Wind and pressure cross-variogram

These are preliminary results of empirical cross-variograms of wind and pressure data downloaded by Tommy for 5 days (each 12 hours) and a zone that birds would potentially explore. The idea is to start some discussions about interpretation and if it's necessary to go further and in which direction.

I'll copy all the codes but the notebook is not running them since it takes a lot of time. That is also the reason why I've only dealt with 10 time steps.

First, we call the packages and introduce the paths that I'll use (I already have them in my work directory):

```r
library(ncdf4)
library(raster)
library(tidyverse)
library(spacetime)
library(sp)
library(gstat)
library(ggplot2)

path_data <- './DataInputs/'
path_output_data <- './DataOutputs/'
path_plots <- './Plots/'
```

Then, I load the data in .nc files and transform them into one single data frame. It's kind of messy because I did it manually for each component (u and v for wind and sp for pressure) to make sure I was doing things right.

```r
###### TOMMY'S LINES

# meridional
v <- stack(brick(paste0(path_data,"wind.nc"), varname = "v10"))
# zonal
u <- stack(brick(paste0(path_data,"wind.nc"), varname = "u10"))

d.names <- names(v)
dates <- paste(substr(d.names, 2, 5), substr(d.names, 7, 8), substr(d.names, 10, 11), sep = "-")
times <- paste(substr(d.names, 13, 14), substr(d.names, 16, 17), substr(d.names, 19, 20), sep = ":")
date.times <- as.POSIXct(paste(dates, times, sep = " "), tz = "GMT")

# crop by max extent of tracks
ext <- extent(0, 90, -65, -29) # LIMITS PROVIDED BY TOMMY
v2 <- crop(v, ext)
u2 <- crop(u, ext)

sp <- stack(brick(paste0(path_data,"weather.nc"), varname = "sp"))
sp.names <- names(sp)
sp2 <- crop(sp, ext)

####### TRANSFORMING INTO DATA FRAMES

#### first with v2

raspt_v2 <- rasterToPoints(v2)
# Create a data frame showing layer name and time
```

```r
dt <- data_frame(Layer = names(v2), dttm = date.times)
# Transform the data
raspt2_v2 <- raspt_v2 %>%
  as_data_frame() %>%
  rename(lon = x, lat = y) %>%
  gather(Layer, v2, -lon, -lat) %>%
  left_join(dt, by = "Layer") %>%
  dplyr::select(lon, lat, dttm, v2)

raspt3_v2 <- raspt2_v2 %>%
  filter(dttm <"2013-01-30") # I'm keeping just one month of data

#### same thing with u2

raspt_u2 <- rasterToPoints(u2)

# Create a data frame showing layer name and time
dt <- data_frame(Layer = names(u2), dttm = date.times)

# Transform the data
raspt2_u2 <- raspt_u2 %>%
  as_data_frame() %>%
  rename(lon = x, lat = y) %>%
  gather(Layer, u2, -lon, -lat) %>%
  left_join(dt, by = "Layer") %>%
  dplyr::select(lon, lat, dttm, u2)

raspt3_u2 <- raspt2_u2 %>%
  filter(dttm <"2013-01-30")

raspt_wind <- left_join(raspt3_v2,raspt3_u2) # joining both v2 and u2 in one data frame

#### now it's sp

raspt_sp2 <- rasterToPoints(sp2)

# Create a data frame showing layer name and time
dt <- data_frame(Layer = names(sp2), dttm = date.times)

# Transform the data
raspt2_sp2 <- raspt_sp2 %>%
  as_data_frame() %>%
  rename(lon = x, lat = y) %>%
  gather(Layer, sp2, -lon, -lat) %>%
  left_join(dt, by = "Layer") %>%
  dplyr::select(lon, lat, dttm, sp2)

raspt3_sp2 <- raspt2_sp2 %>%
  filter(dttm <"2013-01-30")

raspt_wind_sp <- left_join(raspt_wind,raspt3_sp2)
raspt_wind_sp$WindSp <- sqrt(raspt_wind_sp$u2^2 + raspt_wind_sp$v2^2)
saveRDS(raspt_wind_sp,file=paste0(path_output_data,"transformed_data.rds")) # I'm saving this in an obj
```

```
rm(list=setdiff(ls(), "raspt_wind_sp")) # I remove everything else from the environment
```

Now, I proceed with the spatial analysis. To save time, I'll just say that I did some time series plots and saw that the temporal autocorrelation varies a lot from location to location.

Because the initial plan was to perform spatiotemporal analyses, I used the spacetime package, which provides us with data classes that allow for time series, spatial and spatiotemporal analyses (it's compatible with some packages that allow for those analyses). There is more info about the data classes in spacetime in the vignette from Pebesma about the package.

```
STFDF_day <- stConstruct(raspt_wind_sp,space=c('lon','lat'),time='dttm',
                        SpatialObj=SpatialPoints(raspt_wind_sp[,c('lon','lat')]))
is(STFDF_day) # Construct as IRREGULAR => 92.7 Mb!
STFDF_day    <- as(STFDF_day, "STFDF")
is(STFDF_day) # Transformed into Spatio Temporal FULL LATTICE Data Frame=> 65.4  Mb only!

# Adding projections
proj4string(STFDF_day) <- "+proj=longlat +ellps=WGS84 +datum=WGS84 +no_defs"
```

OK, for this part I decided to use only 5 days of data because the analyses take time and I wanted something I could show you quickly. Also, I am running other stuff in my laptop at the same time.

Quick explanation of variograms:

If a variable is spatially autocorrelated, then it's value at a given location will be similar to its value at a close location. If the process is stationary, then this relationship does not depend on the exact location we are looking at, but rather exclusively on the distance between locations. So when locations are further away, the values correlated less. If - for this spatially autocorrelated variable - we compare the values of several points separated by small distances in space, then the variance of those values will be smaller than the variance of points separated by large distances. A variogram quantifies the increment in variance between two points, separated by a distance h, when h increases. Specifically, it's:

$\hat{\gamma}(h) = \frac{1}{2n(h)} \sum_{x_i - x_j = h} (Z(x_i) - Z(x_j))^2$

I'm computing only empirical variograms, i.e. calculated from the data. No models are used because this is really exploratory.

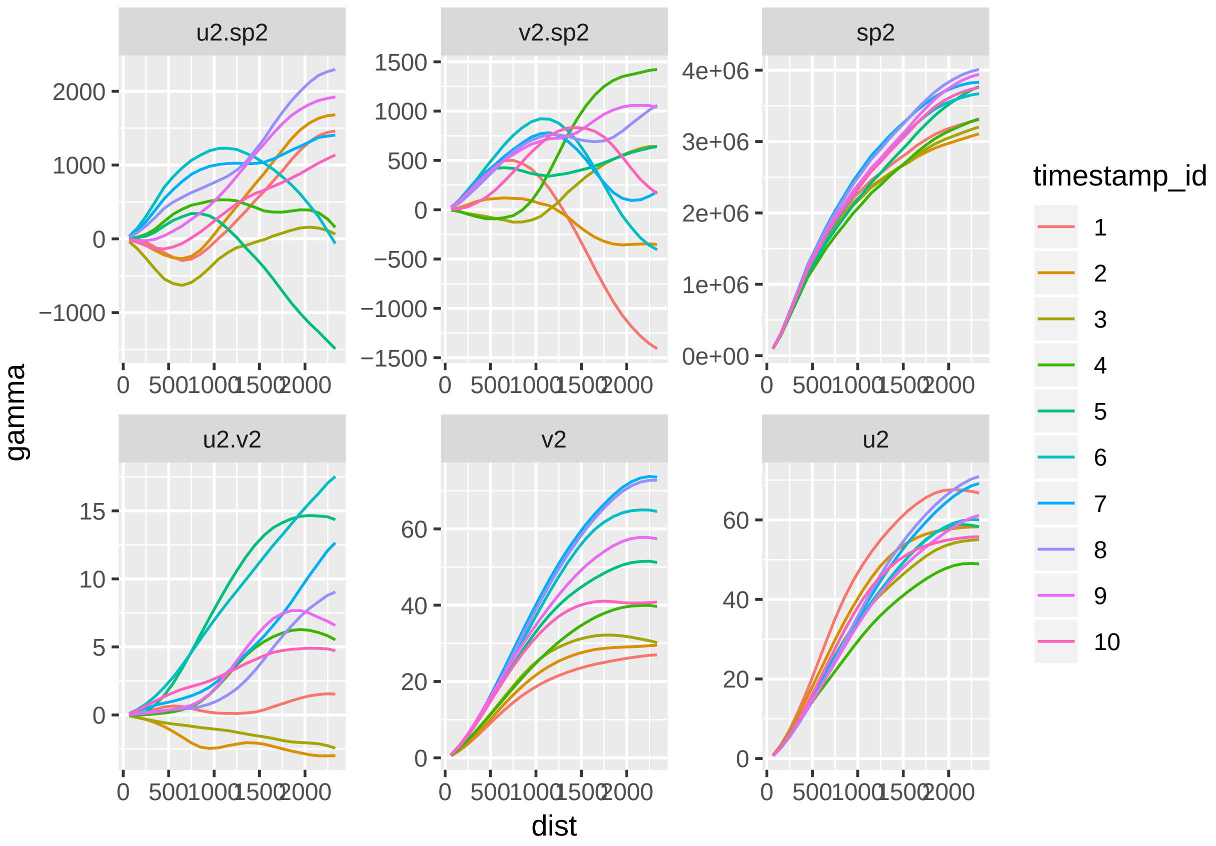
What we expect is a curve showing that the variogram function is increasing with distance, until certain point, showing that the distance is large enough that at that point there is no autocorrelation any more (distances larger than that don't make a difference in the values). I hope I'm explaining myself correctly.

I'm computing variograms for each variable (u, v and sp) and paired cross-variograms, which show how they covariate in space. I'm running an independence analysis for each timestamp (10 timestamps, 5 days), expecting to find common patterns.

Anyway, here is how you can do it in R:

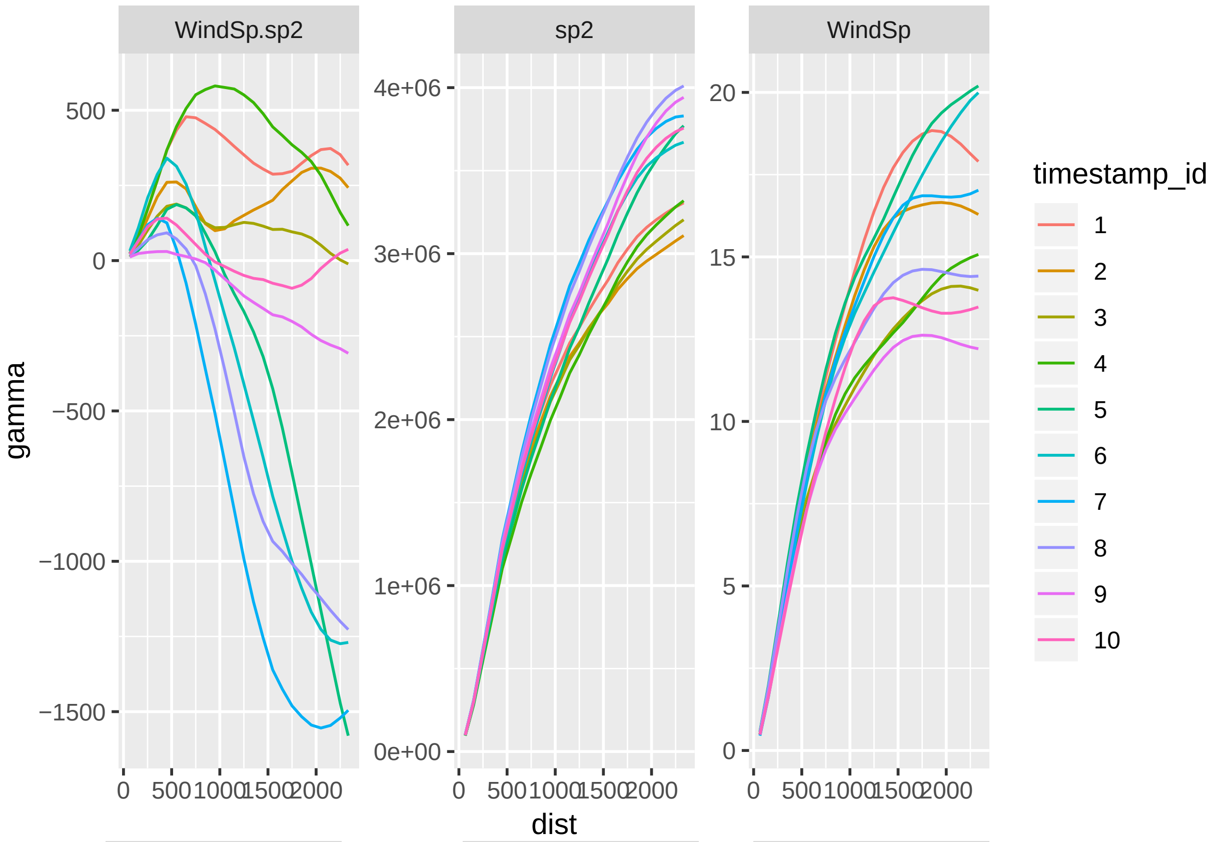
```
time_stamp <- STFDF_day@endTime
num_stamps <- 10 # 5 days (10 observations)
n_h <- 100 # distance h

elem <- matrix(NA,n_h,3)
cross_vario_list <- rep(list(elem),num_stamps)

for (i in 1:num_stamps){
sel <- STFDF_day[,time_stamp[i],c("WindSp","sp2")]
g <- gstat(NULL, "WindSp", WindSp ~ 1, data = sel[!is.na(sel[,"WindSp"]$WindSp),"WindSp"])
# g <- gstat(g, "v2", v2 ~ 1, data = sel[!is.na(sel[,"v2"]$v2),"v2"])
g <- gstat(g, "sp2", sp2 ~ lon + lat, data = sel[!is.na(sel[,"sp2"]$sp2),"sp2"]) # from previous analys
vm <- variogram(g,width=n_h)
cross_vario_list[[i]] <- data.frame(dist=vm$dist,gamma=vm$gamma,id=vm$id)
}

cross_df <- do.call(rbind.data.frame,cross_vario_list)
cross_df$timestamp_id <- as.factor(rep(1:num_stamps,each=24*3))  # 24 is a consequence of n_h 100km
save(cross_df,file=paste0(path_output_data,"cross_vario_df_windsp.rds"))

ggplot(data = cross_df, mapping = aes(x = dist, y = gamma, color = timestamp_id)) +
  geom_line() +
  facet_wrap(~ id, scales = "free")
ggsave(paste0(path_plots,"cross_vario_windsp.pdf"), width = 20, height = 20, units = "cm")
```

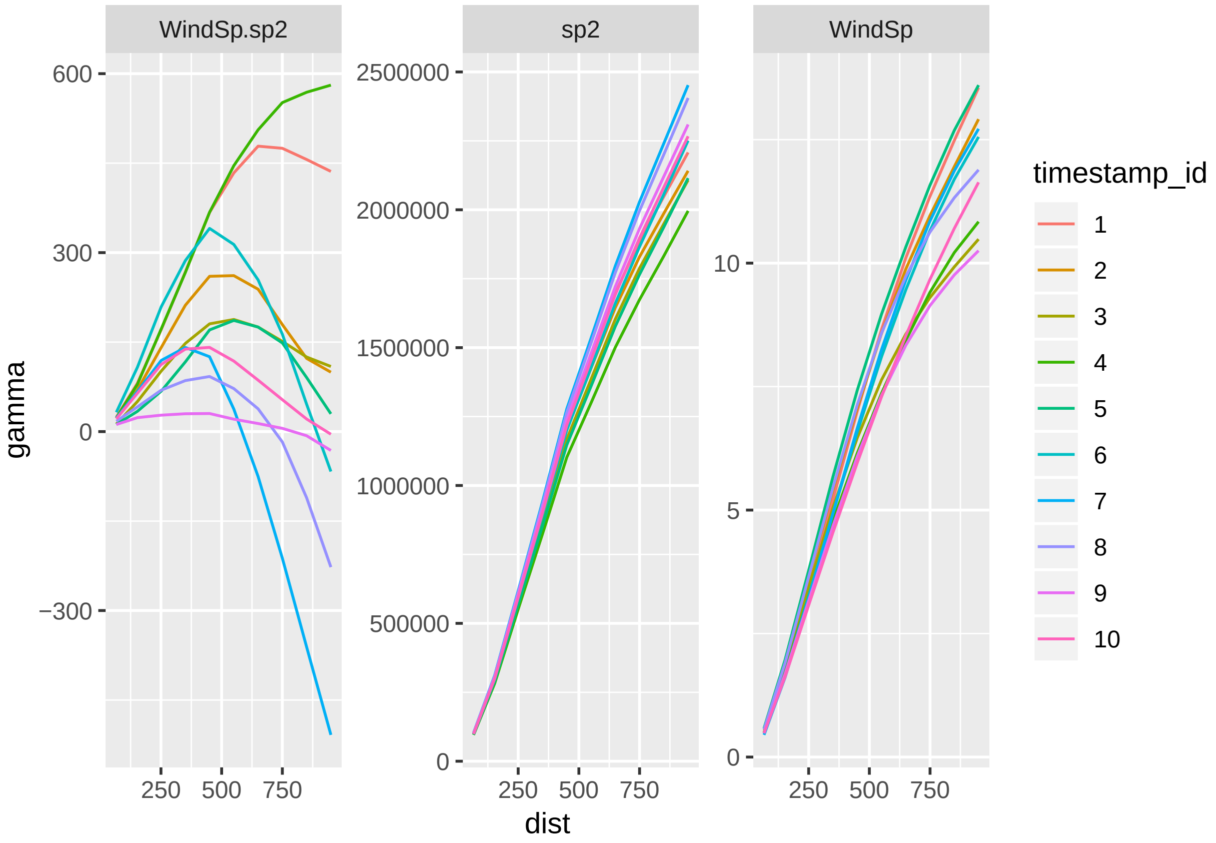
And here is the graphic output:



X: Distance in km, Y: variogram values. Time stamps are the daytimes of January of 2013 (i.e. 1: first day at 00:00, 2: Jan 1 12:00, and so on). Let's focus on the cross-variograms. They vary from datetime to datetime. Don't worry about the number of observations. There is like more than a million observations per point in the empirical variogram. In some datetimes there seems to be more spatial covariation than at others. Ranges differ, and in some cases there is even negative covariation. What is going on? How can we interpret it? Does it make sense? What should we expect to get if I include a larger time series?

And here is the graphic output:





Not the same pattern for every timestamp.