Statistics for High Dimensional Data (S4HDD) and CompStat Lab
a.a. 2023/2024 (2nd edition)

Prof. Francesco Finazzi francesco.finazzi@unibg.it

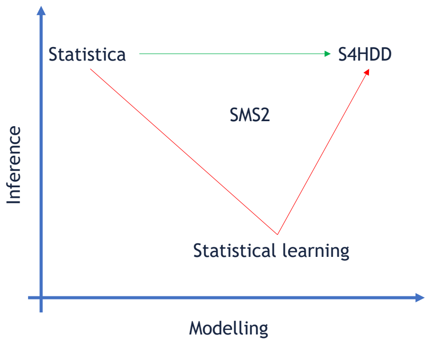
Prof. Alessandro Fassò alessandro.fasso@unibg.it

Course structure

- **9 CFUs**
  - 3 CFUs on time series modelling (ex SMS2) (Prof. Fassò)
  - 6 CFUs on space and space-time data modelling (Prof. Finazzi)
  - Space-time modelling will be addressed late in the course
- **Lectures**
  - Theory
  - Coding (MATLAB)
  - Analysis of data sets
  - Scientific article discussion

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Roadmap



UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Why this course

- To provide advanced statistical models/methods/tools that enable the extraction of useful information from temporal, spatial and spatio-temporal data sets (possibly large)…
- …in order to make decisions…
- …following a statistical inference approach…
- …thus, knowing the risk to make the wrong decision.

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Prerequisites

- Statistical inference
  - Distributions
  - Estimators
  - Confidence intervals
  - Hypothesis testing
- Simple and multiple regression
- Validation and cross-validation techniques
- Monte Carlo and bootstrap techniques
- Calcolo numeric (6 CFU) Prof. Maggioni

Textbooks

- Time series analysis and its applications: with R examples, Robert H. Shumway, David S. Stoffer
- Model-based Geostatistics, Peter J. Diggle, Paulo J. Ribeiro
- Statistics for spatio-temporal data, Noel A. Cressie, Christopher K. Wikle
- Spatio-temporal statistics with R, Christopher K. Wikle, Andrew Zammit-Mangion, Noel Cressie

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Student evaluation

- Teamwork - Groups of 1 or 2 students(?)
- You choose the data set to analyze
- The analysis is based on the models/methods seen during the course
- You can use R, MATLAB or Python
- The evaluation will be based on a report
- Each group will present in front of the class (in English)
- More details during the course…

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Software

- MATLAB, R (and Python) are the most common environments used by researchers in space-time data analysis.
- Software for space-time data analysis is often open source (and not fully tested - use at your own risk!)
- In this course:
  - R is used for time series data analysis
  - MATLAB is used for space and space-time data analysis
- R and MATLAB are not fully interchangeable when it comes to “complex” statistical models and methods
- D-STEM (v2) will be the MATLAB software for space-time data analysis (download from Journal of Statistical Software)

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Examples of spatio-temporal data sets

- COVID19 pandemic data sets (hospitalizations and casualties by country over time)
- Climate change data sets (satellite measurements of temperature, humidity, etc.)
- Air quality data sets (PM_{10}, $PM_{2.5}$, NOx etc. observed at monitoring stations)
- Smartphone-based real time detection of earthquakes (www.sismo.app)
- Internet data analytics (app/web visits by country/region over time)
- People mobility data (www.facebook.com/covid19mobility)
- Digital image/video analysis (tracking, object recognition, etc.)
- Ecology (point processes)
- In general, addressing interesting societal problems requires the analysis of space-time data sets

Why modelling spatio-temporal data

- For understanding the underlying data generation process
- For temporal, spatial and spatio-temporal prediction
  - Methods differ for continuous/discrete space and/or time
- For emulating complex phenomena (what is a model?)
- For making decisions

Modelling steps

- Model proposal (which equation, which covariates, which random variables, which relations between variables)
- Model estimation (maximum likelihood, expectation-maximization algorithm)
- Model validation (k-fold cross-validation, leave-one-out validation)
- Model adoption (offline or online)
- Model update/improvement (when/if new data become available)

A first spatial model

$$y(\boldsymbol{s}) = \boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta} + \varepsilon(\boldsymbol{s}) \quad (1)$$

- $y(\boldsymbol{s})$ is the observation at generic spatial location $\boldsymbol{s} \in \boldsymbol{R}^2$ or $\boldsymbol{s} \in \boldsymbol{S}^2$
- $\boldsymbol{x}(\boldsymbol{s})$ are spatial covariates (no missing data)
- $\boldsymbol{\beta}$ is a vector of parameters
- $\varepsilon(\boldsymbol{s})$ is the random error at spatial location $\boldsymbol{s}$ (e.g., $\varepsilon(\boldsymbol{s}) \sim NID(0, \sigma_\varepsilon^2)$)
- $\boldsymbol{R}^2$ is the 2D space (plane) while $\boldsymbol{S}^2$ is the sphere embedded in $\boldsymbol{R}^3$

Model complexity

- When dealing with spatio-temporal data sets model complexity is often an issue
- Elements of model complexity
  - Univariate, bivariate, multivariate models
  - Number of covariates (p>n problem)
  - Number of spatial and/or temporal points (big n problem)
  - Dependences among variables
  - Space-time (non)separability
  - Latent variables/processes
  - Non Gaussianity

How to estimate the spatial model?

- As usual we need data
- For us a data set has this form:

$$\{(\boldsymbol{x}_1(\boldsymbol{s}_1), y_1(\boldsymbol{s}_1)), \dots, (\boldsymbol{x}_n(\boldsymbol{s}_n), y_n(\boldsymbol{s}_n))\}$$

$$\boldsymbol{x}_i = \left(x_{i1}, \dots, x_{ip}\right)', i = 1, \dots, n$$

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Model residuals

$$e(\boldsymbol{s}) = y(\boldsymbol{s}) - \boldsymbol{x}(\boldsymbol{s})'\widehat{\boldsymbol{\beta}}$$
$$e(\boldsymbol{s}) = y(\boldsymbol{s}) - \hat{y}(\boldsymbol{s})$$

- We check if residuals $e(\boldsymbol{s})$ are IID which means:
  - Are residuls spatially uncorrelated?
  - Is the residul variance constant in space? (Homoscedasticity)

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Towards more complex models

- In general, residuals $e(\boldsymbol{s})$ are spatially correlated
- Why? Because $\boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta}$ cannot entirely capture the data variability
- This is similar to the case of serially/temporally correlated residuals in classic regression models ($\boldsymbol{x}'\boldsymbol{\beta}$)
- When this happens we might
  - Add covariates
  - Add transformations of covariates (polynomials)
  - Add interactions between covariates
- This may not be enough to have IID residuals

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Towards more complex models

- What happens if we ignore the spatial correlation of residuals?
  - The model fitting capability is lower than it should be
  - Variance of estimators is wrong
  - Confidence intervals might have significance levels lower than their nominal levels
  - Spatial predictions are poor (especially in model validation)

Spatial model with latent variable

$$y(\boldsymbol{s}) = \boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta} + \alpha w(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \quad (2)$$

- $y(\boldsymbol{s})$, $\boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta}$ and $\varepsilon(\boldsymbol{s})$ are the usual terms of a regression model (see model 1 in previous slides)
- $w(\boldsymbol{s})$ is a latent (non observed) random variable spatially correlated with unitary variance
- $corr\big(w(\boldsymbol{s}), w(\boldsymbol{s}')\big) = \rho(\boldsymbol{s}, \boldsymbol{s}'; \boldsymbol{\theta})$
- $\rho(\boldsymbol{s}, \boldsymbol{s}'; \boldsymbol{\theta})$ is a bivariate function (on $\boldsymbol{s}$ and $\boldsymbol{s}'$) with unknown parameter vector $\boldsymbol{\theta}$
- α is a scale parameter to be estimated

Spatial model with latent variable

$$y(\boldsymbol{s}) = \boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta} + \alpha w(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \quad (2)$$

- $w(\boldsymbol{s}) \perp \varepsilon(\boldsymbol{s})$, or $\mathrm{cov}\big(w(\boldsymbol{s}), \varepsilon(\boldsymbol{s})\big) = 0$
- Conditionally on $w(\boldsymbol{s})$, observed data $y_i(\boldsymbol{s}_i)$ are realizations of mutually independent random variables Y_i with conditional mean $\mathrm{E}(Y_i|w) = \boldsymbol{x}_i(\boldsymbol{s}_i)'\boldsymbol{\beta} + \alpha w(\boldsymbol{s}_i)$ and conditional variance σ_ε^2
- What is the difference between $\boldsymbol{s}_i$ and $\boldsymbol{s}$?

Spatial model with latent variable

$$y(\boldsymbol{s}) = \boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta} + \alpha w(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \quad (2)$$

- $w(\boldsymbol{s})$ is spatially varying (in general each location $\boldsymbol{s}$ has a different $w(\boldsymbol{s})$ value)
- Is $\varepsilon(\boldsymbol{s}) \equiv 0$ (and $e(\boldsymbol{s}) = 0$) if $w(\boldsymbol{s})$ can be choosen to explain $y(\boldsymbol{s})$ at each spatial location?
- How overfitting is avoided?
- What is the role of spatial correlation?

Spatial correlation function

- Correlation function $\rho(\boldsymbol{s}, \boldsymbol{s}'; \boldsymbol{\theta})$ can be any postive-definite function
- This condition imposes that the linear combination $\sum_{i=1}^{m} a_i w(\boldsymbol{s}_i)$ has positive variance
- $\boldsymbol{\theta}$ is a vactor of unknown parameters (can be of any size)

Examples of spatial correlation functions

- $\rho(\boldsymbol{s}, \boldsymbol{s}'; \boldsymbol{\theta})$ depends on spatial coordinates $\boldsymbol{s}$ and $\boldsymbol{s}'$
- Such a function describe the most «flexible» spatial correlation
- In 1D, the correlation between $w(0)$ and $w(1)$ would be different from the correlation between $w(1)$ and $w(2)$
- In most cases, $\rho(\boldsymbol{s}, \boldsymbol{s}'; \boldsymbol{\theta}) = \rho(\|\boldsymbol{s} - \boldsymbol{s}'\|; \boldsymbol{\theta}) = \rho(u; \boldsymbol{\theta})$ where $\| \quad \|$ is the distance (Euclidean or geodetic) between $\boldsymbol{s}$ and $\boldsymbol{s}'$.
- In this case the spatial correlation only depends on the distance u and not on the coordinates.

Exponential spatial correlation function

- Exponential spatial correlation function

$$\rho(u; \theta) = exp(-u/\theta)$$

- $\theta > 0$ is a scalar value
- Simple but not flexible
- Not always suitable to model spatial correlation

Matérn spatial correlation function

- Matérn correlation function:

$$\rho(u; \theta) = \{2^{\kappa-1}\Gamma(\kappa)\}^{-1}(u/\phi)K_\kappa(u/\phi)$$

- K_κ is the modified Bessel function of the second kind of order κ
- The parameter $\kappa > 0$ is called the order of the Matérn and determines the differentiability of $w(\boldsymbol{s})$ (more details later)
- The parameter $\phi > 0$ determines the rate at which the correlation decays to zero with increasing u

Gaussian process

- $w(\boldsymbol{s})$ is not directly observed but can be inferred from the observed data set $\{(\boldsymbol{x}_1(\boldsymbol{s}_1), y_1(\boldsymbol{s}_1)), \dots, (\boldsymbol{x}_n(\boldsymbol{s}_n), y_n(\boldsymbol{s}_n))\}$
- First, the distribution of $w(\boldsymbol{s})$ must be specified
- Formally, $w(\boldsymbol{s})$ is a zero mean Gaussian process (GP) with unitary variance and spatial correlation function $\rho(u; \boldsymbol{\theta})$
- $w(\boldsymbol{s})$ is a gaussian process if the joint distribution of $w(\boldsymbol{s}_1), \dots, w(\boldsymbol{s}_n)$ is multivariate normal (see SMS2) for any integer n and any set of locations $\boldsymbol{s}_1, \dots, \boldsymbol{s}_n$

Simulations

- How to simulate a GP with a given spatial correlation function in MATLAB?
- Simulations are useful to understand if a given correlation function is suitable to model the observed data set
- A GP is continuous in space. This means that:
  - It can be simulated on a regular grid or
  - On a irregular grid
  - For any given number of spatial locations
- When the GP is simulated on a regular grid, the simulated values referes to the centres of the pixels

Statistics for High Dimensional Data (and CompStat Lab)
a.a. 2023/2024 (2nd edition)

Prof. Francesco Finazzi francesco.finazzi@unibg.it
Prof. Alessandro Fassò alessandro.fasso@unibg.it

Lesson 2

Spatial model with latent variable

$$y(\boldsymbol{s}) = \boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta} + \alpha w(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \quad (2)$$

- $w(\boldsymbol{s}) \sim GP\big(0, \rho(\|\boldsymbol{s} - \boldsymbol{s}'\|; \boldsymbol{\theta})\big)$
- $\rho(\|\boldsymbol{s} - \boldsymbol{s}'\|; \boldsymbol{\theta}) = corr\big(w(\boldsymbol{s}), w(\boldsymbol{s}')\big)$
- $\varepsilon(\boldsymbol{s}) \sim N(0, \sigma_\varepsilon^2)$
- The unknown parameter set is $\Psi = \{\boldsymbol{\beta}, \alpha, \sigma_\varepsilon^2, \boldsymbol{\theta}\}$
- How to estimate Ψ from data?

Maximum likelihook estimate

- We rely on MLE to estimate the model parameter vector
- Because MLE has good properties and…
- Because we can use the EM algorithm (if needed)
- Likelihood function of (2) is:

$$L(\Psi; \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{X}) = L(\Psi; \boldsymbol{y}|\boldsymbol{w}, \boldsymbol{X})L(\Psi; \boldsymbol{w})$$

- $\boldsymbol{y} = (y_1, \dots, y_n)'$ is the vector of observations at n spatial locations
- $\boldsymbol{w} = (w_1, \dots, w_n)'$ is the vector of latent variables at n spatial locations
- $\boldsymbol{X}$ is the $n \times p$ design matrix

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Likelihood decomposition

$$\begin{aligned} L(\Psi; \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{X}) &= L(\Psi; \boldsymbol{y}|\boldsymbol{w}, \boldsymbol{X})L(\Psi; \boldsymbol{w}) \\ &= L(\boldsymbol{\beta}, \alpha, \sigma_\varepsilon^2; \boldsymbol{y}|\boldsymbol{w}, \boldsymbol{X})L(\boldsymbol{\theta}; \boldsymbol{w}) \end{aligned}$$

- Each likelihood term depends on a subset of Ψ.
- $L(\Psi; \boldsymbol{y}, \boldsymbol{w}, \boldsymbol{X})$ is the complete-data likelihood (which assumes $\boldsymbol{w}$ to be known)
- $L(\boldsymbol{\beta}, \alpha, \sigma_\varepsilon^2; \boldsymbol{y}|\boldsymbol{w}, \boldsymbol{X})$ and $L(\boldsymbol{\theta}; \boldsymbol{w})$ are densities of n −variate normal distributions

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

Log-likelihood function

- As usual we prefer to work with $\log(L_\Psi)$
- $-2\log L_\Psi$ is given by:

$$log|\Sigma_\varepsilon| + \boldsymbol{e}'\Sigma_\varepsilon^{-1}\boldsymbol{e} + log\,|\Sigma_w| + \boldsymbol{w}'\Sigma_w^{-1}\boldsymbol{w}$$

where

- $\boldsymbol{e} = \boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta} - \alpha\boldsymbol{w}$
- $\Sigma_\varepsilon = \sigma_\varepsilon^2\boldsymbol{I}_n$, with $\boldsymbol{I}_n$ the identity matrix of dimension n
- Σ_w is the $n \times n$ correlation matrix (e.g., $exp(-D/\theta)$, with D the distance matrix)

UNIVERSITÀ DEGLI STUDI DI BERGAMO | Dipartimento di Ingegneria Gestionale, dell'Informazione e della Produzione

ML estimate

- MLE is given by

$$\hat{\Psi} = argmin_{\boldsymbol{\beta},\alpha,\sigma_\varepsilon^2,\boldsymbol{\theta}} \; log|\Sigma_\varepsilon| + \boldsymbol{e}'\Sigma_\varepsilon^{-1}\boldsymbol{e} + log\,|\Sigma_w| + \boldsymbol{w}'\Sigma_w^{-1}\boldsymbol{w}$$

- Argmin because we are considering $-2\log(L_\Psi)$
- Unfortunately minimizing $-2\log(L_\Psi)$ is not feasible, plus $\boldsymbol{w}$ is latent and not observed
- We must rely on the EM algorithm

EM algorithm

- The EM is an iterative algorithm for MLE
- First iteration starts with initial values $\hat{\Psi}^{\langle 0\rangle}$ (usually given by OLS and method of moments)
- E-step

$$Q\left(\Psi, \hat{\Psi}^{\langle m\rangle}\right) = E_{\hat{\Psi}^{\langle m\rangle}}(-2\log\mathrm{L}(\Psi;\boldsymbol{y},\boldsymbol{w},\boldsymbol{X})|\boldsymbol{y})$$

- M-step

$$\hat{\Psi}^{\langle m+1\rangle} = argmax_\Psi \; Q\left(\Psi, \hat{\Psi}^{\langle m\rangle}\right)$$

EM algorithm, E-step

- E-step

$$\begin{aligned} &E_{\hat{\Psi}^{\langle m\rangle}}(-2\log\mathrm{L}(\Psi;\boldsymbol{y},\boldsymbol{w},\boldsymbol{X})|\boldsymbol{y}) \\ &= tr\left[\Sigma_\varepsilon^{-1}\left(E(\boldsymbol{e}|\boldsymbol{y})E(\boldsymbol{e}|\boldsymbol{y})' + Var(\boldsymbol{e}|\boldsymbol{y})\right)\right] \\ &+ tr\left[\Sigma_w^{-1}\left(E(\boldsymbol{w}|\boldsymbol{y})E(\boldsymbol{w}|\boldsymbol{y})' + Var(\boldsymbol{w}|\boldsymbol{y})\right)\right] \end{aligned}$$

- $E(\boldsymbol{e}|\boldsymbol{y}) = \boldsymbol{y} - \boldsymbol{X\beta} - \alpha E(\boldsymbol{w}|\boldsymbol{y})$
- $Var(\boldsymbol{e}|\boldsymbol{y}) = Var(\boldsymbol{y} - \boldsymbol{X\beta} - \alpha\boldsymbol{w}|\boldsymbol{y}) = \alpha^2 Var(\boldsymbol{w}|\boldsymbol{y})$

EM algorithm, E-step

- $E(\boldsymbol{w}|\boldsymbol{y}) = Cov(\boldsymbol{w},\boldsymbol{y})Var(\boldsymbol{y})^{-1}[\boldsymbol{y} - \boldsymbol{X\beta}]$ (see multivariate normal)
- $Var(\boldsymbol{w}|\boldsymbol{y}) = \Sigma_w - Cov(\boldsymbol{w},\boldsymbol{y})Var(\boldsymbol{y})^{-1}Cov(\boldsymbol{w},\boldsymbol{y})'$
- $Var(\boldsymbol{y}) = Var(\boldsymbol{X\beta} + \alpha\boldsymbol{w} + \boldsymbol{\varepsilon}) = Var(\alpha\boldsymbol{w} + \boldsymbol{\varepsilon}) = \alpha^2 Var(\boldsymbol{w}) + Var(\boldsymbol{\varepsilon}) + 2\mathrm{Cov}(\boldsymbol{w},\boldsymbol{\varepsilon})$
- $Var(\boldsymbol{w}) = \Sigma_w$
- $Var(\boldsymbol{\varepsilon}) = \Sigma_\varepsilon = \sigma_\varepsilon^2\boldsymbol{I}_n$
- $2\mathrm{Cov}(\boldsymbol{w},\boldsymbol{\varepsilon}) = \boldsymbol{0}$ (from model assumptions)
- $Cov(\boldsymbol{w},\boldsymbol{y}) = Cov(\boldsymbol{w},\boldsymbol{X\beta} + \alpha\boldsymbol{w} + \boldsymbol{\varepsilon}) = Cov(\boldsymbol{w},\alpha\boldsymbol{w}) = \alpha Cov(\boldsymbol{w},\boldsymbol{w}) = \alpha Var(\boldsymbol{w}) = \alpha\Sigma_w$

EM algorithm, M-step

$$\widehat{\Psi}^{\langle m+1\rangle} = argmax_{\Psi}\, Q\left(\Psi, \widehat{\Psi}^{\langle m\rangle}\right)$$

$$\frac{dQ\left(\Psi, \widehat{\Psi}^{\langle m\rangle}\right)}{d\Psi} = 0$$

$$\alpha^{\langle m+1\rangle} = \frac{tr\left[\left(\boldsymbol{y} - \boldsymbol{X}\boldsymbol{\beta}^{\langle m\rangle}\right)E(\boldsymbol{w}|\boldsymbol{y})'\right]}{tr[E(\boldsymbol{w}|\boldsymbol{y})E(\boldsymbol{w}|\boldsymbol{y})' + Var(\boldsymbol{w}|\boldsymbol{y})]}$$

$$\boldsymbol{\beta}^{\langle m+1\rangle} = (\boldsymbol{X}'\boldsymbol{X})^{-1}\left[\boldsymbol{X}'\left(\boldsymbol{y} - \alpha^{\langle m+1\rangle}E(\boldsymbol{w}|\boldsymbol{y})\right)\right]$$

EM algorithm, M-step

$$\sigma_{\varepsilon}^{2\langle m+1\rangle} = \frac{1}{n} tr[E(\boldsymbol{e}|\boldsymbol{y})E(\boldsymbol{e}|\boldsymbol{y})' + Var(\boldsymbol{e}|\boldsymbol{y})]$$

$$\boldsymbol{\theta}^{\langle m+1\rangle} = argmin_{\boldsymbol{\theta}}\ \ log\,|\Sigma_w^{-1}(\boldsymbol{\theta})| + tr[\Sigma_w^{-1}(\boldsymbol{\theta})(\widehat{\boldsymbol{w}}\widehat{\boldsymbol{w}}')]$$

Where $\widehat{\boldsymbol{w}} = E_{\boldsymbol{\theta}^{\langle m\rangle}}(\boldsymbol{w}|\boldsymbol{y}) = E(\boldsymbol{w}|\boldsymbol{y})$

Statistics for High Dimensional Data (and CompStat Lab) a.a. 2023/2024 (2nd edition)

Prof. Francesco Finazzi francesco.finazzi@unibg.it
Prof. Alessandro Fassò alessandro.fasso@unibg.it

Lesson 3

Spatial model with latent variable

$$y(\boldsymbol{s}) = \boldsymbol{x}(\boldsymbol{s})'\boldsymbol{\beta} + \alpha w(\boldsymbol{s}) + \varepsilon(\boldsymbol{s}) \quad (2)$$

- $w(\boldsymbol{s}) \sim GP\left(0, \rho(\|\boldsymbol{s} - \boldsymbol{s}'\|; \boldsymbol{\theta})\right)$
- $\rho(\|\boldsymbol{s} - \boldsymbol{s}'\|; \boldsymbol{\theta}) = corr\left(w(\boldsymbol{s}), w(\boldsymbol{s}')\right)$
- $\varepsilon(\boldsymbol{s}) \sim N(0, \sigma_\varepsilon^2)$
- The unknown parameter set is $\Psi = \{\boldsymbol{\beta}, \alpha, \sigma_\varepsilon^2, \boldsymbol{\theta}\}$
- Ψ is estimated using the EM algorithm

Prediction using model

- Once estimated, the model is used for prediction:

$$\hat{y}(\boldsymbol{s}_i) = \boldsymbol{x}(\boldsymbol{s}_i)'\widehat{\boldsymbol{\beta}} + \hat{\alpha}\widehat{w}(\boldsymbol{s}_i)$$

- The above formula gives the prediction at spatial locations $\boldsymbol{s}_i, i = 1, \dots, n$
- $\widehat{w}(\boldsymbol{s}_i) = E(w(\boldsymbol{s}_i)|Y)$
- How to estimate y at a generic spatial location $\boldsymbol{s}$?
- (First of all we need $\boldsymbol{x}(\boldsymbol{s})$)

Spatial prediction

- First of all we need $\boldsymbol{x}(\boldsymbol{s})$ (spatial covariates at $\boldsymbol{s}$)
- Remember that $E(\boldsymbol{w}|\boldsymbol{y}) = Cov(\boldsymbol{w}, \boldsymbol{y})Var(\boldsymbol{y})^{-1}[\boldsymbol{y} - \boldsymbol{X\beta}]$
- Similarly $\widehat{w}(\boldsymbol{s}) = E(w(\boldsymbol{s})|\boldsymbol{y}) = Cov(w(\boldsymbol{s}), \boldsymbol{y})Var(\boldsymbol{y})^{-1}[\boldsymbol{y} - \boldsymbol{X\beta}]$
- In practice the prediction $\widehat{w}(\boldsymbol{s})$ depends on the vector $\boldsymbol{y}$ of all observations
- More in details: $Cov(w(\boldsymbol{s}), \boldsymbol{y}) = Cov(w(\boldsymbol{s}), \boldsymbol{X\beta} + \alpha\boldsymbol{w} + \boldsymbol{\varepsilon}) = Cov(w(\boldsymbol{s}), \alpha\boldsymbol{w}) = \alpha Cov(w(\boldsymbol{s}), \boldsymbol{w})$
- $Cov(w(\boldsymbol{s}), \boldsymbol{w})$ is a $1 \times n$ vector with elements $Cov\left(w(\boldsymbol{s}), w(\boldsymbol{s}_i)\right) = exp\left(-\frac{\|s-s_i\|}{\theta}\right)$.
- $Var(\boldsymbol{y})^{-1}[\boldsymbol{y} - \boldsymbol{X\beta}]$ are the same seen for the EM algorithm and do not depend on $w(\boldsymbol{s})$.

Spatial prediction

- When prediction is done for multiple spatial locations $\boldsymbol{S} = \{\boldsymbol{s}_1, \dots, \boldsymbol{s}_M\}$:

$$\widehat{\boldsymbol{w}}(\boldsymbol{S}) = E(\boldsymbol{w}(\boldsymbol{S})|\boldsymbol{y}) = Cov(\boldsymbol{w}(\boldsymbol{S}), \boldsymbol{y})Var(\boldsymbol{y})^{-1}[\boldsymbol{y} - \boldsymbol{X\beta}]$$

- $Cov(\boldsymbol{w}(\boldsymbol{S}), \alpha\boldsymbol{w}) = \alpha Cov(\boldsymbol{w}(\boldsymbol{S}), \boldsymbol{w})$
- $Cov(\boldsymbol{w}(\boldsymbol{S}), \boldsymbol{w})$ is a $M \times n$ matrix
- $Var(\boldsymbol{y})^{-1}[\boldsymbol{y} - \boldsymbol{X\beta}]$ is computed only one time!
- $Var(\boldsymbol{w}(\boldsymbol{S})|\boldsymbol{y})$ is the prediction uncertainty
- How to make spatial prediction with D-STEM?

Multivariate models

- It is not uncommon to jointly model multiple variables (e.g., multiple pollutants, multiple met. variables, etc.)
- The spatial model becomes multivariate

$$\boldsymbol{y}(\boldsymbol{s}) = \boldsymbol{X}(\boldsymbol{s})'\boldsymbol{\beta} + \boldsymbol{\alpha}\boldsymbol{w}(\boldsymbol{s}) + \boldsymbol{\varepsilon}(\boldsymbol{s}) \quad (2)$$

- $\boldsymbol{y}(\boldsymbol{s})$ is a $p \times 1$ vector
- $\boldsymbol{w}(\boldsymbol{s}) \sim GP_p(\boldsymbol{0}, \boldsymbol{V}\rho(\|\boldsymbol{s}-\boldsymbol{s}'\|; \boldsymbol{\theta}))$ is a p-variate Gaussian random process
- $\boldsymbol{V}$ is a $p \times p$ correlation matrix
- $\boldsymbol{\varepsilon}(\boldsymbol{s}) \sim N_p(\boldsymbol{0}, \boldsymbol{\Sigma}_\varepsilon^2)$ is a p-variate Normal random variable with $\boldsymbol{\Sigma}_\varepsilon^2$ diagonal
- The unknown parameter set is $\Psi = \{\boldsymbol{\beta}, \boldsymbol{\alpha}, \boldsymbol{\Sigma}_\varepsilon^2, \boldsymbol{\theta}, \boldsymbol{V}\}$

Why a multivariate model?

- Why not fitting a model for each variable?
- If two or more variables are correlated, spatial prediction can benefit from this correlation
- Especially if one variables is observed at few spatial locations w.r.t. the other variables
- But computing time is higher (matrices are $pn \times pn$ if all variables are observed at n locations)

Bivariate model

- A bivariate model is a (simple) special case of the multivariate model

$$\begin{bmatrix} y_1(\boldsymbol{s}) \\ y_2(\boldsymbol{s}) \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_1(\boldsymbol{s})' & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{x}_2(\boldsymbol{s})' \end{bmatrix} \begin{bmatrix} \boldsymbol{\beta}_1(\boldsymbol{s}) \\ \boldsymbol{\beta}_2(\boldsymbol{s}) \end{bmatrix} + \begin{bmatrix} \alpha_1 w_1(\boldsymbol{s}) \\ \alpha_2 w_2(\boldsymbol{s}) \end{bmatrix} + \begin{bmatrix} \varepsilon_1(\boldsymbol{s}) \\ \varepsilon_2(\boldsymbol{s}) \end{bmatrix}$$

- $\boldsymbol{x}_1(\boldsymbol{s})'$ and $\boldsymbol{x}_2(\boldsymbol{s})'$ can have different lengths
- $\boldsymbol{V} = \begin{bmatrix} 1 & corr(w_1(\boldsymbol{s}), w_2(\boldsymbol{s})) \\ corr(w_2(\boldsymbol{s}), w_1(\boldsymbol{s})) & 1 \end{bmatrix}$
- $\boldsymbol{\Sigma}_\varepsilon^2 = \begin{bmatrix} \sigma_{1,\varepsilon}^2 & 0 \\ 0 & \sigma_{2,\varepsilon}^2 \end{bmatrix}$

Linear model of coregionalization

- $\boldsymbol{V}\rho(\|\boldsymbol{s}-\boldsymbol{s}'\|; \boldsymbol{\theta})$ is called linear coregionalization model
- $\boldsymbol{\theta}$ is common to the p variables!
- This could be a limit because all the variables are forced to share the same spatial correlation (same function and strenght)
- $\boldsymbol{V}$ is a symmetric correlation matrix, only $p(p-1)/2$ elements of $\boldsymbol{V}$ are estimated

Data structure

- A multivatiate data set can be classified depending on spatial locations:
  - **Isotopic**: all p variables are observed at the same n spatial locations.
  - **Fully heterotipic**: the p variables do not share a single spatial location.
  - **Partially heterotopic**: some of the p variables are observed at a subset of the n spatial locations. This is the most common case and it is the case handled by D-STEM.

Statistics for High Dimensional Data (and CompStat Lab)
a.a. 2023/2024 (2st edition)

bitvector,bitvector

Prof. Francesco Finazzi francesco.finazzi@unibg.it
Prof. Alessandro Fassò alessandro.fasso@unibg.it

Lesson 4

Spatio-temporal model: the dynamic coregionalization model (DCM)

$$
\begin{aligned}
y(\boldsymbol{s},t) &= \boldsymbol{x}_{\boldsymbol{\beta}}(\boldsymbol{s},t)'\boldsymbol{\beta} + \boldsymbol{x}_{\boldsymbol{z}}(\boldsymbol{s})'\boldsymbol{z}(t) + \alpha w(\boldsymbol{s},t) + \varepsilon(\boldsymbol{s},t) \\
\boldsymbol{z}(t) &= \boldsymbol{G}\boldsymbol{z}(t-1) + \boldsymbol{\eta}(t)
\end{aligned}
$$

- $\boldsymbol{x}_{\boldsymbol{\beta}}(\boldsymbol{s},t)$ and $\boldsymbol{x}_{\boldsymbol{z}}(\boldsymbol{s})$ are vectors of covariates. Note that $\boldsymbol{x}_{\boldsymbol{z}}(\boldsymbol{s})$ is time invariant.
- $w(\boldsymbol{s},t) \sim GP\left(0, \rho(\|\boldsymbol{s}-\boldsymbol{s}'\|; \boldsymbol{\theta})\right)$ is correlated over space but IID over time
- $\boldsymbol{z}(t)$ is $q \times 1$ dimensional with Markovian dynamics
- $\boldsymbol{G}$ is a stable $q \times q$ transition matrix
- $\boldsymbol{\eta}(t) \sim N(\boldsymbol{0}, \boldsymbol{\Sigma}_{\eta})$ is the innovation with $\boldsymbol{\Sigma}_{\eta}$ the variance-covariance matrix
- $\varepsilon(\boldsymbol{s},t) \sim N(0, \sigma_{\varepsilon}^2)$ is the measurement error

Why the Markovian dynamics?

- The Markovian dynamic helps to describe the temporal persistence which usually characterizes temporal phenomena.
- For instance: even if we stop air pollution emissions, pollutant concentration will not drop instantly to zero.
  - Emission would be a model covariate

Data matrix

- $\boldsymbol{Y} = (\boldsymbol{y}_1, \dots, \boldsymbol{y}_T)$ is the data matrix
- Each $\boldsymbol{y}_t$ is the vector of spatial observation at time $t = 1, \dots, T$
- In each $\boldsymbol{y}_t$, missing data are possible

- Covariates are in a data array $\boldsymbol{X} = \{\boldsymbol{X}_1, \dots, \boldsymbol{X}_T\}$, each $\boldsymbol{X}_t$ is a $n \times b$ matrix where b is the number of covariates. $\boldsymbol{X}$ cannot have missing data.

Likelihood function

- The parameter vector is $\Psi = \{\boldsymbol{\beta}, \alpha, \sigma_\varepsilon^2, \boldsymbol{\theta}, \boldsymbol{G}, \boldsymbol{\Sigma}_\eta\}$
- Likelihood function is:

$$L(\Psi; \boldsymbol{Y}, \boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{X}) = L(\Psi; \boldsymbol{Y} | \boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{X}) L(\Psi; \boldsymbol{Z}) L(\Psi; \boldsymbol{W})$$

- $\boldsymbol{W} = (\boldsymbol{w}_1, \dots, \boldsymbol{w}_T)$
- $\boldsymbol{Z} = (\boldsymbol{z}_1, \dots, \boldsymbol{z}_T)$

Log-likelihood function

- $-2\log L_\Psi$ is given by:

$$T log|\boldsymbol{\Sigma}_\varepsilon| + \sum_{t=1}^{T} \boldsymbol{e}_t{}' \boldsymbol{\Sigma}_\varepsilon^{-1} \boldsymbol{e}_t + T log|\boldsymbol{\Sigma}_\eta| + \sum_{t=1}^{T} (\boldsymbol{z}_t - \boldsymbol{G}\boldsymbol{z}_{t-1})' \boldsymbol{\Sigma}_\eta^{-1} (\boldsymbol{z}_t - \boldsymbol{G}\boldsymbol{z}_{t-1}) + T log\, |\boldsymbol{\Sigma}_w| + \sum_{t=1}^{T} \boldsymbol{w}_t{}' \boldsymbol{\Sigma}_w^{-1} \boldsymbol{w}_t$$

where

- $\boldsymbol{e}_t = \boldsymbol{y}_t - \boldsymbol{X}_{\beta,t}\boldsymbol{\beta} - \boldsymbol{X}_{z,t}\boldsymbol{z}_t - \alpha \boldsymbol{w}_t$
- $\boldsymbol{\Sigma}_\varepsilon = \sigma_\varepsilon^2 \boldsymbol{I}_n$, with $\boldsymbol{I}_n$ the identity matrix of dimension n
- $\boldsymbol{\Sigma}_w$ is the $n \times n$ correlation matrix (e.g., $exp(-D/\theta)$, with D the distance matrix)

Model estimation – EM algorithm

- Model estimation is (again) based on the EM algorithm
- $E(w(\boldsymbol{s},t)|\boldsymbol{Y})$ and $Var(w(\boldsymbol{s},t)|\boldsymbol{Y})$ are given by the same formulas of $E(w(\boldsymbol{s})|\boldsymbol{Y})$ and $Var(w(\boldsymbol{s})|\boldsymbol{Y})$ (because $w(\boldsymbol{s},t)$ are IID over time)
- $E(\boldsymbol{z}(t)|\boldsymbol{Y})$ and $Var(\boldsymbol{z}(t)|\boldsymbol{Y})$ are given by the Kalman smoother
- However, $Cov(\boldsymbol{z}(t), w(\boldsymbol{s},t)|\boldsymbol{Y}) \neq \boldsymbol{0}$. E-step and M-step are more complicated than the spatial case.

Statistics for High Dimensional Data (and CompStat Lab)
a.a. 2023/2024 (2nd edition)

Prof. Francesco Finazzi francesco.finazzi@unibg.it
Prof. Alessandro Fassò alessandro.fasso@unibg.it

Lesson 5

Spatio-temporal model: the (univariate) hidden dynamic geostatistical model (HDGM)

$$\begin{aligned} y(\boldsymbol{s},t) &= \boldsymbol{x}_{\boldsymbol{\beta}}(\boldsymbol{s},t)'\boldsymbol{\beta} + az(\boldsymbol{s},t) + \varepsilon(\boldsymbol{s},t) \\ z(\boldsymbol{s},t) &= gz(\boldsymbol{s},t-1) + \eta(\boldsymbol{s},t) \end{aligned}$$

- $\eta(\boldsymbol{s},t) \sim GP(0, \rho(\|\boldsymbol{s}-\boldsymbol{s}'\|;\boldsymbol{\theta}))$ is correlated over space but IID over time
- $z(\boldsymbol{s},t)$ is scalar and has Markovian dynamic
- a is a scale coefficient (v in D-STEM)
- g is the transition coefficient
- $\varepsilon(\boldsymbol{s},t) \sim N(0,\sigma_\varepsilon^2)$ is the measurement error
- The model parameter set is $\Psi = \{\boldsymbol{\beta}, a, \sigma_\varepsilon^2, \boldsymbol{\theta}, g\}$

- Which are the main differences with the DCM?
- Which model is better?

Spatio-temporal model: the (multivariate) hidden dynamic geostatistical model (HDGM)

$$\begin{aligned} \boldsymbol{y}(\boldsymbol{s},t) &= \boldsymbol{X}_{\boldsymbol{\beta}}(\boldsymbol{s},t)'\boldsymbol{\beta} + \boldsymbol{z}(\boldsymbol{s},t) + \boldsymbol{\varepsilon}(\boldsymbol{s},t) \\ \boldsymbol{z}(\boldsymbol{s},t) &= \boldsymbol{G}\boldsymbol{z}(\boldsymbol{s},t-1) + \boldsymbol{\eta}(\boldsymbol{s},t) \end{aligned}$$

- $\boldsymbol{y}(\boldsymbol{s},t)$ and $\boldsymbol{z}(\boldsymbol{s},t)$ are $p \times 1$ vectors
- $\boldsymbol{\eta}(\boldsymbol{s},t) \sim GP_p(\boldsymbol{0}, \boldsymbol{V}\rho(\|\boldsymbol{s}-\boldsymbol{s}'\|;\boldsymbol{\theta}))$ is a p-variate Gaussian random process
- $\boldsymbol{V}$ is a variance-covariance matrix (in Calculli et al. $\boldsymbol{V}$ is a correlation matrix and there is the scaling matrix $\boldsymbol{A}$)
- $\boldsymbol{z}(\boldsymbol{s},t)$ has Markovian dynamic
- $\boldsymbol{G}$ is a diagonal $p \times p$ transition matrix
- $\boldsymbol{\varepsilon}(\boldsymbol{s},t) \sim N_p(\boldsymbol{0}, \boldsymbol{\Sigma}_\varepsilon)$ is a p-variate Normal random variable with $\boldsymbol{\Sigma}_\varepsilon$ diagonal
- The model parameter set is $\Psi = \{\boldsymbol{\beta}, \boldsymbol{\Sigma}_\varepsilon, \boldsymbol{\theta}, \boldsymbol{V}, \boldsymbol{G}\}$

Model estimation

- The HDGM is estimated similarly to the DCM
- But we only have the $z(\boldsymbol{s},t)$ latent variable which is estimated in the E-step by the Kalman smoother.
- Spatial prediction is also done by the Kalman smoother assuming that y is not observed at the spatial prediction locations (it is added as NaN in the y vector).

Statistics for High Dimensional Data (and CompStat Lab)
a.a. 2023/2024 (2nd edition)

Prof. Francesco Finazzi francesco.finazzi@unibg.it
Prof. Alessandro Fassò alessandro.fasso@unibg.it

Lesson 6

Towards spatio-temporal functional models

- In many cases, data are observed at high frequency/resolution in at least one dimension (spatial or temporal)
- It usually happens with the temporal dimension
  - For instance, pollutant concentrations observed hourly or every 15 minutes
  - In general, high frequency observations (100, 1000, 10.000 per day)
- In space it is less common because a high resolution sampling is usually very expensive but…
  - In 3D space, one dimension may be sampled at higher resolution than the others

Towards spatio-temporal functional models

- Which are the problems with high frequency/resolution data?
  - Usually the original data set is very large and so the computational burden (of classic spatio-temporal models)
  - Data may be collected asynchronously over time (e.g., different monitoring stations may have different clocks)
  - Data may have large gaps over time (how does the Kalman Smoother perform in this case?)
  - Temporal correlation is usually very high (the Markovian model may explain the data but it is not very useful for prediction)

Functional data analysis (FDA)

- In FDA, the object of the statistical inference is a continuous function rather than scalar/vector values
- For instance, the temperature measured by a sensor over the 24h of the day can be described by a (smooth?) function
  - Independently of how many observations we take
  - Independently of where in time these observations are taken
- Which function or class of functions should we use?
  - The function should describe the «global» data pattern
  - In a way, the function filters out the data noisy
  - The researcher should be able to control the function smoothness

Splines

- Spline is a class of functions which allows to easily control the function smoothness
  - By selecting the proper basis functions
  - By selecting the proper knots
- B-spline basis are useful to describe non-periodic functions
  - Knots can be placed ad-hoc along the function domain (more knots where the function should change more rapidly)
- Fourier basis can describe periodic functions
  - Smoothness is controlled by the number of basis

How to describe functional data in a space-time model?

- We now want to model the generic observation $y(\boldsymbol{s}, t, h)$
  - $\boldsymbol{s}$ and t are the usual spatial and temporal indexes
  - $h \in \mathbb{R}$ is the «functional» dimension (spatial or temporal)
- Examples
  - h could describe the continuous time within the day while t is the index of days
  - h could describe altitude in a 3D space while $\boldsymbol{s}$ describes the generic location across the globe

The functional HDG model in D-STEM

- D-STEM implements the (univariate) functional version of the HDG model:

$$
\begin{aligned}
y(\boldsymbol{s}, t, h) &= \boldsymbol{x}(\boldsymbol{s}, t, h)'\boldsymbol{\beta}(h) + \boldsymbol{\phi}(h)'\boldsymbol{z}(\boldsymbol{s}, t) + \varepsilon(\boldsymbol{s}, t, h) \\
\boldsymbol{z}(\boldsymbol{s}, t) &= \boldsymbol{G}\boldsymbol{z}(\boldsymbol{s}, t-1) + \boldsymbol{\eta}(\boldsymbol{s}, t)
\end{aligned}
$$

- $\boldsymbol{\phi}(h)$ are the basis functions, $\boldsymbol{z}(\boldsymbol{s}, t)$ are the spline coefficients
- All details are in Wang et al. (2021) Journal statistical software