 Max. :585.00

#Part j. # Assume an anisotropic exponential model. Apply the likfit function with # coords = coordinates(df) # data = df$data

df <- data.frame(sic.100$coords, data = sic.100$data)  
coords <- coordinates(df[,c("V2","V3")])  
row.names(coords) <- 1:nrow(coords)  
row.names(df) <- 1:nrow(df)  
ml.df\_dat <- likfit(geodata = sic.100,coords,data = df$data , ini = c(1000,10), nug = 10000,cov.model = "exponential")

## kappa not used for the exponential correlation function  
## ---------------------------------------------------------------  
## likfit: likelihood maximisation using the function optim.  
## likfit: Use control() to pass additional  
## arguments for the maximisation function.  
## For further details see documentation for optim.  
## likfit: It is highly advisable to run this function several  
## times with different initial values for the parameters.  
## likfit: WARNING: This step can be time demanding!  
## ---------------------------------------------------------------  
## likfit: end of numerical maximisation.

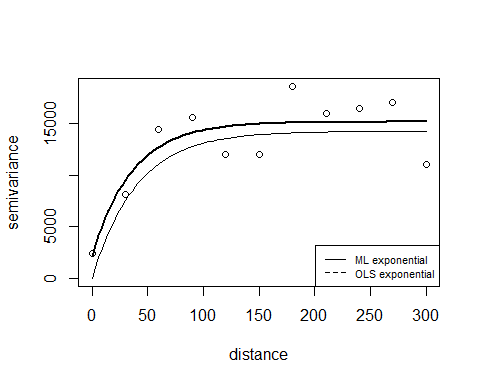
bin\_df\_dat <- variog(geodata =sic.100,coords ,data = df$data, uvec=seq(0,300,l=11))

## variog: computing omnidirectional variogram

ols.df\_dat <- variofit(bin\_df\_dat , ini = c(1000,10),nugget=10000,weights="equal", cov.model = "exponential")

## variofit: covariance model used is exponential   
## variofit: weights used: equal   
## variofit: minimisation function used: optim

plot(bin\_df\_dat)  
lines(ml.df\_dat , max.dist = 300)  
lines(ols.df\_dat, lwd = 2, max.dist = 300)  
legend("bottomright", legend=c("ML exponential","OLS exponential"),lty=c(1,2),lwd=c(1,1), cex=0.7)



#Question: Is this model isotropic? # Given that the initial values is not too far off, we can reasonably say # this model is isotropic Given that each of the lines for the Maximum likelihood and Ordinarly Least Squares.

#Part k. Produce a 2D variogram plot for the model. # data ~ 1 with cutoff = 200 and width = 10

library(sp)  
library(devtools)

## Warning: package 'devtools' was built under R version 4.3.3

## Loading required package: usethis

## Warning: package 'usethis' was built under R version 4.3.3

library(gstat)

## Warning: package 'gstat' was built under R version 4.3.3

plot(variogram(data~1,df\_spdf,map=TRUE,cutoff=200,width=10))

