Spatial autocorrelation

Eddie Wu

2023-09-19

Introduction

This .Rmd file is to test the spatial autocorrelation in our black carp data file, and try to find the best distance that reduces the spatial autocorrelation when doing subsampling. In this file, we:

1. Show the distribution of our data points on a map.
2. Use moran's I test to test the global spatial autocorrelation in our dataset.
3. Use the correlog plots to examine how spatial autocorrelation changes with distances.
4. Use the variogram to further test point 2.
5. Subsample at 250 km and 550 km, and check the local moran's I after sub-sampling to see if spatial autocorrelation has been reduced.
6. Conduct latitudinal stratification analysis.
7. Look at the Chinese dataset - 17 data points.

```
library(gstat)
library(ggplot2)
library(dplyr)
library(spdep)
library(sp)
library(nlme)
library(ape)
library(MuMIn)
library(raster)
library(ncf)
library(knitr)
library(rnaturalearth)
library(sf)
```

```
## Import location data
location <- read.csv("location_no_temps.csv")
location <- unique(location)

location.new <- location %>% # remove duplicating locations
  filter(Location != "South Ukraine") # remove South Ukraine

# Clean the data
carp <- read.csv("eddie_carp_new.csv")
carp.r <- carp %>%
  filter(!row_number() == 5) %>%
  filter(sex != "male")
```

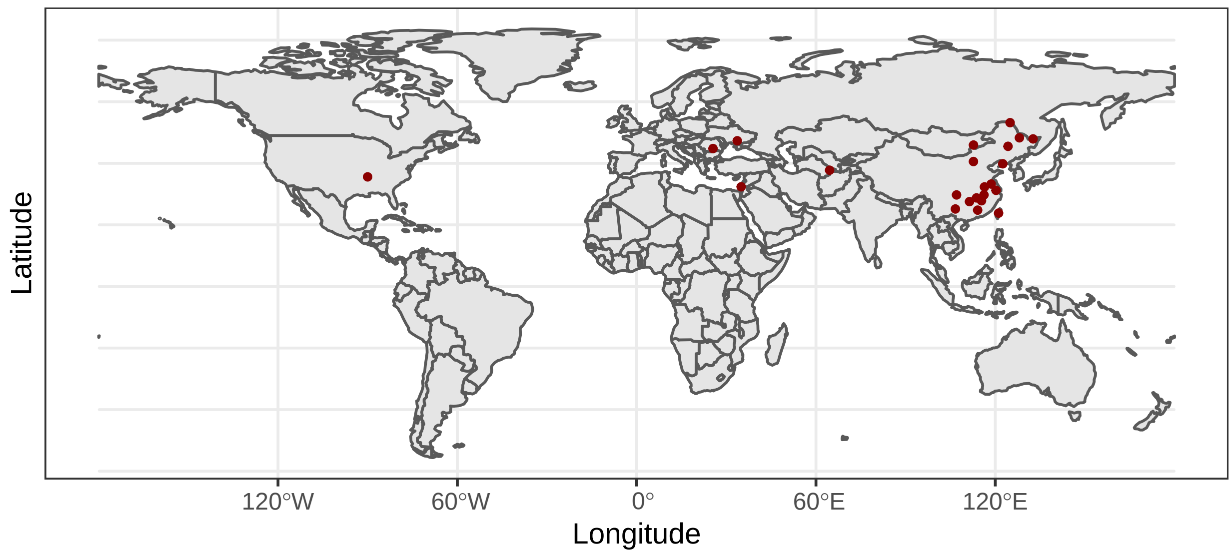
```
carp.new <- carp.r %>%
  filter(!row_number() == 20) # remove South Ukraine


## Download one file to get the spatial points
# (this defines what projection to use when converting to spatial objects)
tmin.1979 <- brick("cpc/tmin.1979.nc", varname = "tmin")
```

```
## Loading required namespace: ncdf4
```

```
tmin.1979<-rotate(tmin.1979)


## Define a function to plot the correlogram for local moran's I

plot_correlog <- function(model){
  # Define margins
  par(mfrow=c(2,1),mar=c(4,4,2,2))

  # Use the coorelog function to develop the relationship
  test <- correlog(coords$long, coords$lat, model$residuals,
                   increment=50, resamp=500, latlon=T)

  # Plot with the entire distance range
  plot(test, main=paste(model$call))
  abline(h=0)

  # Reduce the distance range to 2500 km
  plot(test, main="", xlim=c(0,2500))
  abline(h=0)
}
```

Data distribution

```
## Get the world data
world <- ne_countries(scale = "small", returnclass = "sf")

## Plots (global plot)
world %>%
  filter(admin != "Antarctica") %>%
  # remove antarctica
  ggplot()+
  geom_sf()+
  geom_point(aes(x=Longitude, y=Latitude), data = location,
             size = 1, color = "darkred")+
  theme_bw()+
  xlab("Longitude") + ylab("Latitude")
```



Section 1: Spatial autocorrelation examination for entire dataset

Model definition

We first need to define all 9 models (3 models for each temperature).

```
# Define the models for AnnualTemp
annual.lin <- lm(log(AAM)~AnnualTemp, data = carp.r)
annual.add <- lm(log(AAM)~AnnualTemp+condition, data = carp.r)
annual.int <- lm(log(AAM)~AnnualTemp*condition, data = carp.r)

# Define the models for ColdTemp
cold.lin <- lm(log(AAM)~ColdTemp, data = carp.r)
cold.add <- lm(log(AAM)~ColdTemp+condition, data = carp.r)
cold.int <- lm(log(AAM)~ColdTemp*condition, data = carp.r)

# Define the model for WarmTemp
warm.lin <- lm(log(AAM)~WarmTemp, data = carp.r)
warm.add <- lm(log(AAM)~WarmTemp+condition, data = carp.r)
warm.int <- lm(log(AAM)~WarmTemp*condition, data = carp.r)
```

Global Moran.I

```
## Make spatial dataframe
coords <- data.frame("long"=location[,3],"lat"=location[,2])
df <- data.frame(a = 1:nrow(location[3]))
```

```r
spatial.data <- SpatialPointsDataFrame(coords,df,proj4string = tmin.1979@crs)

# Get a distance matrix from all points
dists <- spDists(spatial.data, longlat = TRUE)


## Global Moran.I
# Annual Temp
Moran.annual.lin <- Moran.I(annual.lin$residuals, dists)
Moran.annual.add <- Moran.I(annual.add$residuals, dists)
Moran.annual.int <- Moran.I(annual.int$residuals, dists)

global.moran.annual <- data.frame(
  Model = c("simple linear", "linear additive", "interaction"),
  Observed = c(Moran.annual.lin$observed, Moran.annual.add$observed,
               Moran.annual.int$observed),
  Expected = c(Moran.annual.lin$expected, Moran.annual.add$expected,
               Moran.annual.int$expected),
  sd = c(Moran.annual.lin$sd, Moran.annual.add$sd, Moran.annual.int$sd),
  p.value = c(Moran.annual.lin$p.value, Moran.annual.add$p.value,
              Moran.annual.int$p.value)
)

kable(global.moran.annual)
```

Model	Observed	Expected	sd	p.value
simple linear	-0.0276451	-0.0454545	0.0317115	0.5743823
linear additive	-0.0171313	-0.0454545	0.0317411	0.3722215
interaction	-0.0154402	-0.0454545	0.0318226	0.3455901

```r
# Cold Temp
Moran.cold.lin <- Moran.I(cold.lin$residuals, dists)
Moran.cold.add <- Moran.I(cold.add$residuals, dists)
Moran.cold.int <- Moran.I(cold.int$residuals, dists)

global.moran.cold <- data.frame(
  Model = c("simple linear", "linear additive", "interaction"),
  Observed = c(Moran.cold.lin$observed, Moran.cold.add$observed,
               Moran.cold.int$observed),
  Expected = c(Moran.cold.lin$expected, Moran.cold.add$expected,
               Moran.cold.int$expected),
  sd = c(Moran.cold.lin$sd, Moran.cold.add$sd, Moran.cold.int$sd),
  p.value = c(Moran.cold.lin$p.value, Moran.cold.add$p.value,
              Moran.cold.int$p.value)
)

kable(global.moran.cold)
```

Model	Observed	Expected	sd	p.value
simple linear	-0.0328926	-0.0454545	0.0318228	0.6930295
linear additive	-0.0201217	-0.0454545	0.0318073	0.4257723

Model	Observed	Expected	sd	p.value
interaction	-0.0160914	-0.0454545	0.0319118	0.3575019

```
# Warm Temp
Moran.warm.lin <- Moran.I(warm.lin$residuals, dists)
Moran.warm.add <- Moran.I(warm.add$residuals, dists)
Moran.warm.int <- Moran.I(warm.int$residuals, dists)

global.moran.warm <- data.frame(
  Model = c("simple linear", "linear additive", "interaction"),
  Observed = c(Moran.warm.lin$observed, Moran.warm.add$observed,
               Moran.warm.int$observed),
  Expected = c(Moran.warm.lin$expected, Moran.warm.add$expected,
               Moran.warm.int$expected),
  sd = c(Moran.warm.lin$sd, Moran.warm.add$sd, Moran.warm.int$sd),
  p.value = c(Moran.warm.lin$p.value, Moran.warm.add$p.value,
              Moran.warm.int$p.value)
)

kable(global.moran.warm)
```

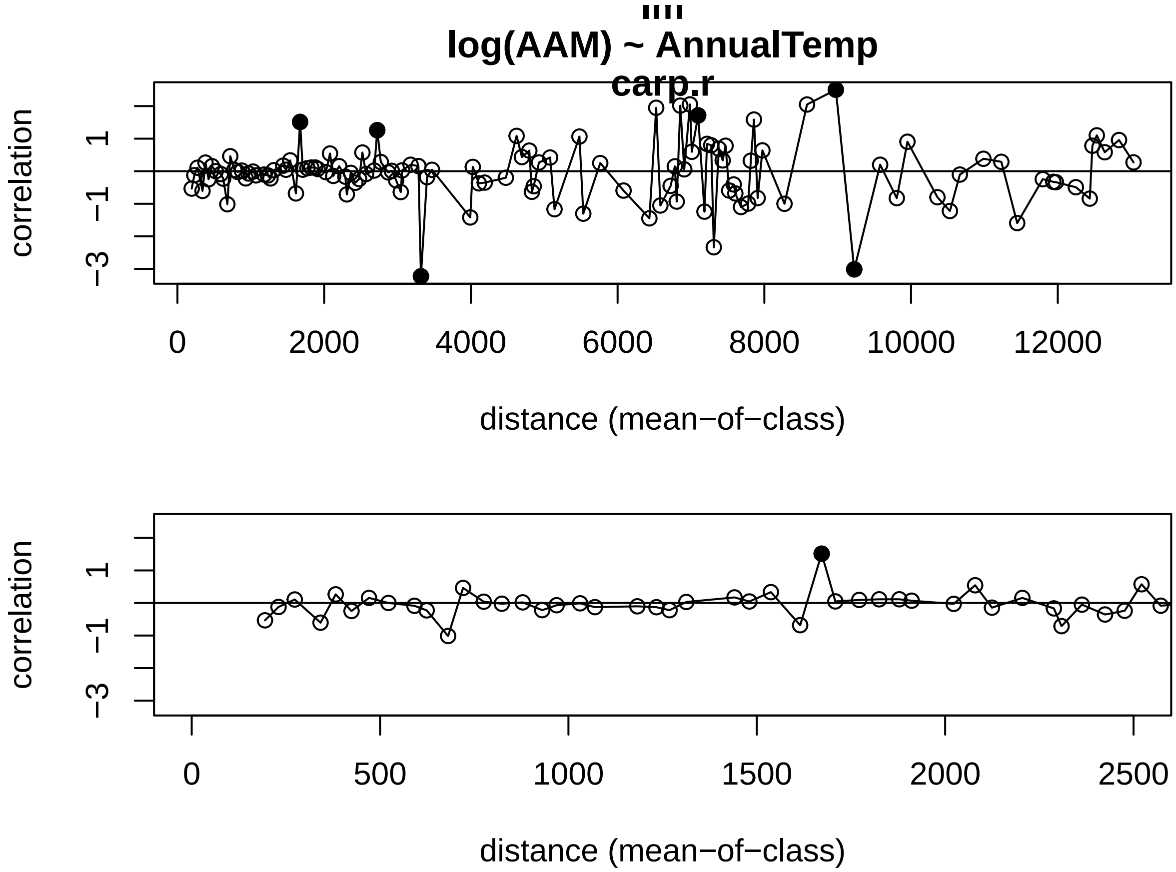
Model	Observed	Expected	sd	p.value
simple linear	-0.0408894	-0.0454545	0.0315415	0.8849196
linear additive	-0.0335627	-0.0454545	0.0315865	0.7065566
interaction	-0.0363573	-0.0454545	0.0315894	0.7733589

There is no global spatial autocorrelation on the entire dataset for all three temperature metrics.

Local spatial autocorrelation

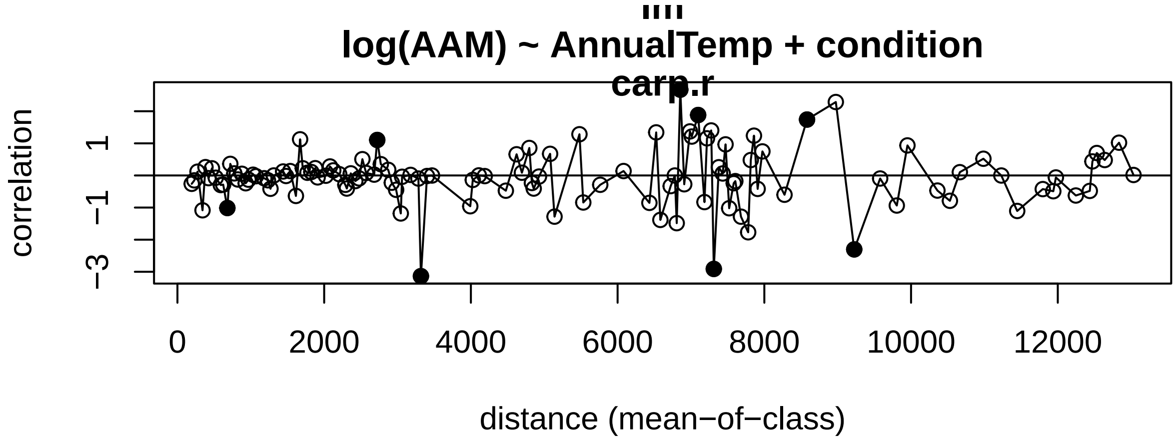
```
## Annual temperature
plot_correlog(annual.lin)
```

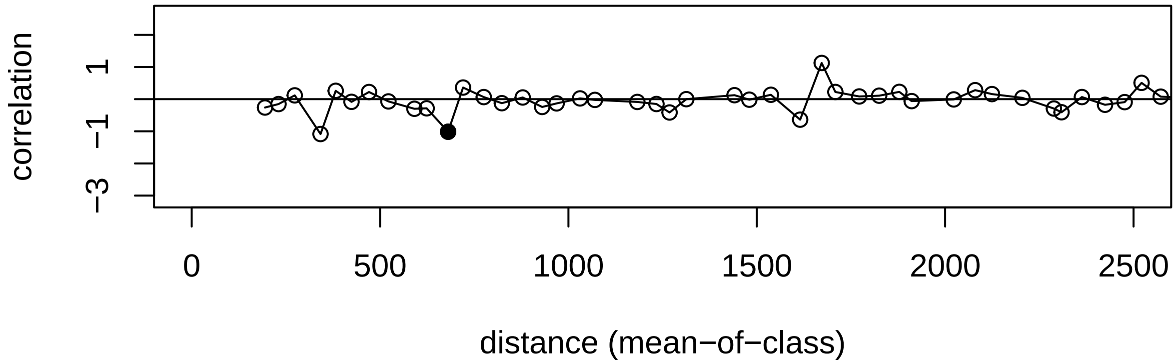
```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   5
```




```
plot_correlog(annual.add)
```

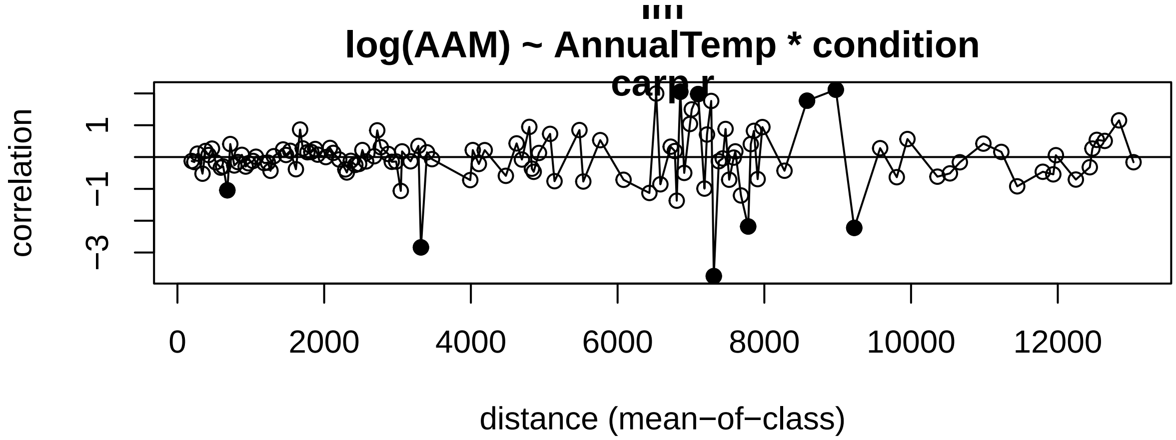
```
## 50  of  500 100  of  500 150  of  500 200  of  500 250  of  500 300  of  500 350  of  500 400  of  5
```

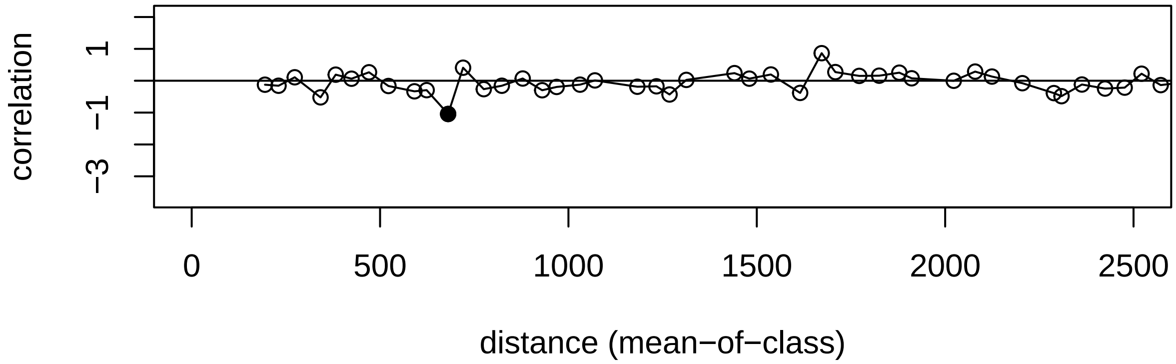





```r
plot_correlog(annual.int)
```

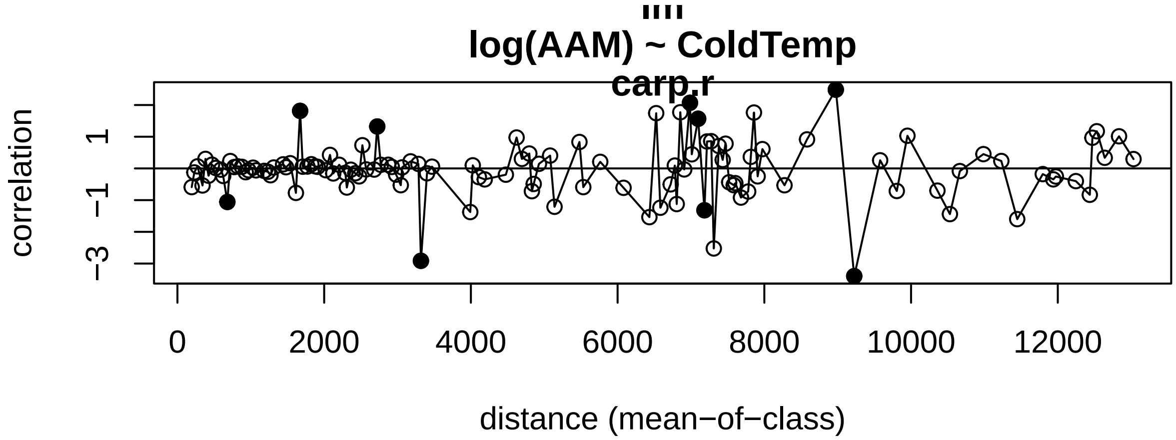
```
## 50  of  500 100  of  500 150  of  500 200  of  500 250  of  500 300  of  500 350  of  500 400  of  50
```

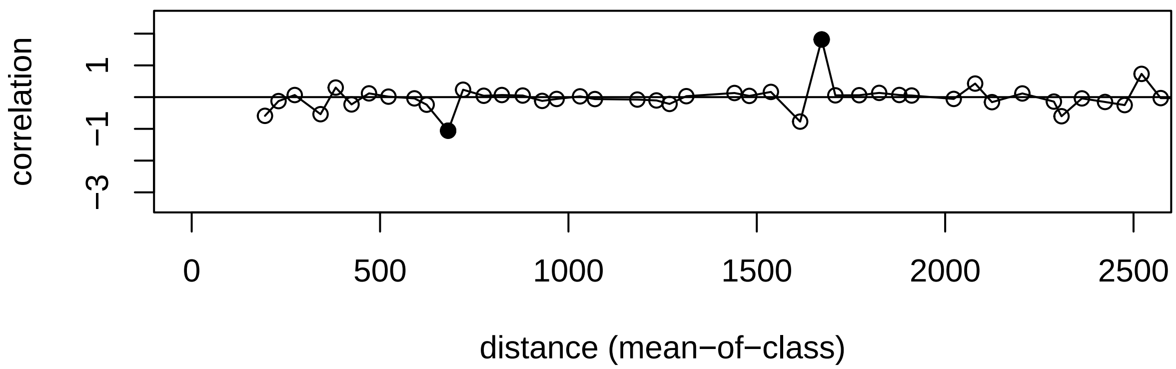





```r
## Cold temperature
plot_correlog(cold.lin)
```

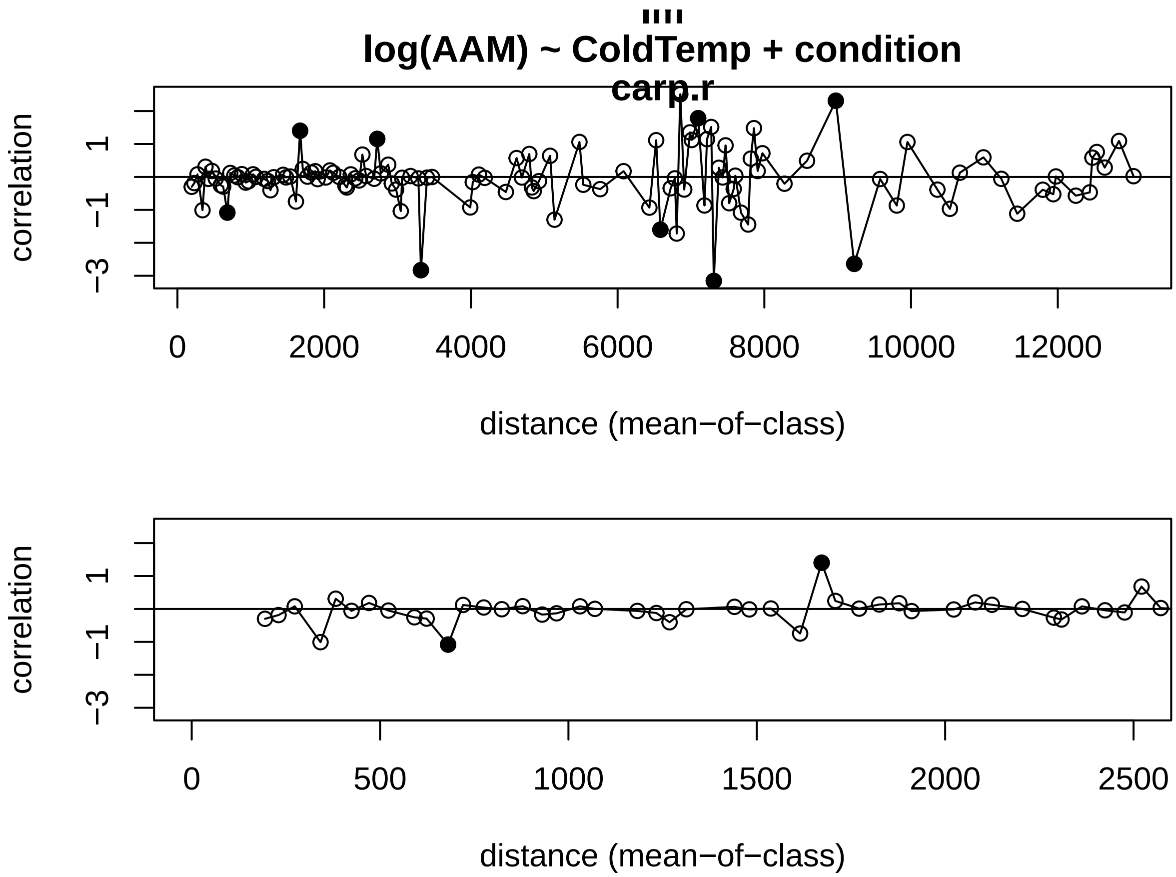
```
## 50  of  500 100  of  500 150  of  500 200  of  500 250  of  500 300  of  500 350  of  500 400  of  50
```





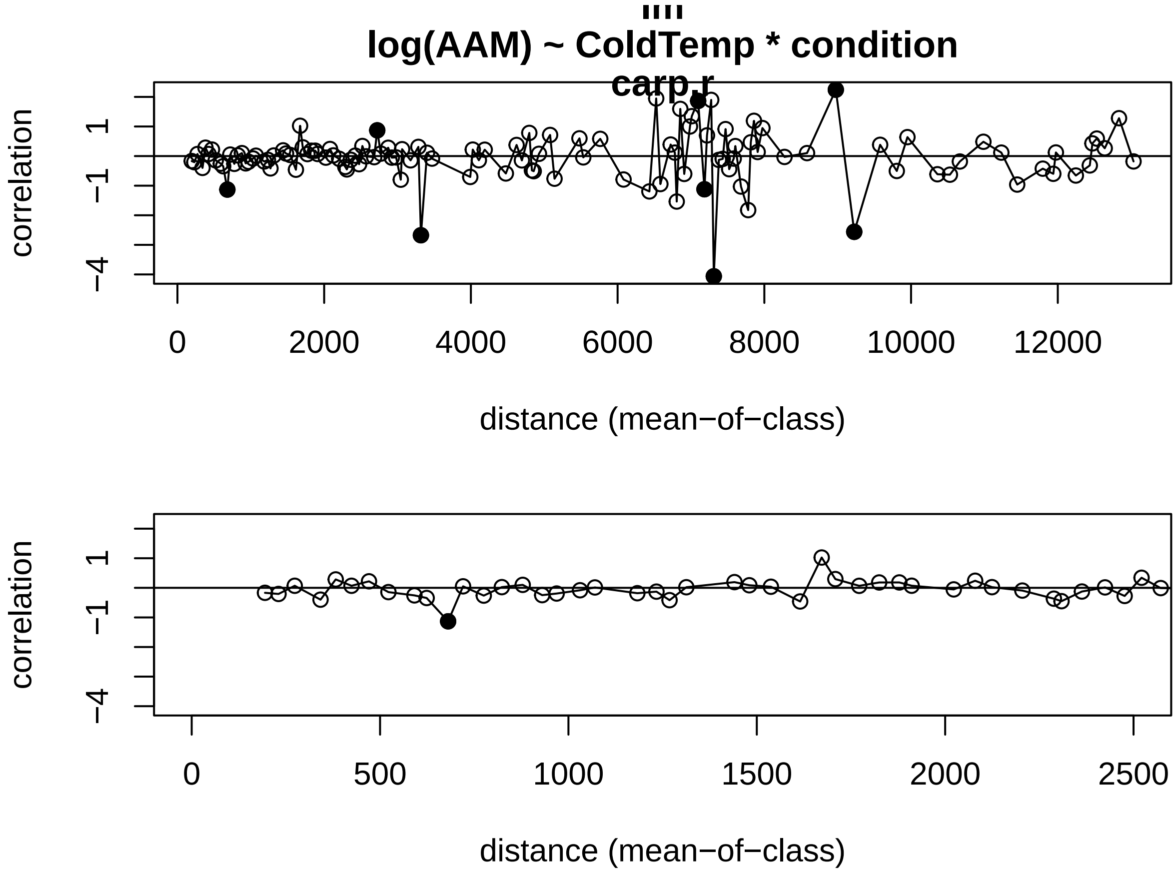

```r
plot_correlog(cold.add)
```

```
## 50  of  500 100  of  500 150  of  500 200  of  500 250  of  500 300  of  500 350  of  500 400  of  50
```



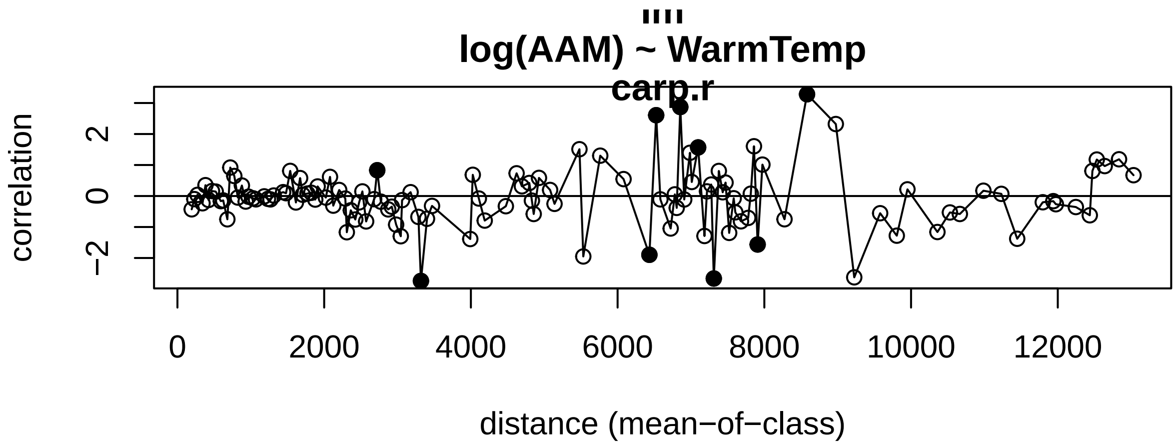

```
plot_correlog(cold.int)
```

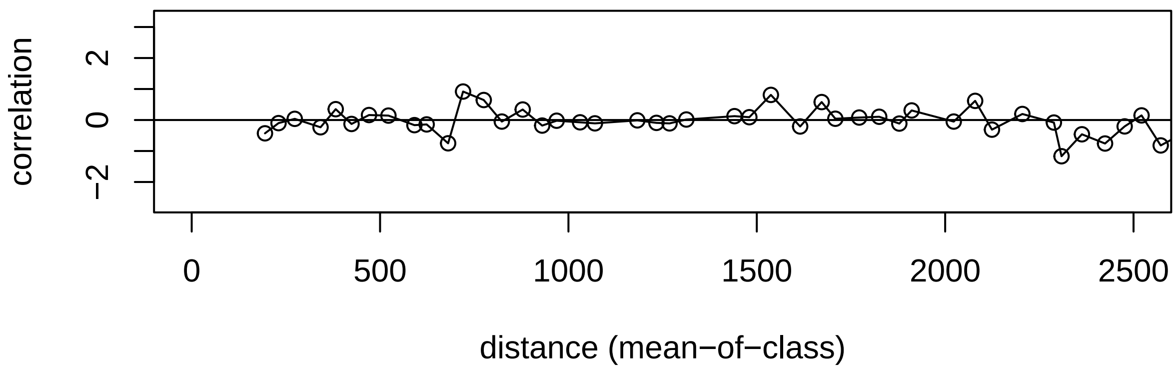
```
## 50  of  500 100  of  500 150  of  500 200  of  500 250  of  500 300  of  500 350  of  500 400  of  50
```




```r
## Warm temperature
plot_correlog(warm.lin)
```

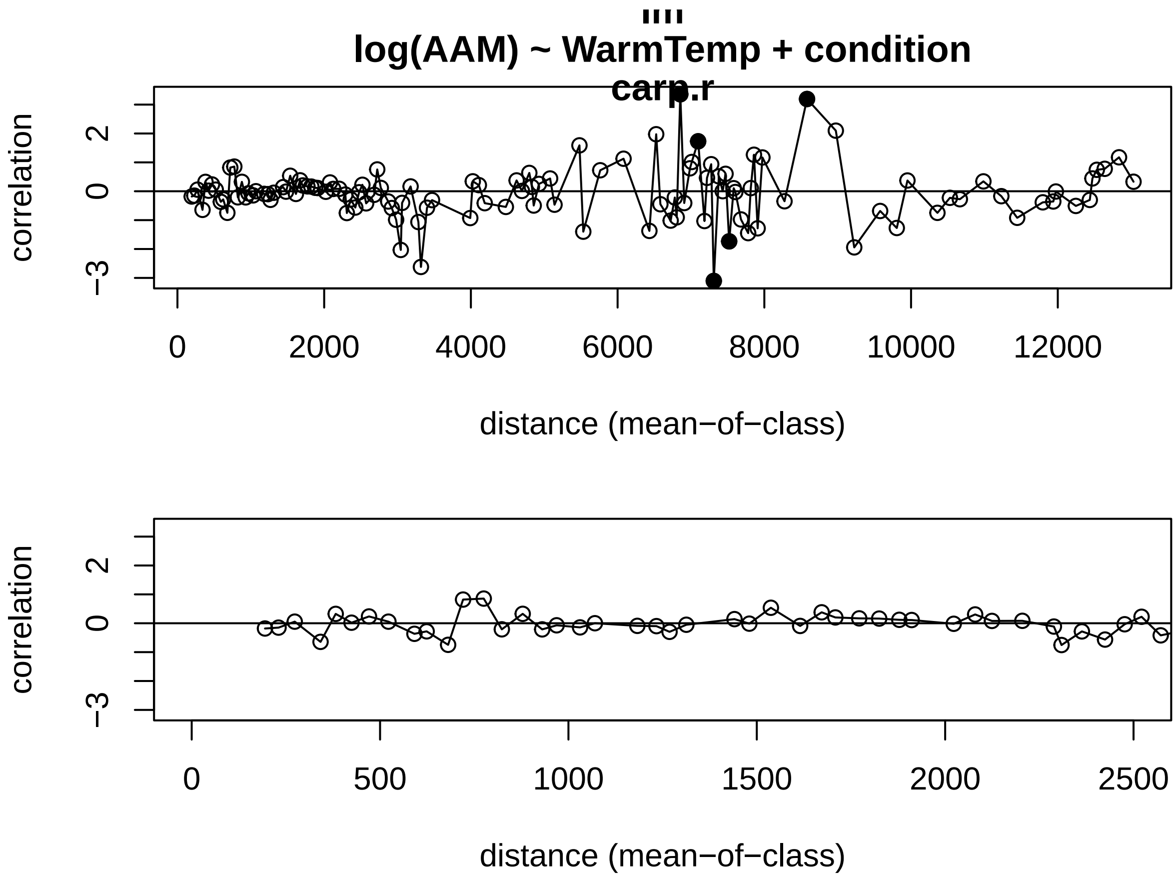
```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   5
```





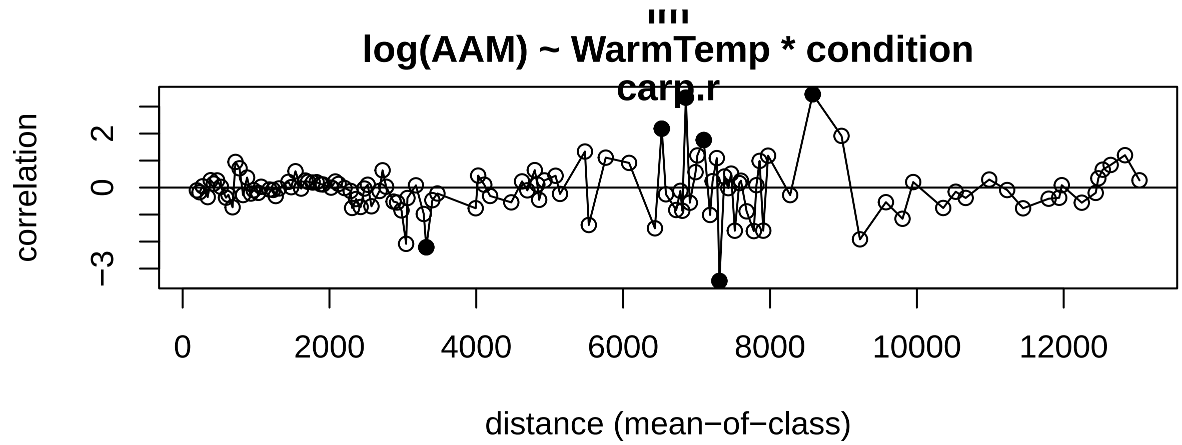

```
plot_correlog(warm.add)
```

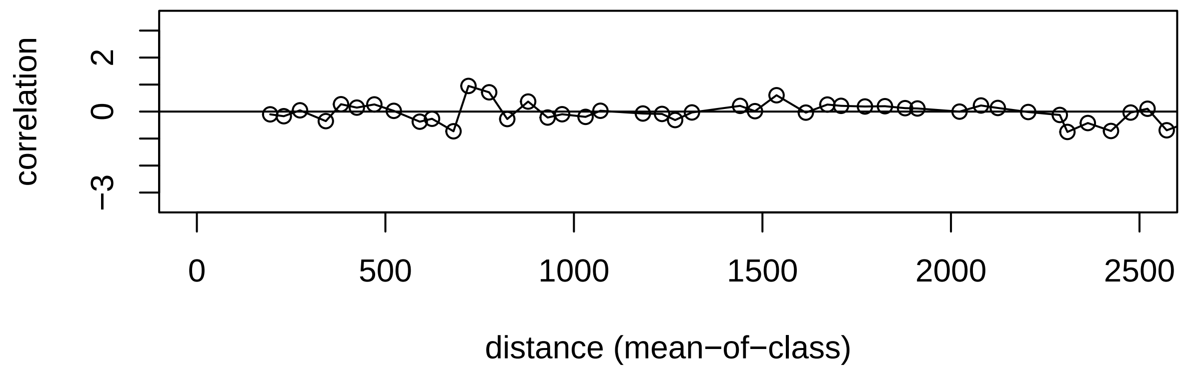
```
## 50 of 500 100 of 500 150 of 500 200 of 500 250 of 500 300 of 500 350 of 500 400 of
```




```
plot_correlog(warm.int)
```

```
## 50   of   500 100  of   500 150  of   500 200  of   500 250  of   500 300  of   500 350  of   500 400  of   50
```





In general, the correlog plots suggest that for both temperature, 550 km is a good distance to reduce local spatial autocorrelation.

If we pick 250 km as the cutoff distance for sub-sampling, we would have 20 location points.

If we pick 550 km as the cutoff distance for sub-sampling, we would have only 12 location points.

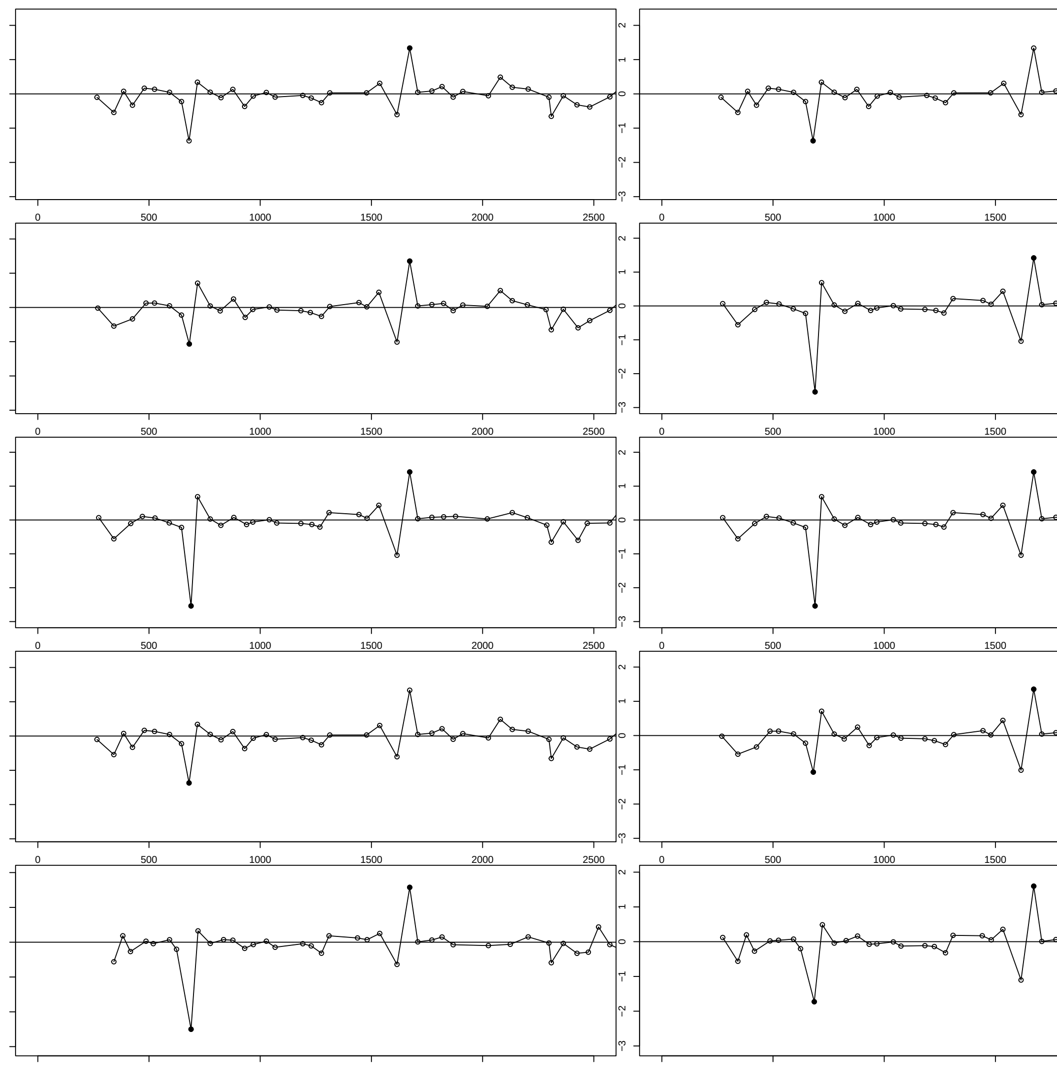
Section 2: Subsampling

Subsampling at 250km

Here, we try to sub-sample at 250km and check the local moran's I (correlog plot) after sub-sampling.

```
# Store the r2 values
r2.250.raw <- matrix(NA,10, 2)
colnames(r2.250.raw) <- c("annual","cold")

table(carp.r$spatial.code.250)

par(mfrow = c(5, 2))
par(mar=c(1,1,1,1))

## Check the local moran's results after sub-sampling
for (i in 1:10){
  sub <- carp.r %>% group_by(spatial.code.250) %>% sample_n(size=1)
  reg.sub.annual <- lm(log(sub$AAM)~sub$AnnualTemp)
  test <- correlog(sub$longitude, sub$latitude, reg.sub.annual$residuals,
```

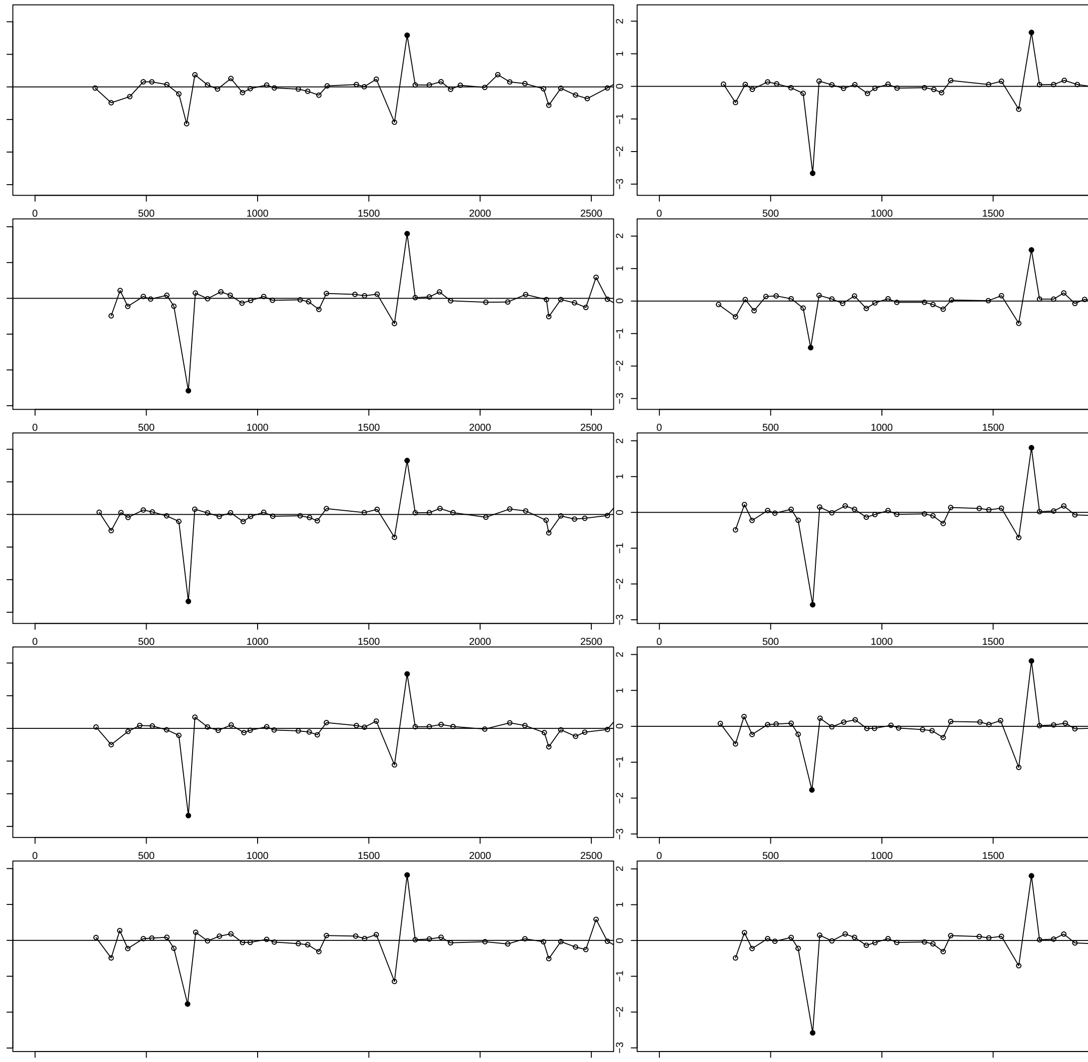
```
                    increment=50, resamp=500, latlon=T)
  plot(test, main="", xlim=c(0,2500))
  abline(h=0)
  r2.250.raw[i,1] <- summary(reg.sub.annual)$adj.r.squared #get the r2
}
```



Local moran's I - annual 250

```
table(carp.r$spatial.code.250)

par(mfrow = c(5, 2))
par(mar=c(1,1,1,1))

## Check the local moran's results after sub-sampling
for (i in 1:10){
  sub <- carp.r %>% group_by(spatial.code.250) %>% sample_n(size=1)
  reg.sub.cold <- lm(log(sub$AAM)~sub$ColdTemp)
  test <- correlog(sub$longitude, sub$latitude, reg.sub.cold$residuals,
                    increment=50, resamp=500, latlon=T)
  plot(test, main="", xlim=c(0,2500))
```

```r
  abline(h=0)
  r2.250.raw[i,2] <- summary(reg.sub.cold)$adj.r.squared #get the r2
}
```



Local moran's I - cold 250

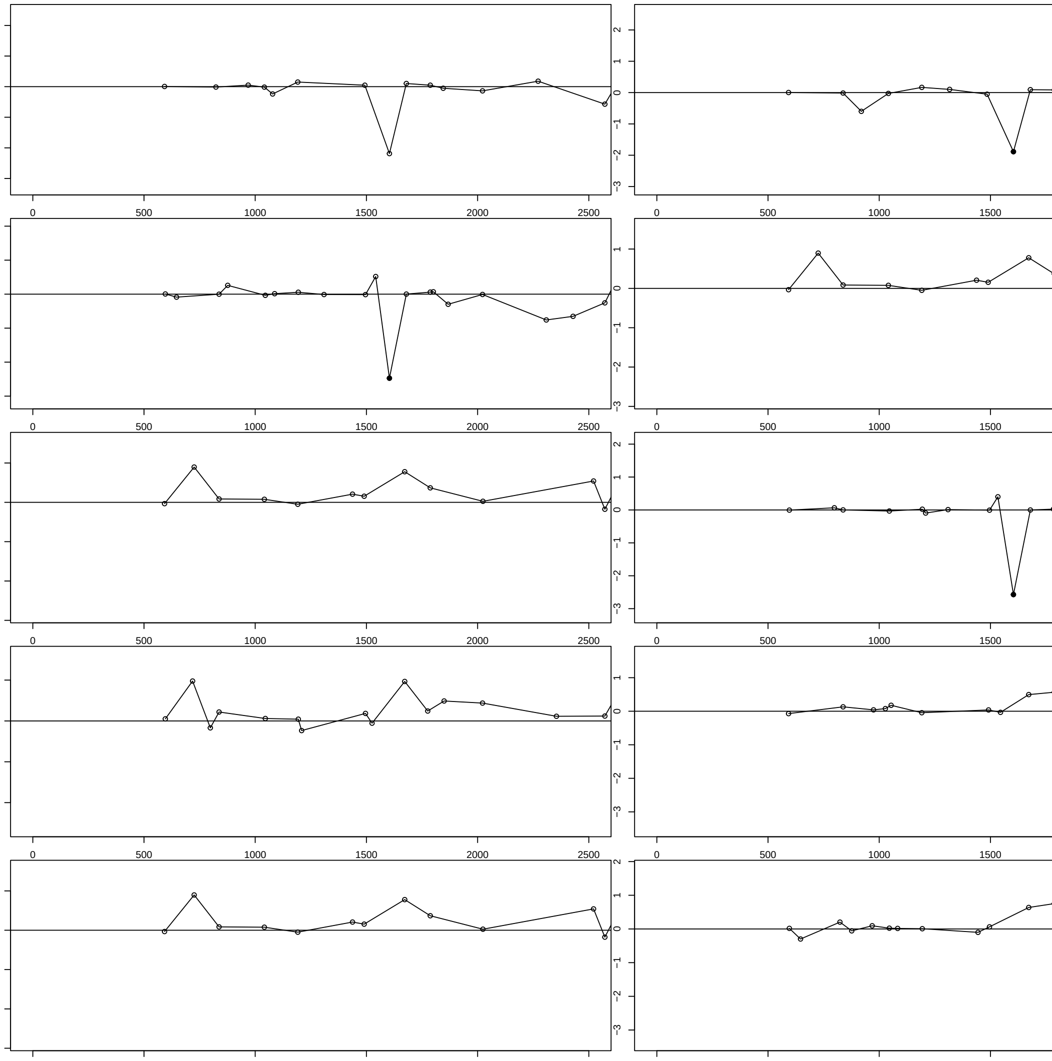
Subsampling at 250 km does not reduce the spatial autocorrelation in our dataset.

Subsampling at 550km

Now, we try to subsample at a larger distance - 550km.

```r
table(carp.r$spatial.code.550)

# Store the r2 values
r2.550.raw <- matrix(NA,10, 2)
colnames(r2.550.raw) <- c("annual","cold")

par(mfrow = c(5, 2))
par(mar=c(1,1,1,1))
```

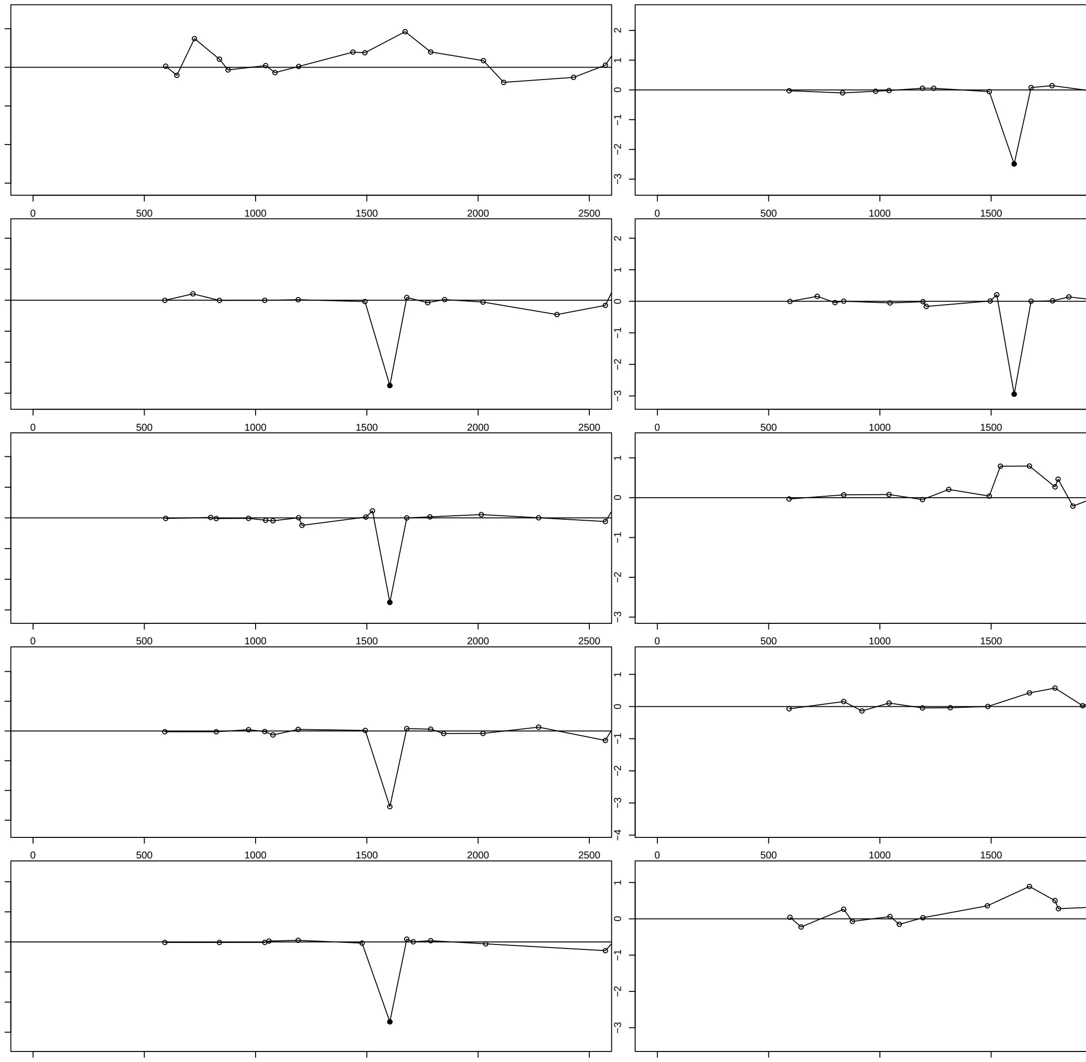
```
## Check the local moran's results after sub-sampling
for (i in 1:10){
  sub <- carp.r %>% group_by(spatial.code.550) %>% sample_n(size=1)
  reg.sub.annual <- lm(log(sub$AAM)~sub$AnnualTemp)
  test <- correlog(sub$longitude, sub$latitude, reg.sub.annual$residuals,
                   increment=50, resamp=500, latlon=T)
  plot(test, main="", xlim=c(0,2500))
  abline(h=0)
  r2.550.raw[i,1] <- summary(reg.sub.annual)$adj.r.squared #get the r2
}
```



Local moran's I - annual 550

```
table(carp.r$spatial.code.550)

par(mfrow = c(5,2))
par(mar=c(1,1,1,1))

## Check the local moran's results after sub-sampling
for (i in 1:10){
```

```
  sub <- carp.r %>% group_by(spatial.code.550) %>% sample_n(size=1)
  reg.sub.cold <- lm(log(sub$AAM)~sub$ColdTemp)
  test <- correlog(sub$longitude, sub$latitude, reg.sub.cold$residuals,
                   increment=50, resamp=500, latlon=T)
  plot(test, main="", xlim=c(0,2500))
  abline(h=0)
  r2.550.raw[i,2] <- summary(reg.sub.cold)$adj.r.squared #get the r2
}
```



Local moran's I - cold 550

We have seen that subsampling at 550 km would largely reduce the spatial autocorrelation in our dataset at low distances, even though we are only left with 12 data points.

Compare the r2 values

```
r2 <- data.frame(
  Time = c("Original", "Sub 250","Sub 550"),
  Annual = c(summary(annual.lin)$adj.r.squared,
             mean(unique(r2.250.raw[,1])),
             mean(unique(r2.550.raw[,1]))),
```

```
  Cold = c(summary(cold.lin)$adj.r.squared,
           mean(unique(r2.250.raw[,2])),
           mean(unique(r2.550.raw[,2])))
)
kable(r2)
```

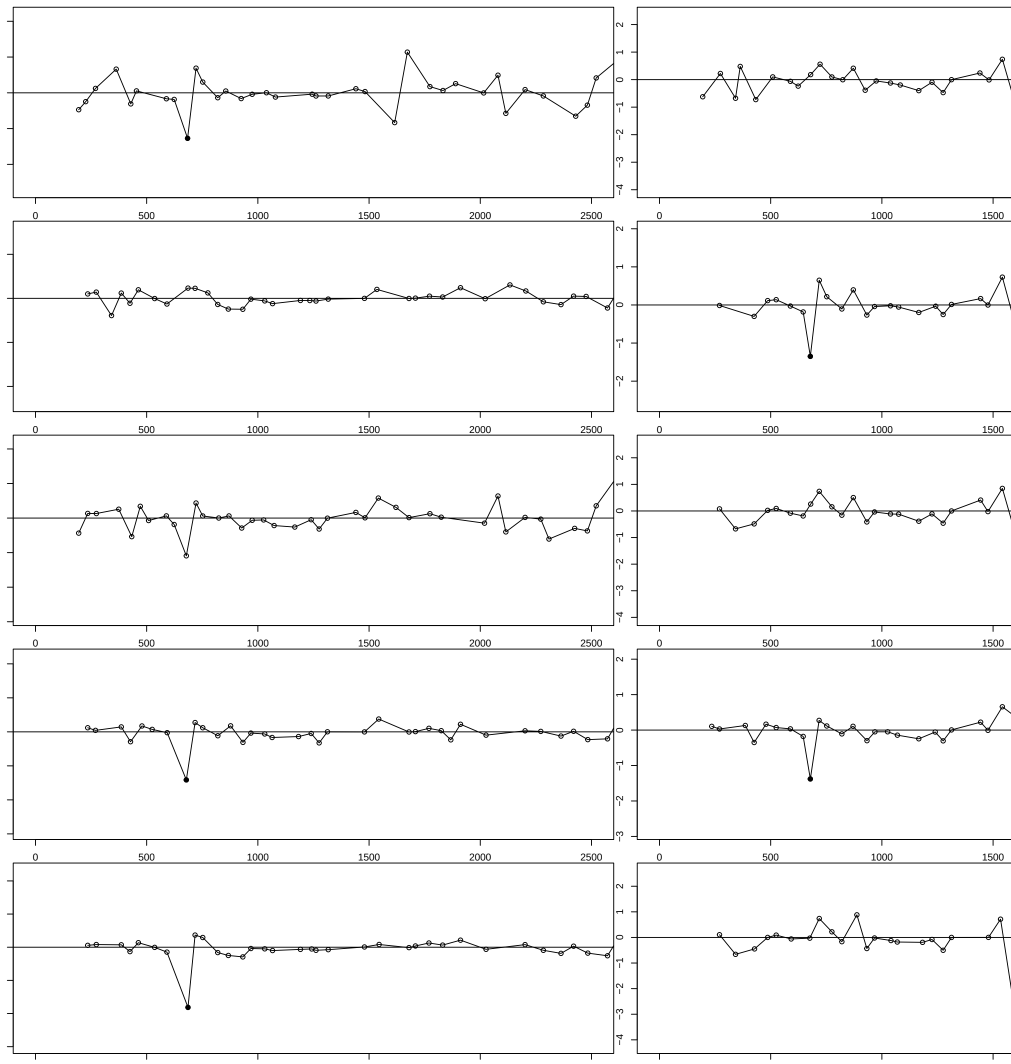
Time	Annual	Cold
Original	0.2533656	0.2751848
Sub 250	0.2509058	0.2703733
Sub 550	0.1007193	0.1507836

However, subsampling at 550 km would reduce our R2 value as well, suggesting that the model would not fit for the sub-dataset.

Latitudinal stratification

Now we would like to explore if stratified sub-sampling using latitude would reduce our spatial autocorrelation at low distances.

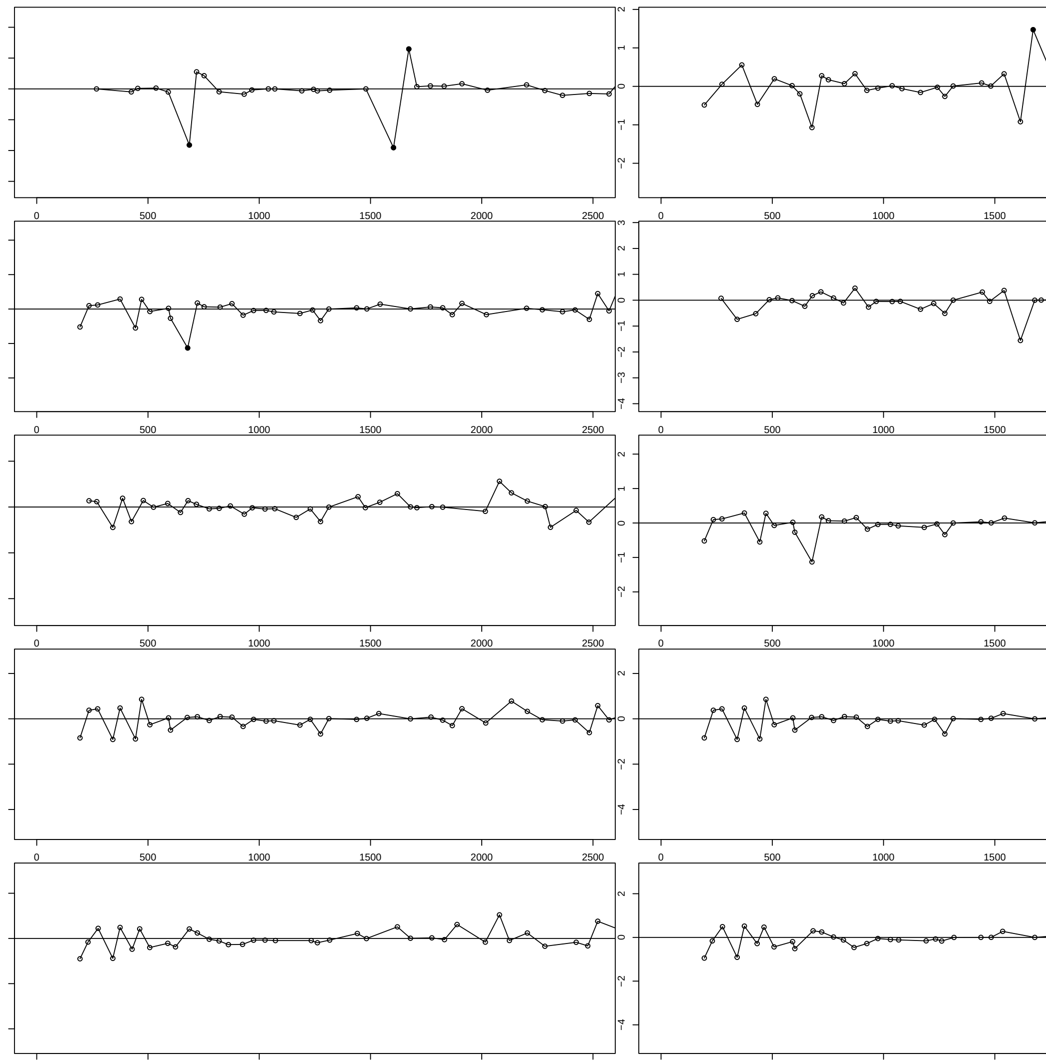
```
table(carp.r$lat.one)

par(mfrow = c(5,2))
par(mar=c(1,1,1,1))

for (i in 1:10){
  sub <- carp.r %>% group_by(lat.one) %>% sample_n(size=1)
  reg.sub.annual <- lm(log(sub$AAM)~sub$AnnualTemp)
  test <- correlog(sub$longitude, sub$latitude, reg.sub.annual$residuals,
                   increment=50, resamp=500, latlon=T)
  plot(test, main="", xlim=c(0,2500))
  abline(h=0)
}
```



one degree latitude bin - annual

```
dev.off()
```

```
table(carp.r$lat.one)

par(mfrow = c(5,2))
par(mar=c(1,1,1,1))

for (i in 1:10){
  sub <- carp.r %>% group_by(lat.one) %>% sample_n(size=1)
  reg.sub.cold <- lm(log(sub$AAM)~sub$ColdTemp)
  test <- correlog(sub$longitude, sub$latitude, reg.sub.cold$residuals,
                   increment=50, resamp=500, latlon=T)
  plot(test, main="", xlim=c(0,2500))
  abline(h=0)
}
```



one degree latitude bin - cold

```
dev.off()
```

- For some subsamples, the negative autocorrelation at 550 km is reduced.
- It seems that the South Ukarine data point is the main cause that we have negative spatial autocorrelation at around 550 km. Therefore, we ought to remove this datapoint from our analyses.

Section 3: Spatial autocorrelation without SU

Now we would like to re-examine the global and local spatial autocorrelation with the South Ukraine data point removed.

Model construction

Again, we first need to define all 9 models (3 models for each temperature). This time we remove the SU data point.

```
# Define the models for AnnualTemp
annual.lin <- lm(log(AAM)~AnnualTemp, data = carp.new)
annual.add <- lm(log(AAM)~AnnualTemp+condition, data = carp.new)
annual.int <- lm(log(AAM)~AnnualTemp*condition, data = carp.new)
```

```r
# Define the models for ColdTemp
cold.lin <- lm(log(AAM)~ColdTemp, data = carp.new)
cold.add <- lm(log(AAM)~ColdTemp+condition, data = carp.new)
cold.int <- lm(log(AAM)~ColdTemp*condition, data = carp.new)

# Define the model for WarmTemp
warm.lin <- lm(log(AAM)~WarmTemp, data = carp.new)
warm.add <- lm(log(AAM)~WarmTemp+condition, data = carp.new)
warm.int <- lm(log(AAM)~WarmTemp*condition, data = carp.new)
```

Global Moran.I

```r
## Make spatial dataframe
coords <- data.frame("long"=location.new[,3],"lat"=location.new[,2])
df <- data.frame(a = 1:nrow(location.new[3]))
spatial.data <- SpatialPointsDataFrame(coords,df,proj4string = tmin.1979@crs)

# Get a distance matrix from all points
dists <- spDists(spatial.data, longlat = TRUE)


## Global Moran.I
# Annual Temp
Moran.annual.lin <- Moran.I(annual.lin$residuals, dists)
Moran.annual.add <- Moran.I(annual.add$residuals, dists)
Moran.annual.int <- Moran.I(annual.int$residuals, dists)

global.moran.annual <- data.frame(
  Model = c("simple linear", "linear additive", "interaction"),
  Observed = c(Moran.annual.lin$observed, Moran.annual.add$observed,
               Moran.annual.int$observed),
  Expected = c(Moran.annual.lin$expected, Moran.annual.add$expected,
               Moran.annual.int$expected),
  sd = c(Moran.annual.lin$sd, Moran.annual.add$sd, Moran.annual.int$sd),
  p.value = c(Moran.annual.lin$p.value, Moran.annual.add$p.value,
              Moran.annual.int$p.value)
)

kable(global.moran.annual)
```

Model	Observed	Expected	sd	p.value
simple linear	-0.0327532	-0.047619	0.0317899	0.6400513
linear additive	-0.0346096	-0.047619	0.0320210	0.6845372
interaction	-0.0452347	-0.047619	0.0321136	0.9408124

```r
# Cold Temp
Moran.cold.lin <- Moran.I(cold.lin$residuals, dists)
Moran.cold.add <- Moran.I(cold.add$residuals, dists)
Moran.cold.int <- Moran.I(cold.int$residuals, dists)

global.moran.cold <- data.frame(
  Model = c("simple linear", "linear additive", "interaction"),
```

```
  Observed = c(Moran.cold.lin$observed, Moran.cold.add$observed,
               Moran.cold.int$observed),
  Expected = c(Moran.cold.lin$expected, Moran.cold.add$expected,
               Moran.cold.int$expected),
  sd = c(Moran.cold.lin$sd, Moran.cold.add$sd, Moran.cold.int$sd),
  p.value = c(Moran.cold.lin$p.value, Moran.cold.add$p.value,
              Moran.cold.int$p.value)
)

kable(global.moran.cold)
```

Model	Observed	Expected	sd	p.value
simple linear	-0.0284140	-0.047619	0.0318113	0.5460304
linear additive	-0.0295230	-0.047619	0.0319752	0.5714354
interaction	-0.0409897	-0.047619	0.0321866	0.8368166

```
# Warm Temp
Moran.warm.lin <- Moran.I(warm.lin$residuals, dists)
Moran.warm.add <- Moran.I(warm.add$residuals, dists)
Moran.warm.int <- Moran.I(warm.int$residuals, dists)

global.moran.warm <- data.frame(
  Model = c("simple linear", "linear additive", "interaction"),
  Observed = c(Moran.warm.lin$observed, Moran.warm.add$observed,
               Moran.warm.int$observed),
  Expected = c(Moran.warm.lin$expected, Moran.warm.add$expected,
               Moran.warm.int$expected),
  sd = c(Moran.warm.lin$sd, Moran.warm.add$sd, Moran.warm.int$sd),
  p.value = c(Moran.warm.lin$p.value, Moran.warm.add$p.value,
              Moran.warm.int$p.value)
)

kable(global.moran.warm)
```

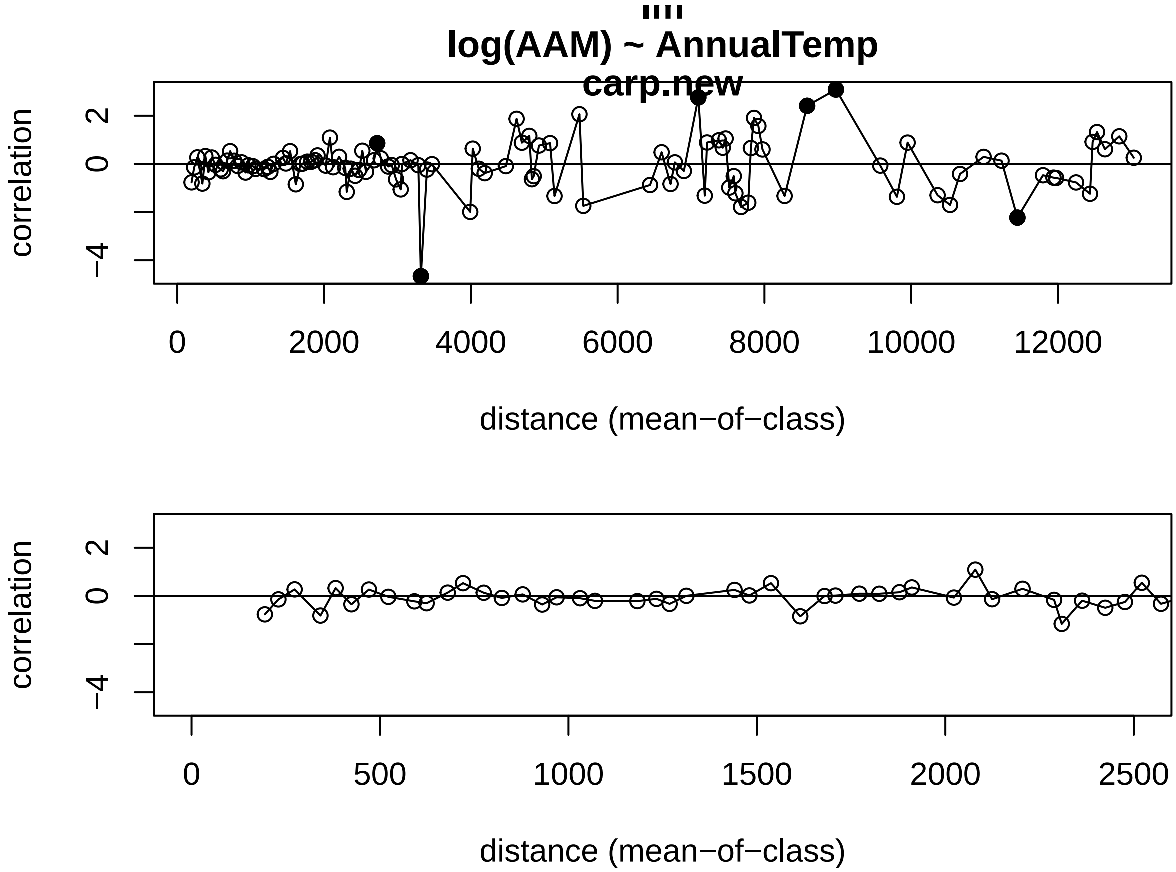
Model	Observed	Expected	sd	p.value
simple linear	-0.0707320	-0.047619	0.0314208	0.4619778
linear additive	-0.0729461	-0.047619	0.0316630	0.4237719
interaction	-0.0841580	-0.047619	0.0315116	0.2462355

There is no global spatial autocorrelation on the entire dataset for all three temperature metrics.

Local spatial autocorrelation

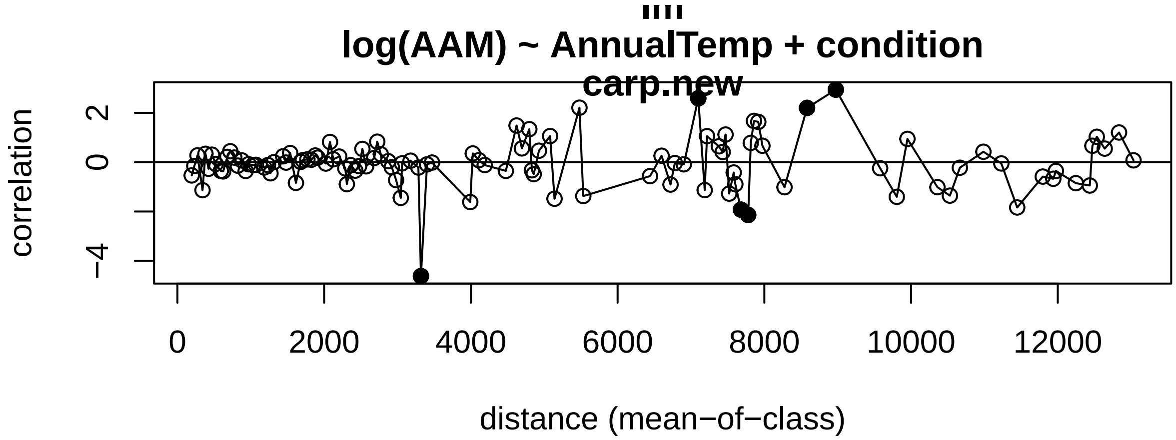
```
## Annual temperature
plot_correlog(annual.lin)
```

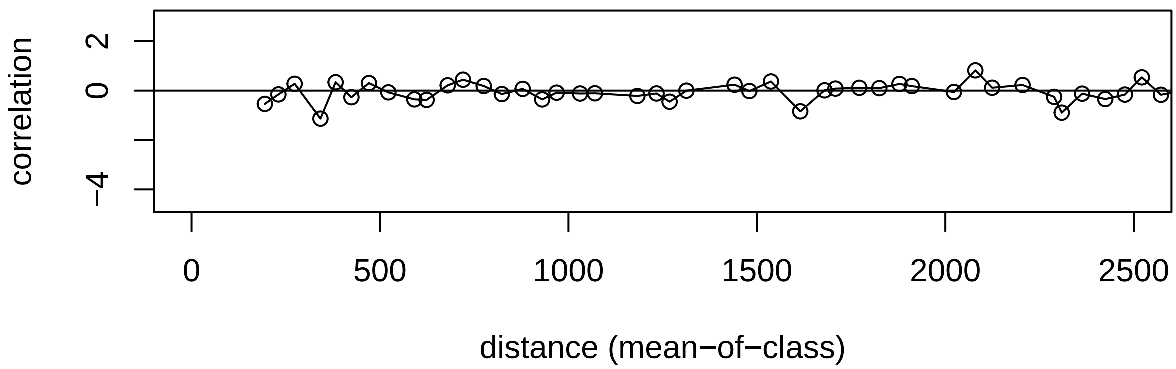
```
## 50   of   500 100  of   500 150  of   500 200  of   500 250  of   500 300  of   500 350  of   500 400  of   50
```




```
plot_correlog(annual.add)
```

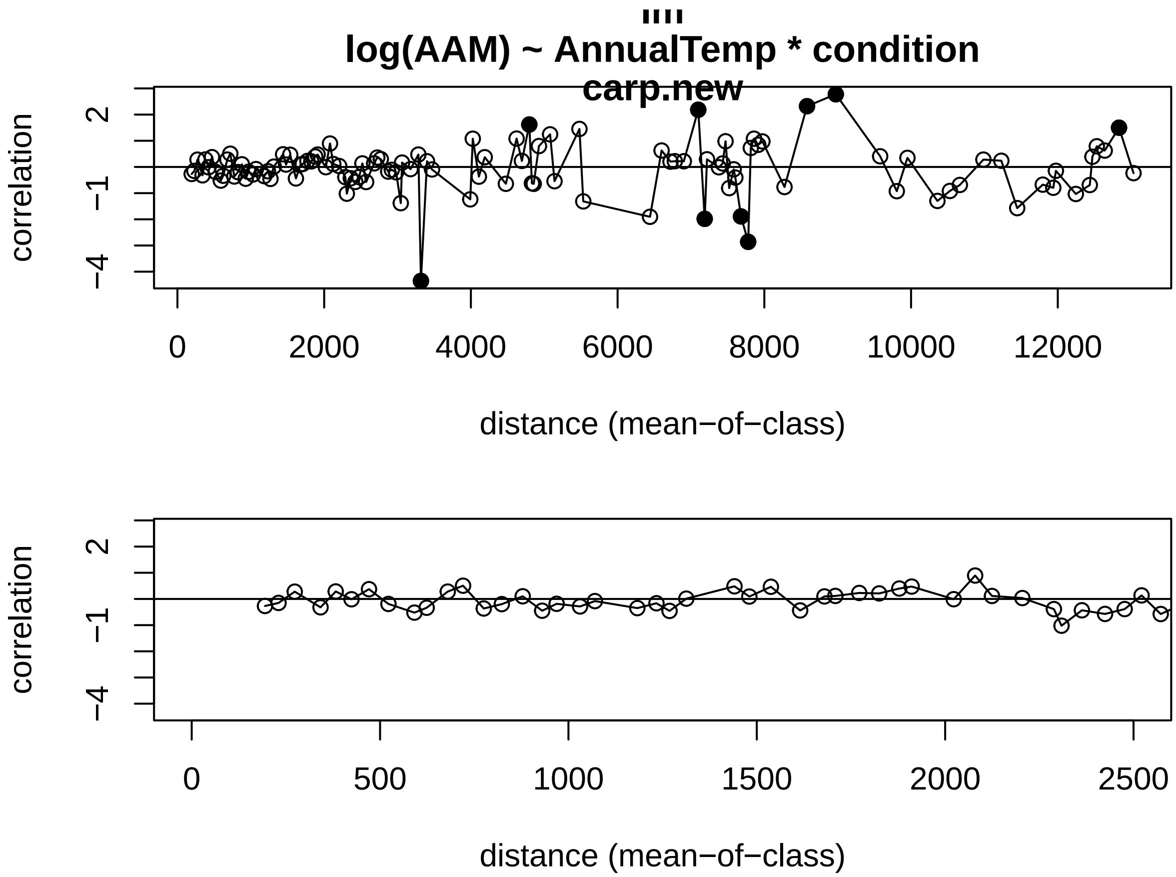
```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   50
```





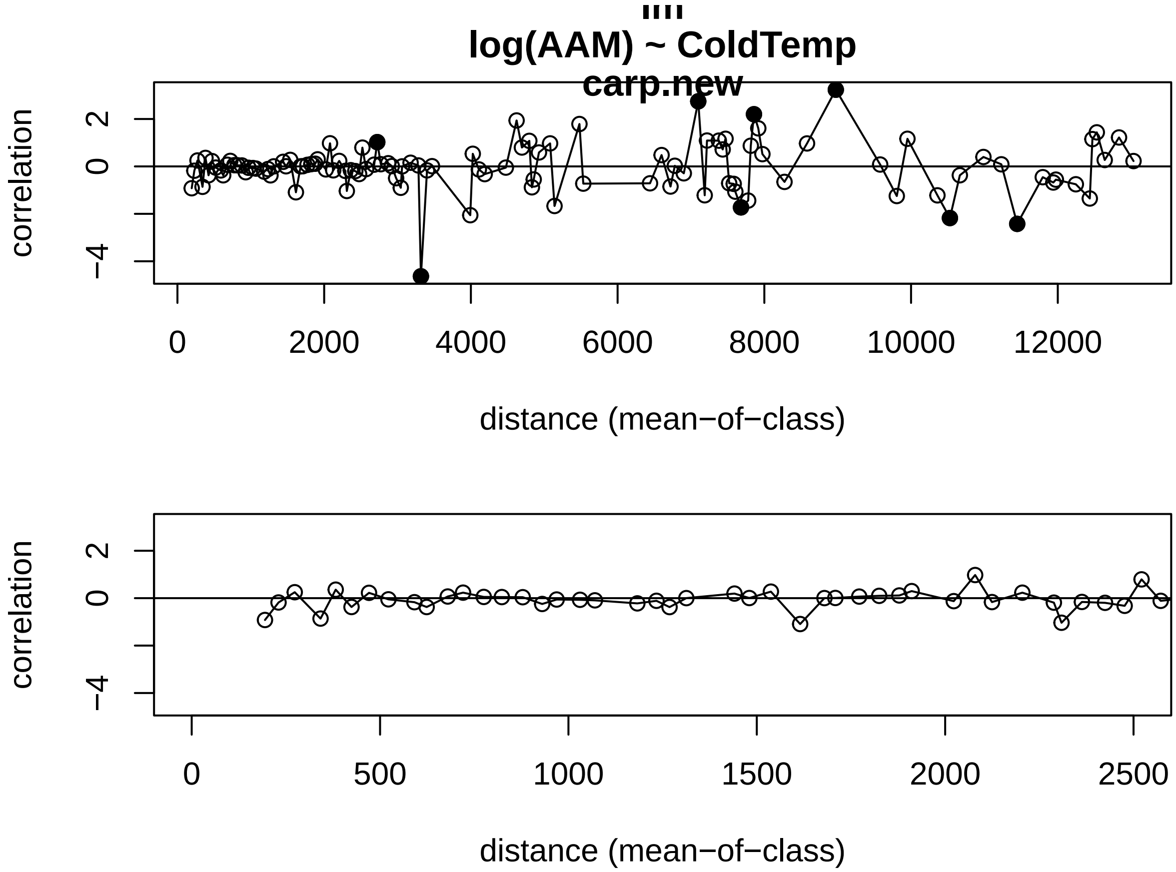

```r
plot_correlog(annual.int)
```

```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   50
```



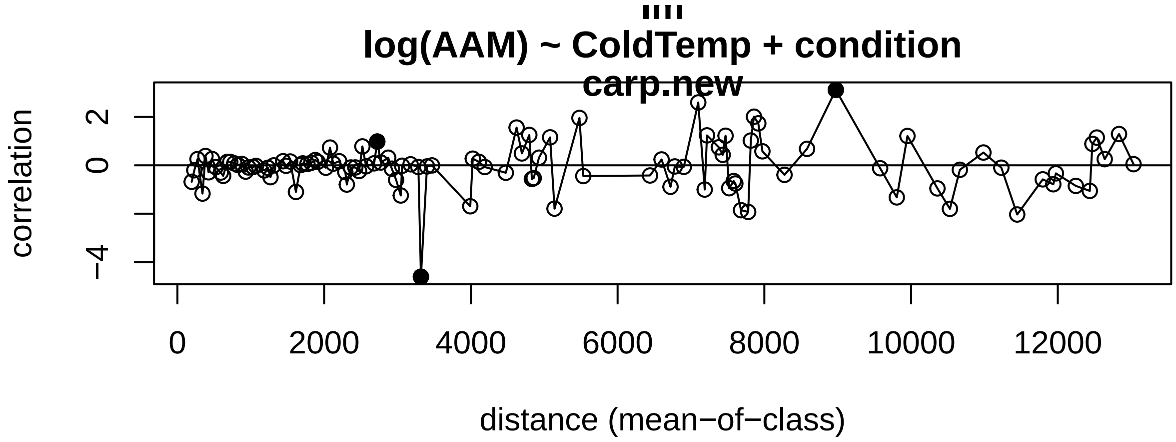

```
## Cold temperature
plot_correlog(cold.lin)
```

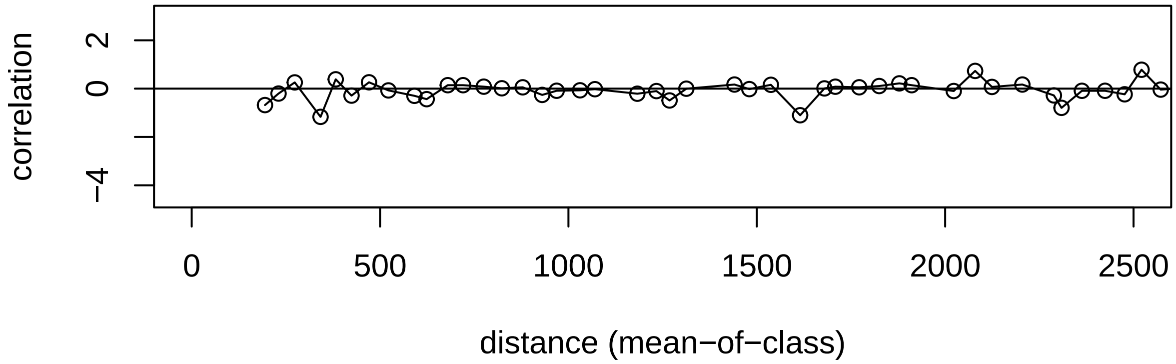
```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   5
```




```
plot_correlog(cold.add)
```

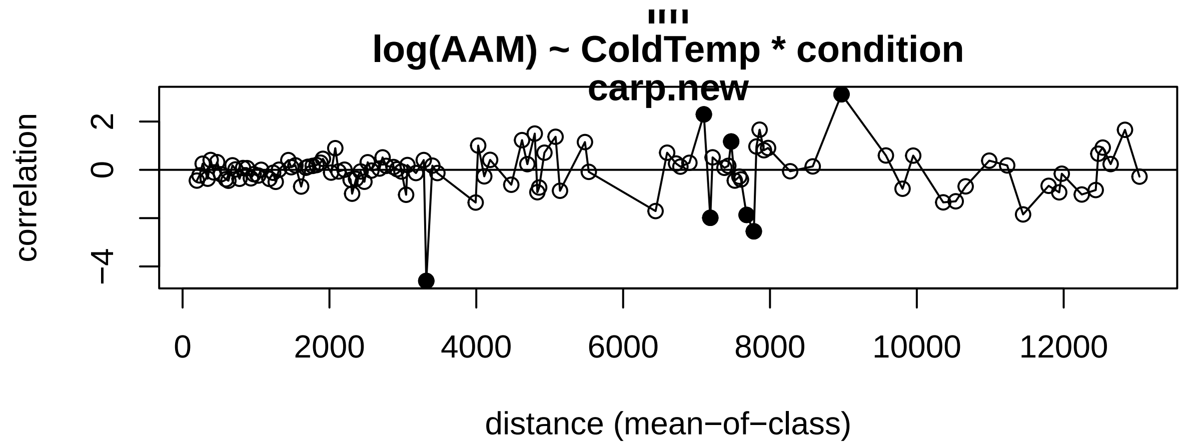
```
## 50  of  500 100  of  500 150  of  500 200  of  500 250  of  500 300  of  500 350  of  500 400  of  5
```

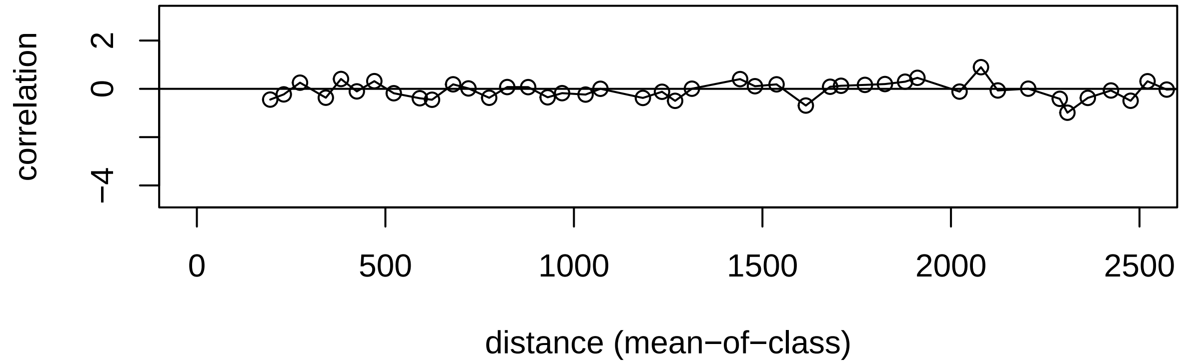





```
plot_correlog(cold.int)
```

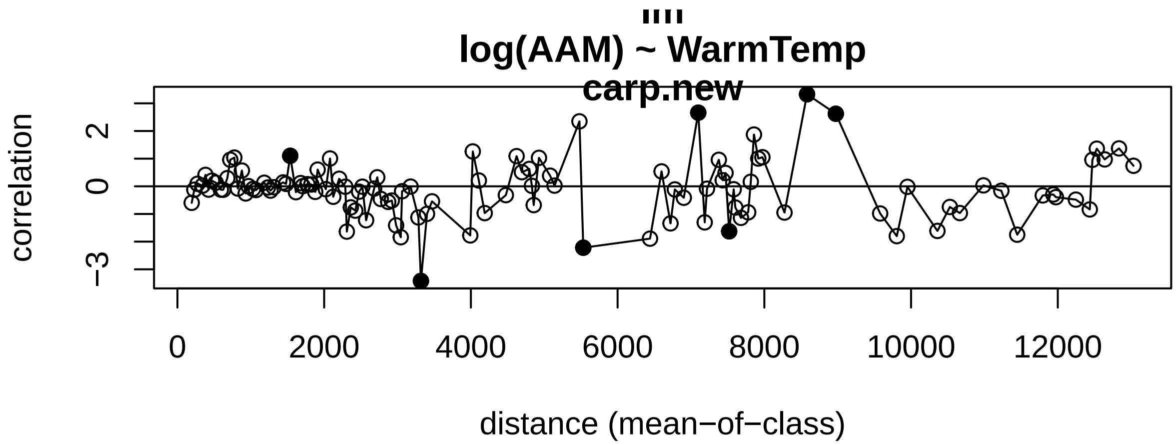
```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   50
```

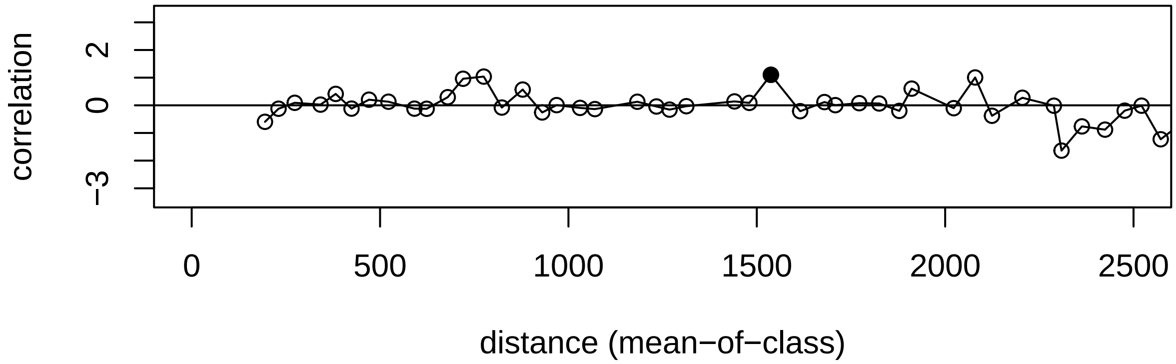





```r
## Warm temperature
plot_correlog(warm.lin)
```

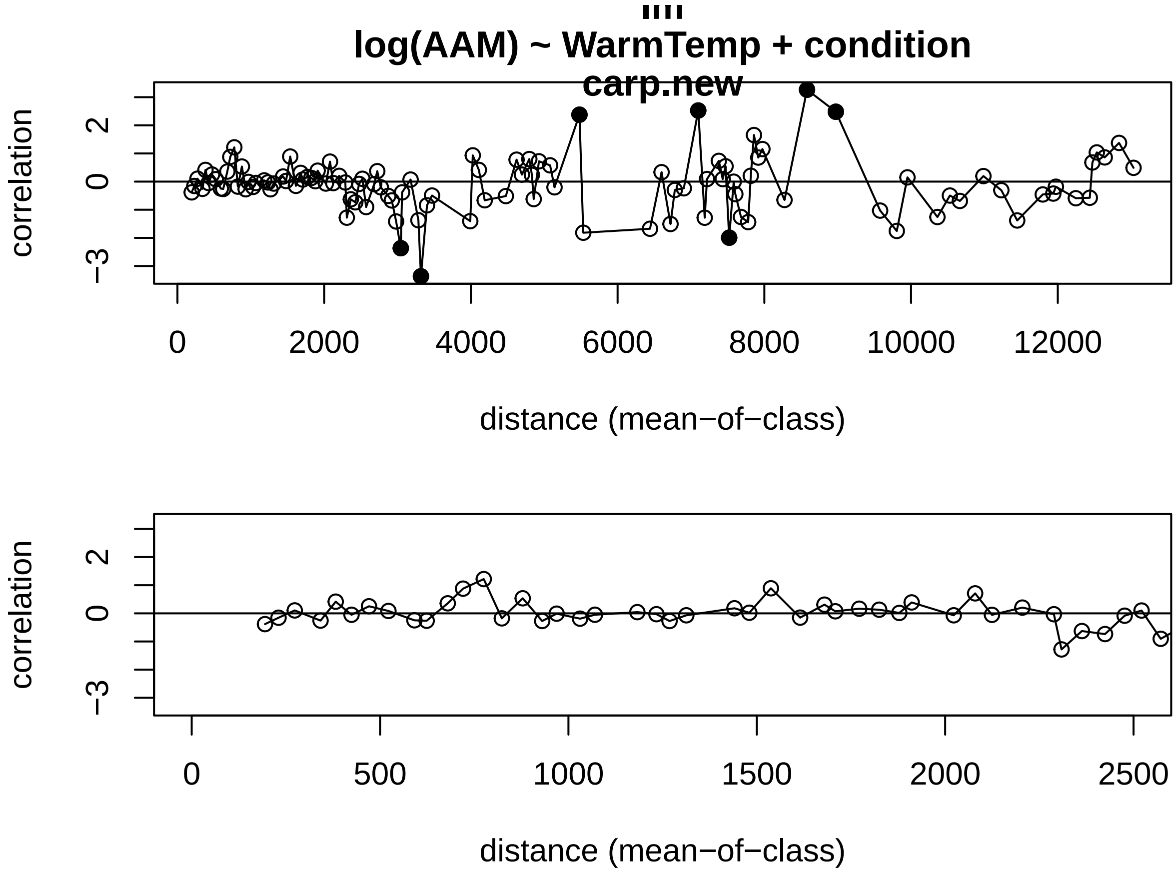
```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   5
```





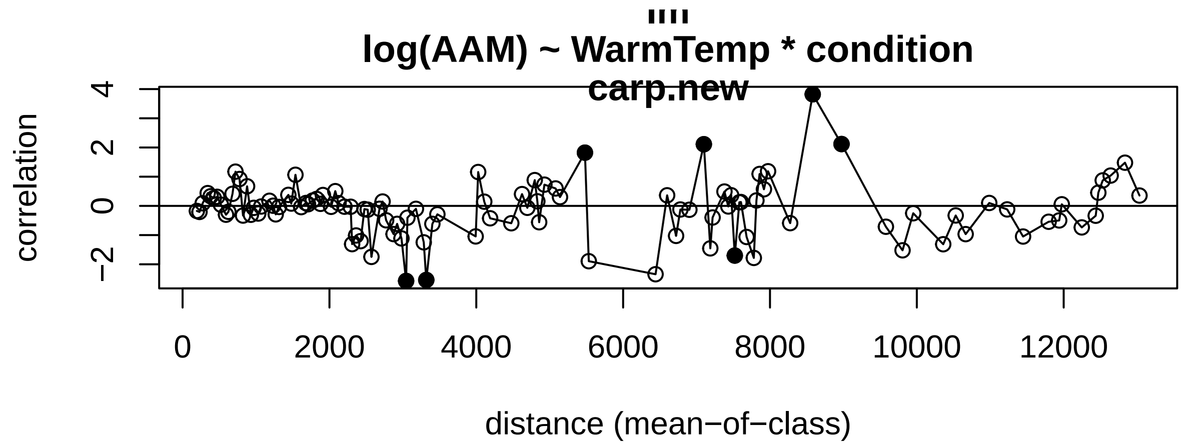

```
plot_correlog(warm.add)
```

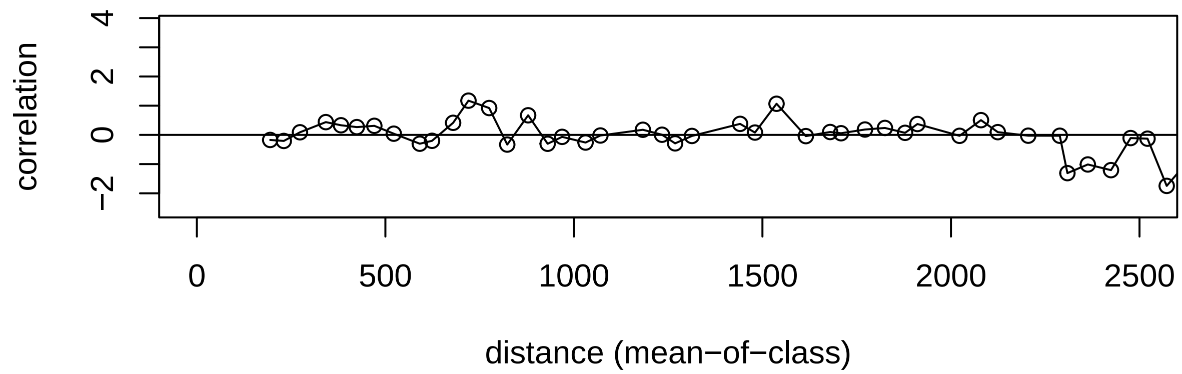
```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   50
```




```
plot_correlog(warm.int)
```

```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   5
```

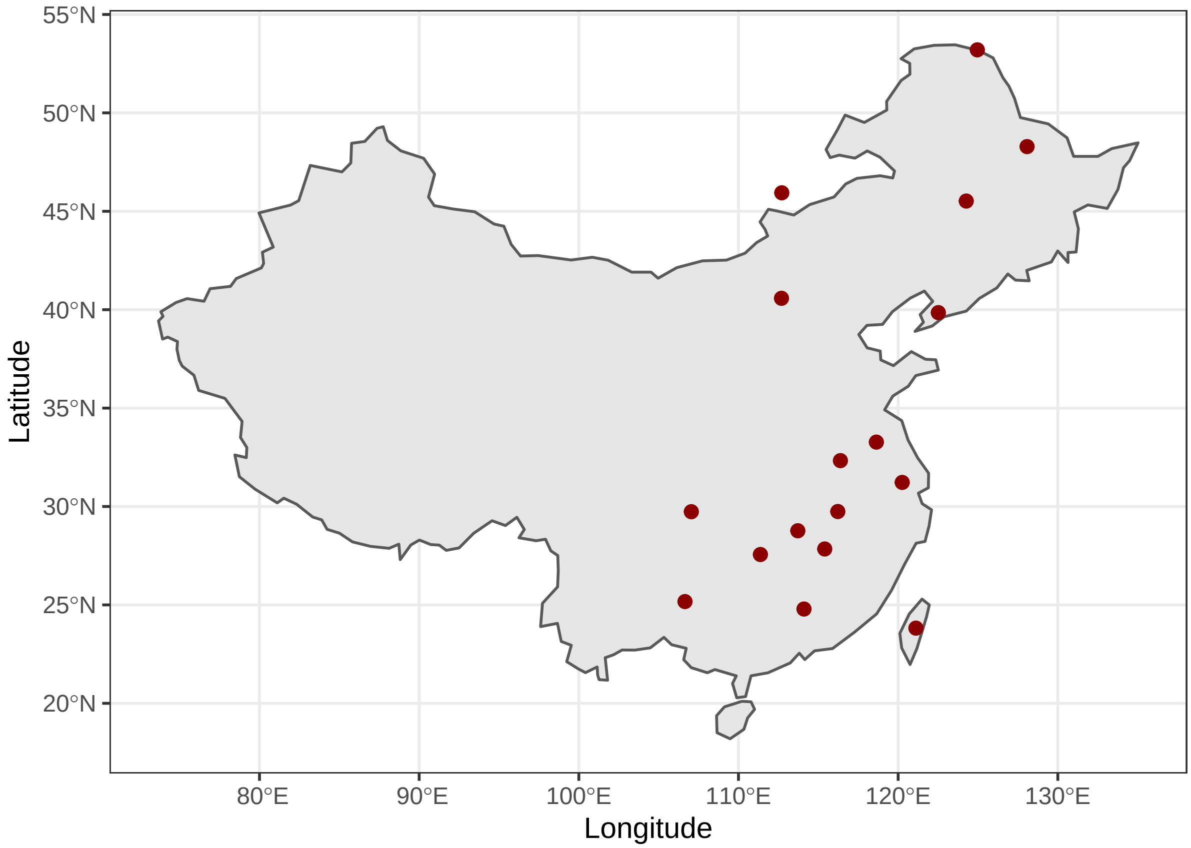




Section 4: China Dataset analyses

Now let's look at the datapoints in China. There are 17 datapoints.

```
## Subset the Chinese dataset
black.china <- carp.r[carp.r$china == "y",]

## Get the world data
world <- ne_countries(scale = "small", returnclass = "sf")

## Plots (china plot)
world %>%
  filter(admin == "China" | admin =="Taiwan") %>%
  # select only China
  ggplot()+
  geom_sf()+
  geom_point(aes(x=longitude, y=latitude), data = black.china,
             size = 2, color = "darkred")+
  theme_bw()+
  xlab("Longitude") + ylab("Latitude")
```



Note that the datapoint outside of the Chinese region is "Northern China". It is calculated through geographical center.

Global moran's I

```
# Run the models
lm.annual.china <- lm(log(AAM)~AnnualTemp, data = black.china)
summary(lm.annual.china)
```

```
##
## Call:
## lm(formula = log(AAM) ~ AnnualTemp, data = black.china)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.166813 -0.109074 -0.008175  0.090822  0.184010
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.997652   0.054821  36.440 4.67e-16 ***
## AnnualTemp  -0.018779   0.003967  -4.734 0.000266 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1174 on 15 degrees of freedom
## Multiple R-squared:  0.5991, Adjusted R-squared:  0.5724
```

```
## F-statistic: 22.42 on 1 and 15 DF,  p-value: 0.000266
```

```r
lm.cold.china <- lm(log(AAM)~ColdTemp, data = black.china)
summary(lm.cold.china)
```

```
##
## Call:
## lm(formula = log(AAM) ~ ColdTemp, data = black.china)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.16135 -0.04743 -0.01076  0.07864  0.17702
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.759579   0.026158  67.268  < 2e-16 ***
## ColdTemp    -0.012227   0.002237  -5.467  6.5e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1072 on 15 degrees of freedom
## Multiple R-squared:  0.6658, Adjusted R-squared:  0.6435
## F-statistic: 29.88 on 1 and 15 DF,  p-value: 6.5e-05
```

```r
## Make spatial dataframe
coords <- data.frame("long"=black.china[,14],"lat"=black.china[,13])
df <- data.frame(a = 1:nrow(black.china[14]))
spatial.data <- SpatialPointsDataFrame(coords,df,proj4string = tmin.1979@crs)

# Get a distance matrix from all points
dists <- spDists(spatial.data, longlat = TRUE)

## Run the Moran.I test on the residuals
Moran.annual <- Moran.I(lm.annual.china$residuals, dists)
Moran.cold <- Moran.I(lm.cold.china$residuals, dists)

global.moran <- data.frame(
  Model = c("Moran.annual", "Moran.cold"),
  Observed = c(Moran.annual$observed, Moran.cold$observed),
  Expected = c(Moran.annual$expected, Moran.cold$expected),
  sd = c(Moran.annual$sd, Moran.cold$sd),
  p.value = c(Moran.annual$p.value, Moran.cold$p.value)
)

kable(global.moran)
```

Model	Observed	Expected	sd	p.value
Moran.annual	-0.0594853	-0.0625	0.0497950	0.9517230
Moran.cold	-0.0450195	-0.0625	0.0495955	0.7244924

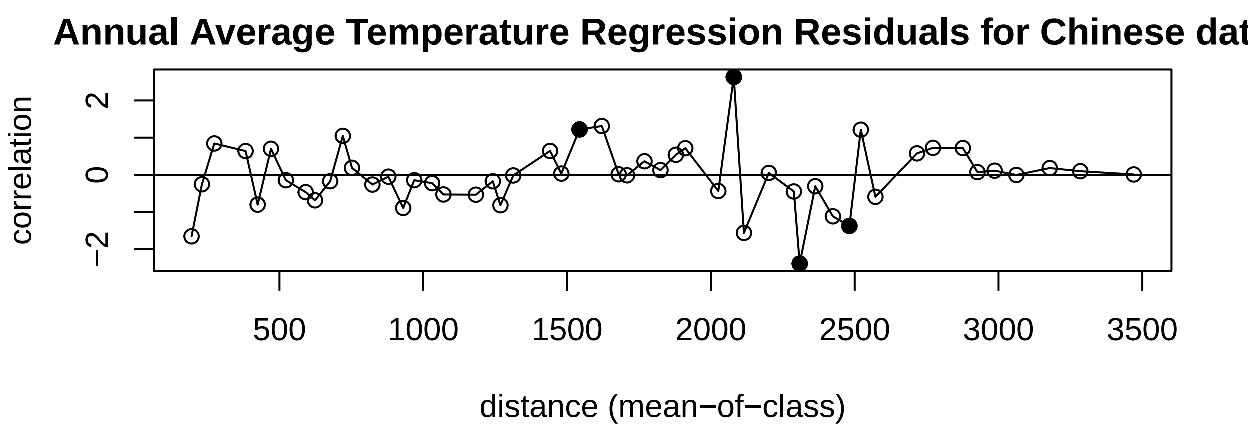
No global spatial autocorrelation in the Chinese dataset.

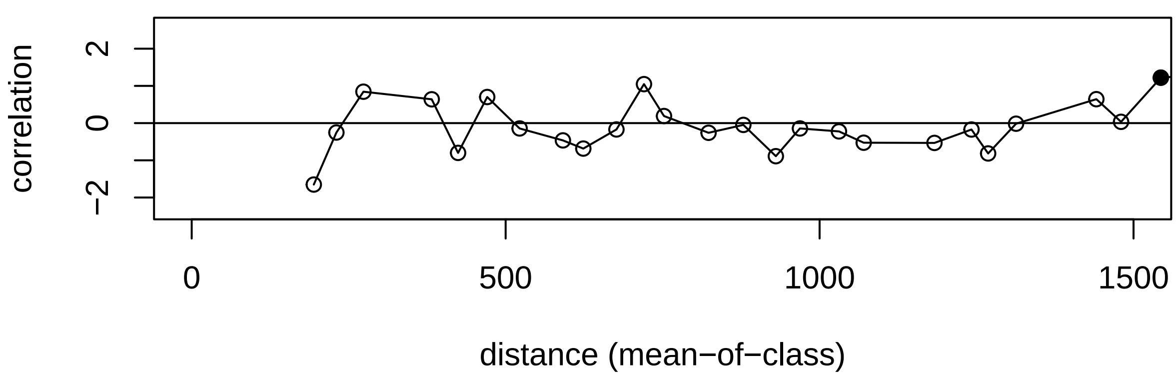
Local moran's I - correlog

```
## Annual temperature
par(mfrow=c(2,1),mar=c(4,4,2,2))

# Use the coorelog function to develop the relationship
test <- correlog(black.china$longitude,
                 black.china$latitude, lm.annual.china$residuals,
                 increment=50, resamp=500, latlon=T)
```

```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   5
```

```
# Plot with the entire distance range
plot(test,
     main="Annual Average Temperature Regression Residuals for Chinese data")
abline(h=0)
# Reduce the distance range to 1500 km
plot(test, main="", xlim=c(0,1500))
abline(h=0)
```

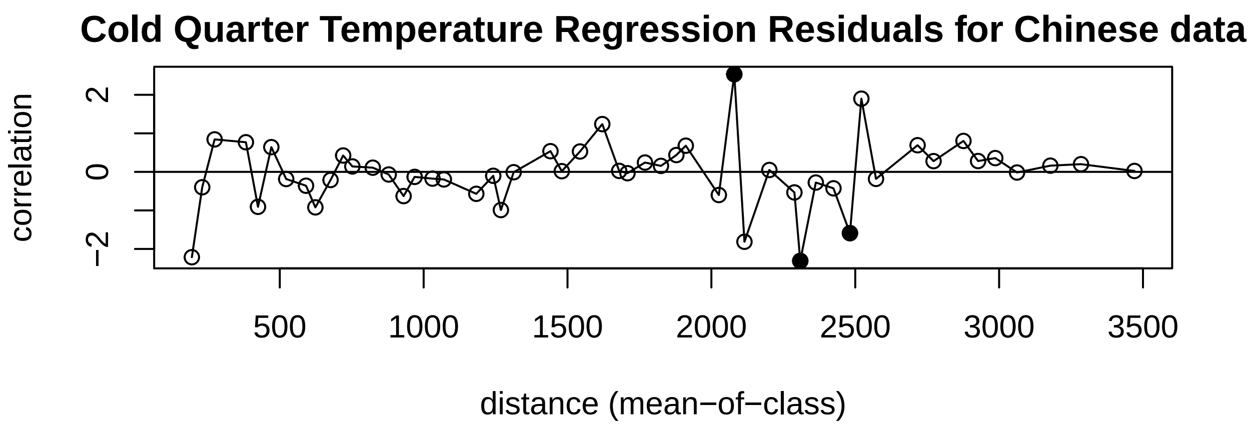


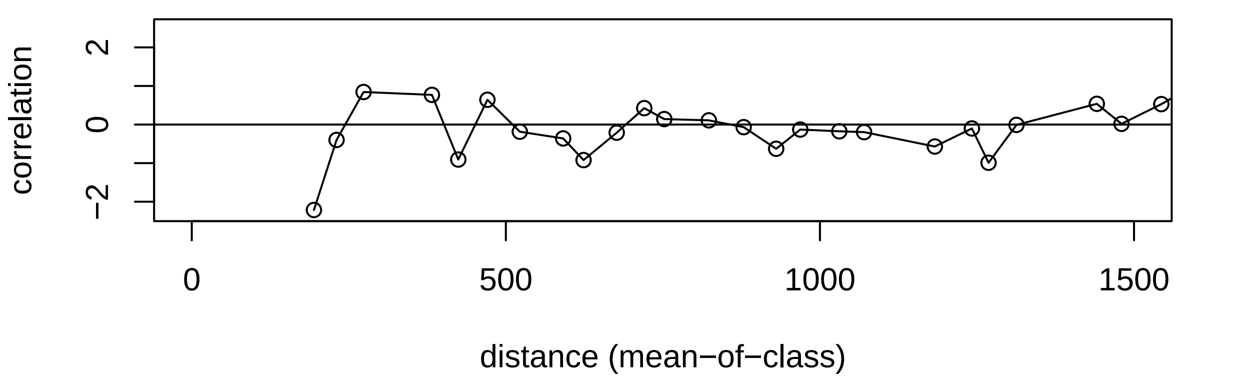



```
## Cold temperature
par(mfrow=c(2,1),mar=c(4,4,2,2))
# Use the coorelog function to develop the relationship
test <- correlog(black.china$longitude,
                 black.china$latitude, lm.cold.china$residuals,
                 increment=50, resamp=500, latlon=T)
```

```
## 50   of   500 100   of   500 150   of   500 200   of   500 250   of   500 300   of   500 350   of   500 400   of   5
```

```r
# Plot with the entire distance range
plot(test,
     main="Cold Quarter Temperature Regression Residuals for Chinese data")
abline(h=0)

# Reduce the distance range to 1500 km
plot(test, main="", xlim=c(0,1500))
abline(h=0)
```

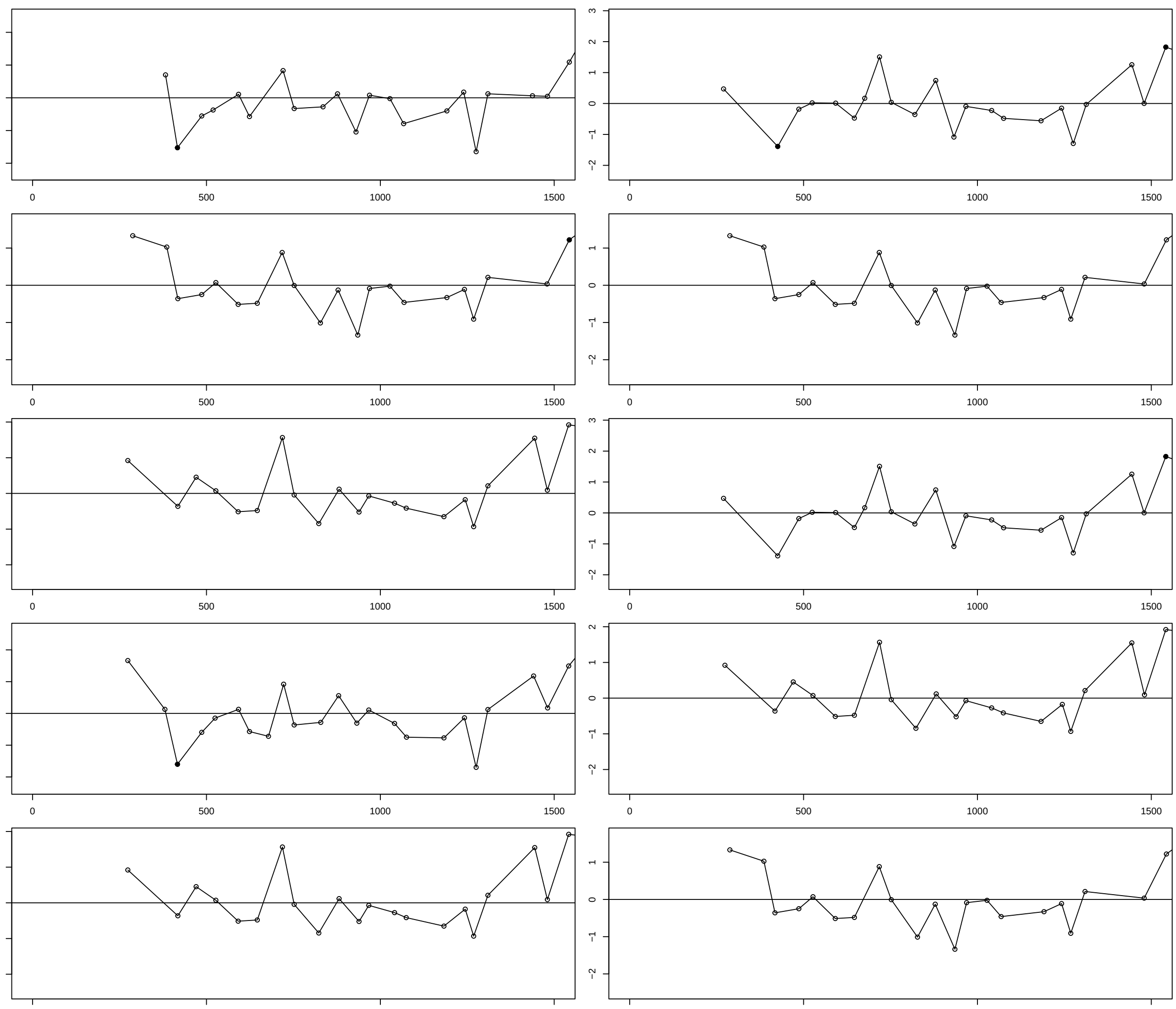




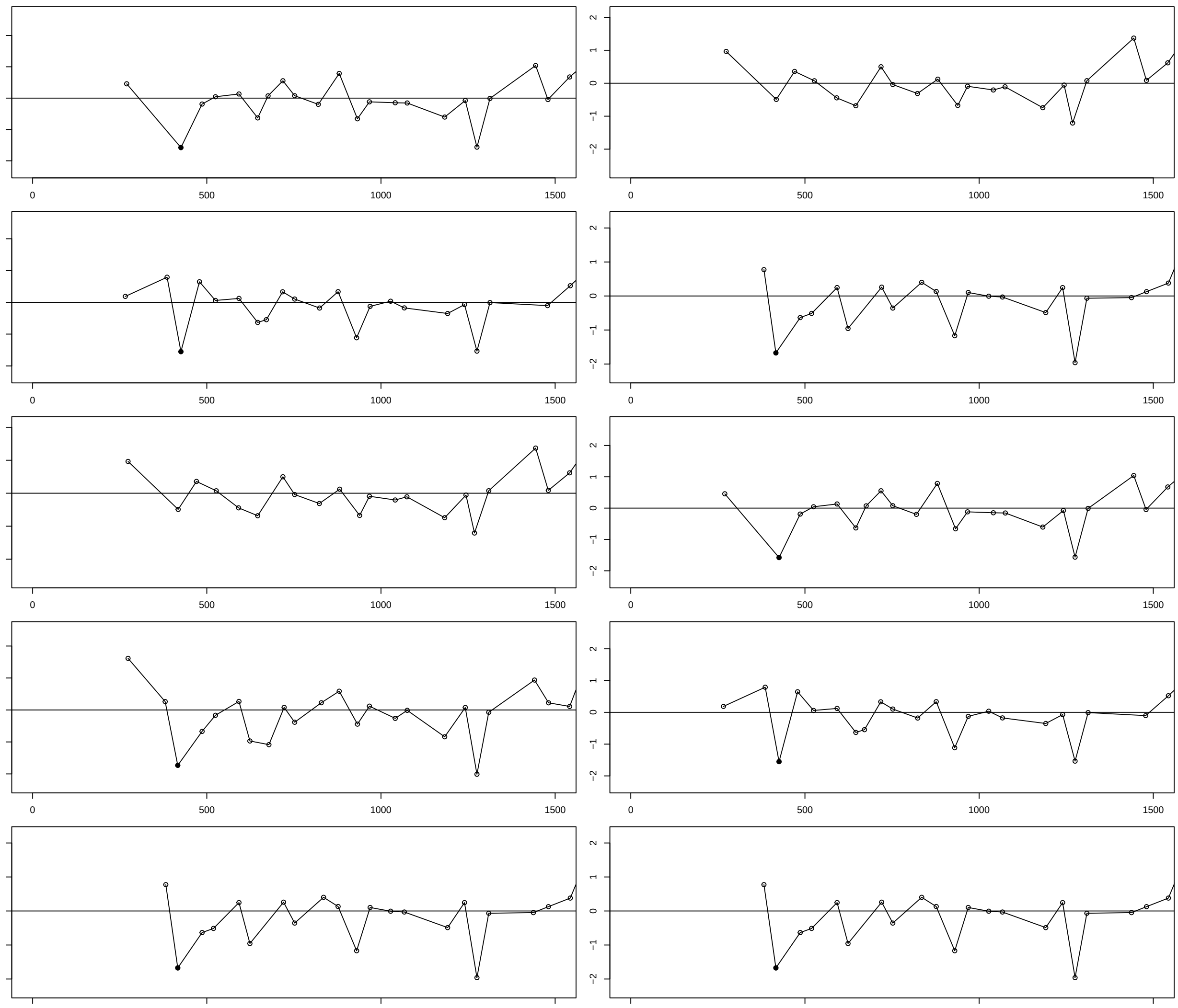
Subsample at 250km for Chinese dataset

Now we subsample at 250km to reduce spatial autocorrelation.

```r
table(black.china$spatial.code.250)

## Annual temperature
par(mfrow=c(5,2), mar=c(1,1,2,2))
for (i in 1:10){
  sub <- black.china %>% group_by(spatial.code.250) %>% sample_n(size=1)
  reg.sub.annual <- lm(log(sub$AAM)~sub$AnnualTemp)

  test <- correlog(sub$longitude, sub$latitude, reg.sub.annual$residuals,
                   increment=50, resamp=500, latlon=T)
  plot(test, main="", xlim=c(0,1500))
  abline(h=0)
}
```



```
## Cold temperature
par(mfrow=c(5,2), mar=c(1,1,2,2))
for (i in 1:10){
  sub <- black.china %>% group_by(spatial.code.250) %>% sample_n(size=1)
  reg.sub.cold <- lm(log(sub$AAM)~sub$ColdTemp)

  test <- correlog(sub$longitude, sub$latitude, reg.sub.cold$residuals,
                   increment=50, resamp=500, latlon=T)
  plot(test, main="", xlim=c(0,1500))
  abline(h=0)
}
```



Conclusions

- There was no global spatial autocorrelation in the entire dataset, but when looking at local clustering, we saw a strong negative autocorrelation at around 550 km.
- We tried to subsample at 250 km, and 550 km. 220 km did not reduce the spatial autocorrelation we saw. 550 km did reduce that, but left us with too few data to fit a model.
- We then tried to subsample by latitudinal straitification. We found that the negative autocorrelation was reduced in some iterations (when South Ukarine was not selected). We hypothesized that particular datapoint is the cause.
- We removed the SU data point and ran the local moran's I again. This time, we have no local clustering at 550 km!