

POLITECNICO DI MILANO

Scuola di Ingegneria Industriale e dell'Informazione

Corso di Studi in Ingegneria Matematica



Tesi di Laurea Magistrale

TITOLO

Relatore: Prof. Laura SANGALLI
Correlatore: Dott. Ing. Mara BERNARDI

Tesi di Laurea di:
Gabriele Mazza Matr. 798794

Anno Accademico 2013-2014

Sommario

Il presente lavoro di tesi propone un modello separabile per l'analisi funzionale di dati distribuiti in spazio e tempo. Il modello è particolarmente adatto allo studio di problemi con domini spaziali complessi, con concavità o buchi. La capacità di tener conto della geometria spaziale rappresenta un forte vantaggio per questo modello, che sarà paragonato alle altre tecniche già esistenti, comprese quelle che non presentano tale particolarità. Oltre allo sviluppo analitico è stato creato un algoritmo e un codice R per il calcolo computazionale della stima. La principale applicazione studiata in questa tesi riguarda la produzione di rifiuti urbani pro capite nei comuni della provincia di Venezia tra il 1997 e il 2011, con un'attenzione speciale agli effetti legati al turismo.

Abstract

The present work of thesis proposes a separable model for the functional analysis of data distributed in space and time. The model is particularly appropriate to the study of problems with complex spatial domains, with concavities or holes. The ability to consider the spatial geometry is a strong advantage of this model, which will be compared to other existing techniques, including those that haven't this peculiarity. In addition to the analytical definition, an algorithm ad an R code for the computation of the estimate has been created. The main application studied in this thesis regards the production of urban waste per capita in the municipalities of Venice province between 1997 and 2011, with a special attention to the effects related to tourism.

Ringraziamenti

da fare

Indice

Introduzione	1
1 Panoramica sui modelli già esistenti	5
2 Presentazione del modello ST-PDE	7
2.1 Caso senza covariate	7
2.1.1 Dati e modello	7
2.1.2 Definizione delle funzioni di base	8
2.1.3 Funzionale di penalizzazione	9
2.1.4 Modello ideale	10
2.1.5 Discretizzazione dei termini di penalizzazione delle derivate	11
2.1.6 Soluzione del problema di stima	14
2.1.7 Proprietà statistiche dello stimatore	16
2.2 Caso con covariate	17
2.2.1 Proprietà statistiche degli estimatori	18
2.3 Stima della varianza di modello e scelta dei parametri di smoothing	18
2.3.1 Varianza di modello	19
2.3.2 Parametri di smoothing	19
3 Sviluppo del codice R	21
3.1 Funzioni di base considerate	21
3.2 Schematizzazione dell'algoritmo di stima	22
3.3 Strutture dati adottate	23
4 Simulazioni nel caso del dominio a forma di C	25
4.1 Triangolazione e istanti temporali	26
4.2 Caso senza covariata	27

4.3	Caso con covariata	30
5	Confronto con altri metodi	35
5.1	Caso senza covariata	37
5.2	Caso con covariata	39
6	Applicazione alla produzione di rifiuti urbani nella provincia di Venezia	41
6.1	Il turismo come possibile covariata	42
6.2	Trattamento del dominio	42
6.2.1	Regression splines	43
6.2.2	Replica dei dati	44
6.2.3	Triangolazione del dominio	45
6.3	Analisi preliminare dei dati	47
6.3.1	Funzioni di base	50
6.4	Applicazione del modello senza covariata	51
6.5	Applicazione del modello con covariata	55
7	Conclusioni e sviluppi futuri	61

Elenco delle figure

1	Dominio spaziale e locazioni dei dati	1
2	Rifiuti e posti letto pro capite	2
3	Differenza tra i dati nei comuni di Cavallino-Treporti e Quar-	
	to d'Altino	3
3.1	Funzioni di base implementate nel codice	22
4.1	Funzione spaziale $g(\underline{p})$	25
4.2	Triangolazione del dominio a forma di C	26
4.3	Stime della funzione $f(\underline{p}, t)$ ad alcuni istanti di tempo, caso senza covariata	28
4.4	Evoluzione temporale in alcuni nodi della triangolazione, caso senza covariata	29
4.5	Stime della funzione $f(\underline{p}, t)$ ad alcuni istanti di tempo, caso con covariata	31
4.6	Evoluzione temporale in alcuni nodi della triangolazione, caso con covariata	32
5.1	Confronto tra i metodi, caso senza covariata	37
5.2	Per alcuni istanti di tempo, funzione test $f(\underline{p}, t)$ reale, dati simulati, stime ottenute rispettivamente con kriging spazio- temporale, soap film smoothing, thin plate splines e ST- PDE, caso senza covariata	38
5.3	Confronto tra i metodi, caso con covariata	39
5.4	Per alcuni istanti di tempo, funzione test $f(\underline{p}, t)$ reale, dati simulati, stime ottenute rispettivamente con soap film smoo- thing, thin plate splines e ST-PDE, caso con covariata	40
6.1	Poligoni disponibili nel pacchetto <i>raster</i>	43

6.2	Smoothing con <i>Regression Splines</i> cubiche per il primo poligono dell'entroterra della provincia di Venezia	45
6.3	Frontiera e punti spaziali per la provincia di Venezia	46
6.4	Dati replicati	47
6.5	Triangolazione della provincia di Venezia	48
6.6	Rifiuti urbani pro capite ogni due anni dal 1997 al 2011	48
6.7	Posti letto pro capite ogni due anni dal 1997 al 2011	49
6.8	Andamento temporale della produzione dei rifiuti urbani nei comuni	50
6.9	Comuni selezionati per l'analisi dei risultati	51
6.10	Stima della funzione spazio-temporale della produzione di rifiuti urbani pro capite nella provincia di Venezia a tempi fissati dal 1997 al 2011, caso senza covariata	53
6.11	Stima della produzione di rifiuti urbani pro capite in alcuni comuni, caso senza covariata	54
6.12	Stima della parte funzionale della produzione di rifiuti urbani pro capite nella provincia di Venezia a tempi fissati dal 1997 al 2011, caso con covariata	57
6.13	Stima della parte funzionale produzione della produzione di rifiuti urbani pro capite in alcuni comuni, caso con covariata	58

Elenco delle tabelle

- 3.1 Tempo di calcolo della stima di \hat{c} (in secondi) nelle simulazioni eseguite sul dominio a forma di C 24

Introduzione

Il presente lavoro di tesi si occupa della costruzione e dell'applicazione di un modello statistico per l'analisi funzionale di dati tempo-varianti e distribuiti su un dominio spaziale. Questa ricerca è motivata dalla necessità di avere un buon metodo d'analisi per i dati riguardanti la produzione dei rifiuti urbani pro capite nei comuni della provincia di Venezia. I dati sono stati raccolti ed elaborati dall'Agenzia Regionale per la Prevenzione e Protezione Ambientale del Veneto (Arpav) e sono disponibili sul sito di Open Data Veneto¹. Le misurazioni sono state registrate per comune. In fig. 1 è stata tracciata la descrizione del dominio con i punti in cui i dati sono localizzati. Il problema presenta anche la dipendenza dal tempo, poiché i dati sono riportati annualmente dal 1997 al 2011.

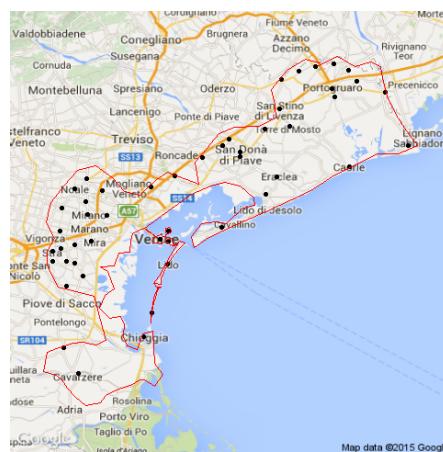


Figura 1: Dominio spaziale e locazioni dei dati

¹<http://dati.veneto.it/dataset/produzione-annua-di-rifiuti-urbani-totale-e-pro-capite-1997-2011>

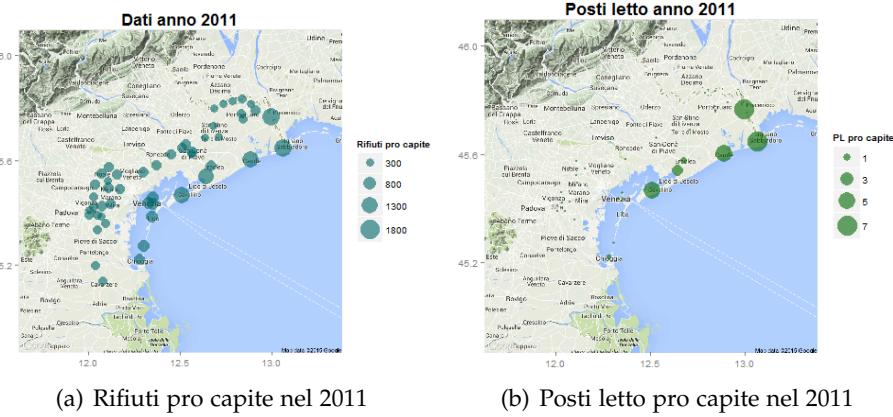


Figura 2: Rifiuti e posti letto pro capite

La produzione di rifiuti urbani sarà certamente influenzata anche dal turismo. Per questo motivo sarà considerato nel modello anche questo effetto, non trascurabile nella stima, attraverso il numero di posti letto pro capite nei comuni (maggiori dettagli saranno forniti in seguito). In fig. 2 sono visualizzati i valori dei rifiuti e dei posti letto pro capite nel 2011, anno più recente nel dataset, come esempio di ciò che si ha a disposizione.

Il problema della produzione dei rifiuti nella provincia di Venezia richiede la formulazione di un modello per l'analisi funzionale di dati definiti in spazio e tempo particolarmente adatto. Sono già disponibili in letteratura tecniche per dati di questo tipo ma non tutte sono adatte al trattamento di questi dati. La geometria del dominio, infatti, è complessa. In alcuni punti essa ha una definizione per nulla grossolana e presenta una grossa concavità nella laguna di Venezia, oltre alle numerose isole. Pertanto non si può evitare di considerare le particolarità del dominio spaziale.

Si consideri ad esempio quanto riportato in 3. Sono stati riportati i comuni di Quarto d'Altino e Cavallino-Treporti, tra di loro non eccessivamente distanti in linea d'aria ma con valori di produzione dei rifiuti decisamente diversi (come si può notare dagli andamenti temporali). Se i dati fossero analizzati con una tecnica che non tiene conto della geometria spaziale i due comuni sarebbero ritenuti vicini tra loro, quando in realtà si ha una grande separazione dovuta alla laguna di Venezia. Tecniche che trascurano la forma del dominio non sono appropriate poiché porterebbero ad una cattiva stima del fenomeno.

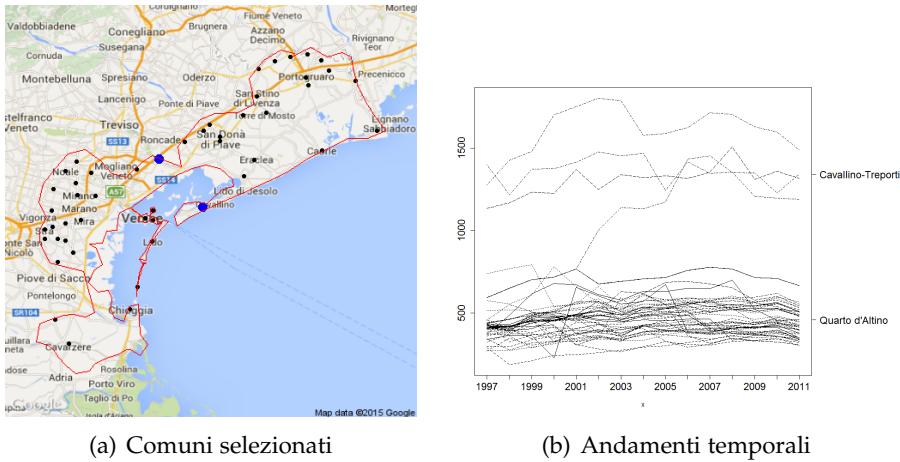


Figura 3: Differenza tra i dati nei comuni di Cavallino-Treporti e Quarto d'Altino

In questa tesi sarà illustrato il modello statistico *Spatio-Temporal Regression model with PDE penalization* (ST-PDE) per l'analisi funzionale di dati distribuiti in spazio e tempo. Questo modello si dimostrerà in grado di assolvere pienamente questo compito e di fornire una buona stima dell'andamento della produzione di rifiuti urbani nella provincia di Venezia. La funzione sarà stimata dalla minimizzazione di un funzionale di penalizzazione con termini differenziali, quindi sarà garantita anche la regolarità della stima. La funzione stimata sarà espressa secondo opportune funzioni di base (elementi finiti in spazio, B-splines in tempo) capaci di tener conto della geometria spaziale. Tramite l'aggiunta di un termine con covariata sarà possibile introdurre anche dell'effetto del turismo. Dalla modellizzazione matematica è stato sviluppato un algoritmo e il conseguente codice R per il calcolo della soluzione numerica della stima.

Il lavoro di tesi sarà strutturato come segue. Nel capitolo 1 è riportato una panoramica dei metodi già esistenti in letteratura. Nel capitolo 2 è presentata la costruzione del modello matematico ST-PDE, seguita dallo sviluppo del codice R descritto nel capitolo 3. Nel capitolo 4 si hanno i primi risultati, derivanti dall'applicazione del modello e del codice R a simulazioni eseguite sul dominio a forma di C descritto in [5] e [7], per il quale è possibile valutare la bontà delle stime ottenute grazie alla perfetta conoscenza del fenomeno reale in ogni punto e in ogni istante. Nel capitolo 5 il modello ST-PDE sarà paragonato ad altri metodi già esistenti per il confronto delle stime ottenute. Nel capitolo 6 si ha l'applicazione allo studio della produzione dei rifiuti nella provincia di Venezia e infine, nel capitolo 7, sono raccolte le conclusioni e i possibili sviluppi futuri.

CAPITOLO 1

Panoramica sui modelli già esistenti

Il modello ST-PDE si inserisce nell'ambito dell'analisi funzionale di dati distribuiti in spazio e tempo. Sia $\underline{p} = (x, y) \in \Omega$ il vettore con le coordinate del punto spaziale, $t \in [T_1, T_2]$ l'istante temporale e $z \in \mathbb{R}$ la variabile risposta. Le metodologie classiche di analisi funzionale, esposte ad esempio in CITA RAMSAY SILVERMAN, possono essere applicate per la stima di un dato tempo-variante. Tuttavia in questo caso si ha anche la dipendenza spaziale, pertanto la funzione da stimare sarà:

$$z = f(\underline{p}, t).$$

Secondo quanto è solitamente fatto per problemi di analisi di dati funzionali, sarà necessario minimizzare un funzionale di penalizzazione che contenga non soltanto gli scarti tra valori osservati e stimati ma anche termini differenziali per la regolarità della stima.

La letteratura su modelli spazio-temporali è già ampia, come è esposto in CITA CRESSIE. Tuttavia le metodologie già disponibili presentano tra loro delle differenze. Non tutti i modelli sono adatti ad essere applicati alla stima della produzione di rifiuti urbani nella provincia di Venezia a causa della forte dipendenza dalla geometria del problema. Per questo motivo, dopo la definizione del modello ST-PDE, sarà dedicato anche uno spazio al confronto delle stime dai diversi metodi a disposizione su dati appositamente generati.

Ad esempio, il kriging spazio-temporale, descritto in CITAZIONI, non è in grado di tener conto del dominio spaziale. Questa tecnica sarà considerata negli studi di simulazione e nei confronti ma non potrà garantire risultati ottimali per questo tipo di analisi.

Le pubblicazioni [1] e [4] studiano metodi per l'analisi funzionale di dati distribuiti in spazio e tempo. Questo lavori, però, si basano su modell-

li che possono assumere una forma più complessa, ipotizzando che una funzione del valore atteso della risposta (nel loro caso il logaritmo) possa essere spiegato da uno o più termini funzionali (quest'ultima possibilità è riportata solo in [1]):

$$\log(\mathbb{E}[z_i]) = f_1 + f_2 + \dots + f_N.$$

Ognuna di queste funzioni può avere dipendenza solo spaziale, solo temporale o entrambe. Queste complicazioni rispetto al modello che sarà studiato da ST-PDE possono essere risolte molto velocemente ipotizzando che la risposta dipenda da un solo termine spazio-temporale. Le tecniche proposte dagli autori sono un ottimo termine di paragone per il modello ST-PDE. La loro applicabilità per un problema con forte dipendenza dalla forma del dominio, come sarà spiegato più nel dettaglio in 5, dipende dalla scelta delle funzioni di base.

Particolare attenzione deve essere posta riguardo a quanto fatto in [6]. In questa pubblicazione è studiato un modello solamente spaziale, definito su un dominio Ω . Tale dominio può assumere le stesse caratteristiche che saranno studiate in questa tesi: definizione complessa, concavità o buchi. La risposta è espressa dal seguente modello:

$$z = f(\underline{p})$$

Per poter stimare la funzione f è introdotto il seguente funzionale di penalizzazione da minimizzare:

$$J_\lambda(f(\underline{p})) = \sum_{i=1}^n (z_i - f(\underline{p}_i))^2 + \lambda \int_{\Omega} (\Delta(f(\underline{p})))^2 d\Omega.$$

Il problema di minimo è risolto attraverso il passaggio ad una forma variazionale che, a seguito dell'introduzione di un'opportuna espansione in elementi finiti per f , è ridotta ad un problema discreto risolvibile con un sistema lineare. L'articolo [6] non contempla la variazione temporale per f ma può essere considerato il punto di partenza per il modello ST-PDE. Non sarà ricavata una forma variazionale per la stima della soluzione, ma l'attenzione alla geometria del problema e l'uso degli elementi finiti rappresentano un forte legame con il modello ST-PDE.

CAPITOLO 2

Presentazione del modello ST-PDE

In questo capitolo è descritto nel dettaglio il modello ST-PDE per l'analisi di dati distribuiti in spazio e tempo ed è calcolata la soluzione al problema di stima.

2.1 Caso senza covariate

2.1.1 Dati e modello

Sia $\{\underline{p}_i = (x_i, y_i); i = 1, \dots, n\}$ un insieme di n punti spaziali in un dominio limitato $\Omega \subset \mathbb{R}^2$ e sia $\{t_j; j = 1, \dots, m\}$ un insieme di m istanti temporali in un intervallo $[T_1, T_2] \subset \mathbb{R}$. In questi punti ed istanti si osservano i dati: siano quindi $\{z_{ij}; i = 1, \dots, n; j = 1, \dots, m\}$ i valori della variabile risposta nel punto \underline{p}_i al tempo t_j .

Come già riportato nel capitolo precedente, l'obiettivo sarà la stima di un campo spazio-temporale che rappresenti il fenomeno da cui sono stati raccolti i dati. Quindi si ipotizza che le osservazioni z_{ij} siano generate da tale campo spazio-temporale con l'aggiunta di rumore:

$$z_{ij} = f(\underline{p}_i, t_j) + \varepsilon_{ij} \quad i = 1, \dots, n \quad j = 1, \dots, m , \quad (2.1)$$

dove $\{\varepsilon_{ij}; i = 1, \dots, n; j = 1, \dots, m\}$ sono residui indipendenti identicamente distribuiti di media nulla e varianza σ^2 . L'obiettivo del modello ST-PDE sarà la stima della funzione $f(\underline{p}, t)$ dalle nm osservazioni a disposizione, minimizzando un funzionale $J_{\lambda}(f(\underline{p}, t))$ con la somma degli scarti quadratici tra le osservazioni e valori stimati e termini di penalizzazione delle derivate per la regolarità della funzione.

2.1.2 Definizione delle funzioni di base

Dall'analisi della letteratura già disponibile per modelli simili, anche se solo spaziali, si può dedurre che non è possibile dare una stima della funzione se essa non risulta espressa in una espansione di opportune funzioni di base. Infatti l'infinita possibilità di variazione di una funzione in un qualsiasi spazio funzionale non renderebbe possibile una stima computazionale del risultato. L'approccio scelto per la costruzione di $f(\underline{p}, t)$ si basa sulla generalizzazione delle espansioni in funzione di base dei casi puramente spaziali o temporali.

Siano

$$\{\varphi_k(t); k = 1, \dots, M\} \subset H^2([T_1, T_2])$$

un insieme di M funzioni di base definite sull'intervallo temporale $[T_1, T_2]$ e

$$\{\psi_l(\underline{p}); l = 1, \dots, N\} \subset H^1(\Omega)$$

in insieme di N funzioni di base definite sul dominio spaziale Ω . Con le combinazioni lineari

$$\sum_{k=1}^M a_k \varphi_k(t) \quad \sum_{l=1}^N b_l \psi_l(\underline{p})$$

è possibile costruire, rispettivamente, funzioni varianti soltanto in tempo e in spazio. La funzione $f(\underline{p}, t)$ nasce ipotizzando che i coefficienti costanti delle espansioni in base precedenti possano variare secondo la variabile che non è espressa dalle funzioni di base:

$$f(\underline{p}, t) = \sum_{k=1}^M a_k(\underline{p}) \varphi_k(t) \tag{2.2}$$

$$f(\underline{p}, t) = \sum_{l=1}^N b_l(t) \psi_l(\underline{p}) . \tag{2.3}$$

Si ha quindi che $\{a_k(\underline{p}); k = 1, \dots, M\}$ sono i coefficienti spazio-varianti dell'espansione in basi di temporali e $\{b_l(t); l = 1, \dots, N\}$ sono i coefficienti tempo-varianti dell'espansione in basi spaziali. In questa costruzione la funzione $f(\underline{p}, t)$ possiede entrambe queste rappresentazioni.

Sono state ricavate due espressioni equivalenti per f ma, come si potrà notare in seguito, questo è coerente (sarà ricavata un'espressione più specifica). Per una corretta definizione del funzionale di penalizzazione $J_{\underline{\lambda}}(f(\underline{p}, t))$ è necessario introdurre, per tali coefficienti, le seguenti condizioni:

$$a_k(\underline{p}) \in H_{n_0}^2(\Omega) \quad \forall k = 1, \dots, M$$

e

$$b_l(t) \in H^2([T_1, T_2]) \quad \forall l = 1, \dots, N ,$$

dove $H_{n_0}^2(\Omega) = \{h \in H^2(\Omega) | \partial_\nu h = 0 \text{ su } \partial\Omega\}$, spazio incluso in $H^2(\Omega)$ con condizioni di Neumann alla frontiera. Altre condizioni possono essere poste (a patto che garantiscano l'applicabilità del funzionale di penalizzazione) ma, per semplicità, saranno ipotizzate queste.

2.1.3 Funzionale di penalizzazione

Per poter stimare $f(\underline{p}, t)$ si introduce la minimizzazione di un funzionale non formato solamente dagli scarti quadratici tra le osservazioni e le stime negli nm punti disponibili. Sono inclusi in esso anche altri due termini, che derivano dalla penalizzazione di opportune derivate in spazio e tempo per poter garantire regolarità alla funzione. Analogamente a quanto fatto il 2.1.2, per costruire tale funzionale si considerano innanzitutto i problemi marginali in spazio e tempo.

Per garantire una buona regolarità spaziale alla funzione è necessario penalizzare il quadrato della norma L^2 del laplaciano (dove per laplaciano si intenderà, da ora in avanti, rispetto alle variabili in spazio \underline{p}). La stessa scelta è stata fatta in altre pubblicazioni come [5], [6] e [7]. Quindi se $g(\underline{p}) : \Omega \mapsto \mathbb{R}$ è una funzione spazio-variante, allora si può definire la penalizzazione della regolarità in spazio tramite:

$$J_S(g(\underline{p})) = \int_{\Omega} (\Delta g(\underline{p}))^2 d\underline{p} .$$

Analogamente in tempo si avrà la penalizzazione del quadrato della norma L^2 della derivata seconda. Se $h(t) : [T_1, T_2] \mapsto \mathbb{R}$, allora si avrà

$$J_T(h(t)) = \int_{T_1}^{T_2} \left(\frac{\partial^2 h(t)}{\partial t^2} \right)^2 dt .$$

Grazie alle ipotesi di regolarità introdotte in sez. 2.1.2 tali penalizzazioni possono essere applicate ai coefficienti degli sviluppi delle eq. 2.2 e 2.3. Per questo motivo si definisce:

$$\begin{aligned} J_{\underline{\Lambda}}(f(\underline{p}, t)) &= \sum_{i=1}^n \sum_{j=1}^m (z_{ij} - f(p_i, t_j))^2 + \\ &\quad + \lambda_S \sum_{k=1}^M J_S(a_k(\underline{p})) + \lambda_T \sum_{l=1}^N J_T(b_l(t)) , \end{aligned}$$

cioè

$$\begin{aligned} J_{\underline{\Lambda}}(f(\underline{p}, t)) &= \sum_{i=1}^n \sum_{j=1}^m (z_{ij} - f(p_i, t_j))^2 + \\ &\quad + \lambda_S \sum_{k=1}^M \int_{\Omega} (\Delta(a_k(\underline{p})))^2 d\underline{p} + \lambda_T \sum_{l=1}^N \int_{T_1}^{T_2} \left(\frac{\partial^2 b_l(t)}{\partial t^2} \right)^2 dt , \quad (2.4) \end{aligned}$$

dove $\lambda_S > 0$ e $\lambda_T > 0$ sono i parametri di smoothing che stabiliscono il peso della penalizzazione della regolarità della funzione rispettivamente in spazio e tempo. Se troppo alti, la funzione stimata tenderà ad essere quasi liscia e distante dai dati. Al contrario, se troppo bassi, la funzione stimata sarà vicina all'interpolazione dei dati e per nulla liscia. Sebbene quest'ultimo caso possa sembrare molto buono poiché vicino ai valori osservati, non è ciò che si desidera, poiché crea una stima troppo dipendente dai dati e spesso con variazioni eccessivamente repentine. Un corretto bilanciamento di questi casi garantisce una buona descrizione del fenomeno.

2.1.4 Modello ideale

Il funzionale di penalizzazione riportato in eq. 2.4 è il risultato di una costruzione derivante dalle penalizzazioni marginali in spazio e tempo, ma idealmente si può ricondurre al seguente:

$$\begin{aligned} \tilde{J}_\lambda(f) = & \sum_{i=1}^n \sum_{j=1}^m (z_{ij} - f(\underline{p}_i, t_j))^2 + \\ & + \lambda_S \int_{T_1}^{T_2} \int_{\Omega} (\Delta f(\underline{p}, t))^2 d\underline{p} dt + \lambda_T \int_{\Omega} \int_{T_1}^{T_2} \left(\frac{\partial^2 f(\underline{p}, t)}{\partial t^2} \right)^2 dt d\underline{p}, \end{aligned} \quad (2.5)$$

dove J_S e J_T sono applicati direttamente alla funzione $f(\underline{p}, t)$ e sono integrati, rispettivamente, sull'intervallo spaziale e il dominio spaziale. Se si applica la forma di $f(\underline{p}, t)$ dell'eq. 2.2 nel termine di penalizzazione del laplaciano in spazio, allora si può ritrovare:

$$\begin{aligned} \int_{T_1}^{T_2} \int_{\Omega} (\Delta f(\underline{p}, t))^2 d\underline{p} dt &= \int_{T_1}^{T_2} \int_{\Omega} \left(\Delta \left(\sum_{k=1}^M a_k(\underline{p}) \varphi_k(t) \right) \right)^2 d\underline{p} dt \\ &= \int_{T_1}^{T_2} \int_{\Omega} \left(\sum_{k=1}^M \Delta a_k(\underline{p}) \varphi_k(t) \right)^2 d\underline{p} dt \\ &= \int_{T_1}^{T_2} \int_{\Omega} \left(\sum_{k=1}^M \Delta a_k(\underline{p}) \varphi_k(t) \right) \left(\sum_{h=1}^M \Delta a_h(\underline{p}) \varphi_h(t) \right) d\underline{p} dt \\ &= \int_{T_1}^{T_2} \int_{\Omega} \left(\sum_{k=1}^M \sum_{h=1}^M \Delta a_k(\underline{p}) \Delta a_h(\underline{p}) \varphi_k(t) \varphi_h(t) \right) d\underline{p} dt \\ &= \sum_{k=1}^M \sum_{h=1}^M \int_{\Omega} \Delta a_k(\underline{p}) \Delta a_h(\underline{p}) d\underline{p} \int_{T_1}^{T_2} \varphi_k(t) \varphi_h(t) dt. \end{aligned} \quad (2.6)$$

Questo termine è equivalente a quello proposto in 2.4 se le basi temporali siano ortonormali, poiché in tal caso l'ultimo integrale in eq. 2.6 vale 1 se $k = h$ e 0 altrimenti.

Allo stesso modo, se si sostituisce la forma di $f(\underline{p}, t)$ dell'eq. 2.3 nella penalizzazione ideale dell'eq. 2.5 si ottiene:

$$\begin{aligned}
 \int_{\Omega} \int_{T_1}^{T_2} \left(\frac{\partial^2 f(\underline{p}, t)}{\partial t^2} \right)^2 dt d\underline{p} &= \int_{\Omega} \int_{T_1}^{T_2} \left(\frac{\partial^2 \sum_{l=1}^N b_l(t) \psi_l(\underline{p})}{\partial t^2} \right)^2 dt d\underline{p} \\
 &= \int_{\Omega} \int_{T_1}^{T_2} \left(\sum_{l=1}^N \frac{\partial^2 b_l(t)}{\partial t^2} \psi_l(\underline{p}) \right)^2 dt d\underline{p} \\
 &= \int_{\Omega} \int_{T_1}^{T_2} \left(\sum_{l=1}^N \frac{\partial^2 b_l(t)}{\partial t^2} \psi_l(\underline{p}) \right) \left(\sum_{h=1}^N \frac{\partial^2 b_h(t)}{\partial t^2} \psi_h(\underline{p}) \right) dt d\underline{p} \\
 &= \int_{\Omega} \int_{T_1}^{T_2} \left(\sum_{l=1}^N \sum_{h=1}^N \frac{\partial^2 b_l(t)}{\partial t^2} \frac{\partial^2 b_h(t)}{\partial t^2} \psi_l(\underline{p}) \psi_h(\underline{p}) \right) dt d\underline{p} \\
 &= \sum_{l=1}^N \sum_{h=1}^N \int_{T_1}^{T_2} \frac{\partial^2 b_l(t)}{\partial t^2} \frac{\partial^2 b_h(t)}{\partial t^2} dt \int_{\Omega} \psi_l(\underline{p}) \psi_h(\underline{p}) d\underline{p}.
 \end{aligned} \tag{2.7}$$

La stessa osservazione del caso precedente vale anche ora: se le basi in spazio sono ortonormali, si ritrova la penalizzazione proposta in 2.4.

In questo lavoro di tesi, quindi, è proposto un modello che risulterà essere computazionalmente semplice ma non perfettamente equivalente a questo modello ideale. In generale le funzioni di base non sono ortonormali. Comunque sia, gli insiemi di basi che saranno adattati sono sparsi, cioè i termini $\int_{T_1}^{T_2} \varphi_k(t) \varphi_l(t) dt$ e $\int_{\Omega} \psi_l(\underline{p}) \psi_k(\underline{p}) d\underline{p}$ sono diversi da zero solo per poche coppie di indici.

2.1.5 Discretizzazione dei termini di penalizzazione delle derivate

Così come è scritto in 2.4, il funzionale $J_{\underline{\lambda}}(f(\underline{p}, t))$ non è ancora adatto ad essere trattato computazionalmente. Per poter avere una forma che renda semplice la stima, $f(\underline{p}, t)$ e $J_{\underline{\lambda}}(f(\underline{p}, t))$ saranno nuovamente discretizzati.

Il primo caso da trattare è l'integrale del laplaciano dei coefficienti spazio-varianti dell'eq. 2.2 in modo analogo a quanto fatto in [6]. Fissato k , l'integrale

$$\int_{\Omega} (\Delta(a_k(\underline{p})))^2 d\underline{p}$$

può essere semplificato introducendo la funzione $g_k(\underline{p}) \in L^2(\Omega)$ come segue:

$$\int_{\Omega} g_k(\underline{p}) v(\underline{p}) d\Omega = \int_{\Omega} \Delta(a_k(\underline{p})) v(\underline{p}) d\Omega \quad \forall v(\underline{p}) \in L^2(\Omega). \tag{2.8}$$

Non è difficile verificare che, se $g_k(\underline{p})$ rispetta l'equazione precedente, per l'arbitrarietà di v allora:

$$\int_{\Omega} \left(\Delta(a_k(\underline{p})) \right)^2 d\underline{p} = \int_{\Omega} \Delta(a_k(\underline{p})) g_k(\underline{p}) d\underline{p}. \quad (2.9)$$

Applicando la formula di Green e tenendo conto delle condizioni di Neumann per $a_k(\underline{p})$, si possono semplificare gli integrali eliminando l'uso del laplaciano:

$$\begin{aligned} \int_{\Omega} \Delta(a_k(\underline{p})) g_k(\underline{p}) d\underline{p} &= - \int_{\Omega} \nabla a_k(\underline{p}) \nabla g_k(\underline{p}) d\underline{p} \\ \int_{\Omega} \Delta(a_k(\underline{p})) v(\underline{p}) d\underline{p} &= - \int_{\Omega} \nabla a_k(\underline{p}) \nabla v(\underline{p}) d\underline{p}. \end{aligned}$$

Per poter calcolare analiticamente questi integrali è necessario introdurre l'uso delle basi spaziali $\{\psi_l(\underline{p}); l = 1, \dots, N\}$ per le funzioni a_k , g_k e v . Siano quindi:

$$a_k(\underline{p}) = \sum_{l=1}^N c_{lk} \psi_l(\underline{p}) \quad g_k(\underline{p}) = \sum_{l=1}^N g_{lk} \psi_l(\underline{p}) \quad v(\underline{p}) = \sum_{l=1}^N v_l \psi_l(\underline{p}).$$

Per semplificare le notazioni saranno usati i seguenti vettori:

$$\underline{c}_k = \begin{bmatrix} c_{1k} \\ c_{2k} \\ \vdots \\ c_{Nk} \end{bmatrix} \quad \underline{g}_k = \begin{bmatrix} g_{1k} \\ g_{2k} \\ \vdots \\ g_{Nk} \end{bmatrix} \quad \underline{v} = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{bmatrix}$$

e gli analoghi per le funzioni di base e le loro derivate parziali:

$$\begin{aligned} \underline{\psi} &= \begin{bmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \psi_N \end{bmatrix} \\ \underline{\psi}_x &= \begin{bmatrix} \partial \psi_1 / \partial x \\ \partial \psi_2 / \partial x \\ \vdots \\ \partial \psi_N / \partial x \end{bmatrix} \quad \underline{\psi}_y = \begin{bmatrix} \partial \psi_1 / \partial y \\ \partial \psi_2 / \partial y \\ \vdots \\ \partial \psi_N / \partial y \end{bmatrix}. \end{aligned} \quad (2.10)$$

Mediante l'uso delle funzioni di base e di ciò che è stato ottenuto dall'applicazione della formula di Green, le relazioni 2.8 e 2.9 diventano:

$$g_k \left(\int_{\Omega} \underline{\psi} \underline{\psi}^T \right) \underline{v} = -\underline{c}_k \left(\int_{\Omega} (\underline{\psi}_x \underline{\psi}_x^T + \underline{\psi}_y \underline{\psi}_y^T) \right) \underline{v} \quad \forall \underline{v} \in \mathbb{R}^N$$

$$\int_{\Omega} \left(\Delta(a_k(\underline{p})) \right)^2 d\underline{p} = -c_k \left(\int_{\Omega} (\underline{\psi}_x \underline{\psi}_x^T + \underline{\psi}_y \underline{\psi}_y^T) \right) \underline{g}_k .$$

Quindi, se si introducono le matrici (analogamente a quanto fatto in [6])

$$R_0 = \int_{\Omega} \underline{\psi} \underline{\psi}^T$$

$$R_1 = \int_{\Omega} (\underline{\psi}_x \underline{\psi}_x^T + \underline{\psi}_y \underline{\psi}_y^T) ,$$

si trova, per l'arbitrarietà di \underline{v} :

$$\int_{\Omega} \left(\Delta(a_k(\underline{p})) \right)^2 d\underline{p} = c_k^T R_1 R_0^{-1} R_1 c_k = c_k^T P_S c_k \quad (2.11)$$

Si noti che la matrice P_S non dipende da k , pertanto è la stessa per tutte le funzioni $a_k(p)$. Inoltre è simmetrica, poiché R_0 e R_1 lo sono.

Grazie all'introduzione della discretizzazione in basi spaziali

$$a_k(\underline{p}) = \sum_{l=1}^N c_{lk} \varphi_l(\underline{p})$$

è stato possibile ridurre la penalizzazione con l'integrale del quadrato di $\Delta a_k(p)$ alla valutazione di una forma quadratica che non cambia con le funzioni a_k . Si ha anche un'altra conseguenza: per l'equivalenza ipotizzata tra le espressioni di $f(p, t)$ in 2.2 e 2.3, allora è necessario che i coefficienti tempo-varianti dell'espansione in basi spaziali assumano la seguente forma:

$$b_l(t) = \sum_{k=1}^M c_{lk} \varphi_k(t) .$$

Si ritrova quindi anche in questo caso l'espansione in funzioni di base.

Non resta altro che discretizzare anche $\int_{T_1}^{T_2} \left(\frac{\partial^2 b_l(t)}{\partial t^2} \right)^2 dt$. Dopo aver introdotto l'uso delle funzioni di base, se si definisce

$$P_T = \begin{bmatrix} \int_{T_1}^{T_2} \varphi_1''(t) \varphi_1''(t) dt & \int_{T_1}^{T_2} \varphi_1''(t) \varphi_2''(t) dt & \dots & \int_{T_1}^{T_2} \varphi_1''(t) \varphi_M''(t) dt \\ \int_{T_1}^{T_2} \varphi_2''(t) \varphi_1''(t) dt & \int_{T_1}^{T_2} \varphi_2''(t) \varphi_2''(t) dt & \dots & \int_{T_1}^{T_2} \varphi_2''(t) \varphi_M''(t) dt \\ \vdots & \vdots & \dots & \vdots \\ \int_{T_1}^{T_2} \varphi_M''(t) \varphi_1''(t) dt & \int_{T_1}^{T_2} \varphi_M''(t) \varphi_2''(t) dt & \dots & \int_{T_1}^{T_2} \varphi_M''(t) \varphi_M''(t) dt \end{bmatrix}$$

e il vettore

$$\underline{c}_l = \begin{bmatrix} c_{l1} \\ c_{l2} \\ \dots \\ c_{lM} \end{bmatrix} ,$$

si ritrova:

$$\int_{T_1}^{T_2} \left(\frac{\partial^2 b_l(t)}{\partial t^2} \right)^2 dt = \underline{c}_l^T P_T \underline{c}_l .$$

Anche la matrice P_T è simmetrica.

Le forme quadratiche con P_S e P_T possono essere ulteriormente sviluppate per dimostrare che la parte di penalizzazione per la regolarizzazione di f in 2.4 è rappresentabile con un'unica forma quadratica. Per mostrarlo, si introduce il vettore

$$\underline{c} = \begin{bmatrix} c_{11} \\ \vdots \\ c_{1M} \\ c_{21} \\ \vdots \\ c_{2M} \\ \vdots \\ c_{NM} \end{bmatrix}$$

e la matrice P , definita con opportuni prodotti di Kronecker come segue:

$$P = \lambda_S (P_S \otimes I_M) + \lambda_T (I_N \otimes P_T) ,$$

dove I_M and I_N sono matrici identità di dimensioni $M \times M$ e $N \times N$ rispettivamente. Allora si avrà:

$$\begin{aligned} \lambda_S \sum_{k=1}^M \int_{\Omega} \left(\Delta(a_k(\underline{p})) \right)^2 d\underline{p} + \lambda_T \sum_{l=1}^N \int_{T_1}^{T_2} \left(\frac{\partial^2 b_l(t)}{\partial t^2} \right)^2 dt = \\ = \lambda_S \sum_{k=1}^M \underline{c}_k^T P_S \underline{c}_k + \lambda_T \sum_{l=1}^N \underline{c}_l^T P_T \underline{c}_l = \underline{c}^T P \underline{c} . \quad (2.12) \end{aligned}$$

A causa della simmetria dei termini con cui è costruita, anche la matrice P è simmetrica.

2.1.6 Soluzione del problema di stima

Grazie a quanto ricavato nel paragrafo precedente, la parte di penalizzazione delle derivate del funzionale $J_{\underline{\lambda}}(f(p, t))$ si è ridotta ad un'unica forma quadratica. Ma questo è stato possibile grazie all'espressione in funzione di base per i coefficienti delle eq. 2.2 e 2.3:

$$a_k(\underline{p}) = \sum_{l=1}^N c_{lk} \psi_l(\underline{p}) \quad b_l(t) = \sum_{k=1}^M c_{lk} \varphi_k(t) .$$

Essendo 2.2 e 2.3 equivalenti, allora in definitiva:

$$f(\underline{p}, t) = \sum_{l=1}^N \sum_{k=1}^M c_{lk} \psi_l(\underline{p}) \varphi_k(t) , \quad (2.13)$$

cioè la funzione da stimare è la combinazione lineare di tutti i possibili prodotti incrociati tra le funzioni di base in tempo e spazio. Questa formulazione può essere considerata la definitiva per la funzione $f(p, t)$ e permette di poter identificare la funzione con il vettore dei suoi coefficienti \underline{c} . Ne consegue anche la possibilità di scrivere in modo più agevole il funzionale $J_{\underline{\lambda}}(f(p, t))$. Siano definiti il vettore dei valori osservati

$$\underline{z} = \begin{bmatrix} z_{11} \\ \vdots \\ z_{1m} \\ z_{21} \\ \vdots \\ z_{2m} \\ \vdots \\ z_{nm} \end{bmatrix} \quad (2.14)$$

e le matrici Ψ (con le valutazioni delle basi spaziali nei punti $\{\underline{p}_i; i = 1, \dots, n\}$) e Φ (con le valutazioni delle basi temporali $\{t_j; j = 1, \dots, m\}$):

$$\Psi = \begin{bmatrix} \psi_1(\underline{p}_1) & \psi_2(\underline{p}_1) & \dots & \psi_N(\underline{p}_1) \\ \psi_1(\underline{p}_2) & \psi_2(\underline{p}_2) & \dots & \psi_N(\underline{p}_2) \\ \vdots & \vdots & \dots & \vdots \\ \psi_1(\underline{p}_n) & \psi_2(\underline{p}_n) & \dots & \psi_N(\underline{p}_n) \end{bmatrix}$$

$$\Phi = \begin{bmatrix} \varphi_1(t_1) & \varphi_2(t_1) & \dots & \varphi_M(t_1) \\ \varphi_1(t_2) & \varphi_2(t_2) & \dots & \varphi_M(t_2) \\ \vdots & \vdots & \dots & \vdots \\ \varphi_1(t_m) & \varphi_2(t_m) & \dots & \varphi_M(t_m) \end{bmatrix}.$$

Le ultime due matrici non sono utili da sole, ma moltiplicate tra loro con prodotto di Kronecker, poiché se

$$B = \Psi \otimes \Phi,$$

allora si può facilmente dire che:

$$\begin{bmatrix} f(\underline{p}_1, t_1) \\ \vdots \\ f(\underline{p}_1, t_m) \\ f(\underline{p}_2, t_1) \\ \vdots \\ f(\underline{p}_2, t_m) \\ \vdots \\ f(\underline{p}_n, t_m) \end{bmatrix} = B\underline{c}.$$

Quindi è possibile dare una forma definitiva al funzionale di penalizzazione:

$$J_{\underline{\lambda}}(\underline{c}) = (\underline{z} - B\underline{c})^T(\underline{z} - B\underline{c}) + \underline{c}^T P \underline{c}, \quad (2.15)$$

e per trovare la stima $\hat{f}(p, t)$ sarà sufficiente ricavare il vettore dei coefficienti \underline{c} risolvendo il problema di minimo:

$$\hat{\underline{c}} = \arg \min_{\underline{c} \in \mathbb{R}^{NM}} J_{\underline{\lambda}}(\underline{c}).$$

Mediante la formulazione ottenuta in 2.15 basta derivare per ottenere la soluzione al problema di stima. Grazie alla simmetria di P , si ha:

$$\frac{\partial}{\partial \underline{c}} J = -2B^T \underline{z} + 2(B^T B + P)\underline{c},$$

che posta uguale a zero porta all'equazione

$$(B^T B + P)\underline{c} = B^T \underline{z}$$

e in conclusione

$$\hat{\underline{c}} = (B^T B + P)^{-1} B^T \underline{z}. \quad (2.16)$$

Il problema di stima è stato risolto e la soluzione si può trovare risolvendo un sistema lineare, seppur di grandi dimensioni (la matrice $B^T B + P$ ha dimensioni $NM \times NM$ e già negli esempi si avranno dimensioni elevate).

L'ultimo elemento da definire è la *smoothing matrix* S , che permette di derivare i valori stimati dal modello direttamente da quelli osservati:

$$\hat{\underline{z}} = B\hat{\underline{c}} = B(B^T B + P)^{-1} B^T \underline{z} = S \underline{z}.$$

2.1.7 Proprietà statistiche dello stimatore

Il modello di partenza indicato in 2.1 può essere scritto anche in forma matriciale

$$\underline{z} = B\underline{c} + \underline{\varepsilon}. \quad (2.17)$$

A causa delle proprietà statistiche di $\underline{\varepsilon}$

$$\mathbb{E}[\underline{\varepsilon}] = \underline{0} \quad \text{Var}[\underline{\varepsilon}] = \sigma^2 I_{nm}$$

si ha

$$\mathbb{E}[\underline{z}] = B\underline{c} \quad \text{Var}[\underline{z}] = \sigma^2 I_{nm}$$

e quindi è immediato ricavare per lo stimatore $\hat{\underline{c}}$ (grazie alle proprietà simmetria di P):

$$\mathbb{E}[\hat{\underline{c}}] = (B^T B + P)^{-1} B^T B \underline{c} \quad \text{Var}[\hat{\underline{c}}] = \sigma^2 (B^T B + P)^{-1} B^T B (B^T B + P)^{-1}.$$

Non è stata ipotizzata la gaussianità per $\underline{\varepsilon}$, ma se così fosse anche $\hat{\underline{c}}$ sarebbe gaussiano. Grazie a quanto appena ricavato è possibile elaborare (con una data significatività) una regione di confidenza per $\hat{\underline{c}}$ e quindi una banda di confidenza per la funzione stimata f .

2.2 Caso con covariate

Il modello si estende facilmente se si prevede che il dato possa essere influenzato da covariate. La forma in eq. 2.1 diventa:

$$z_{ij} = \underline{w}_{ij}^T \underline{\beta} + f(\underline{p}_i, t_j) + \varepsilon_{ij} \quad i = 1, \dots, n \quad j = 1, \dots, m , \quad (2.18)$$

dove \underline{w}_{ij} è il vettore delle p covariate associate a z_{ij} e $\underline{\beta}$ è il vettore dei coefficienti di regressione. Di conseguenza il funzionale discreto di 2.15 diventa:

$$J_{\underline{\lambda}}(\underline{\varepsilon}) = (\underline{z} - W\underline{\beta} - B\underline{\varepsilon})^T (\underline{z} - W\underline{\beta} - B\underline{\varepsilon}) + \underline{\varepsilon}^T S \underline{\varepsilon} ,$$

dove W è la matrice $nm \times p$ con i vettori $\{\underline{w}_{ij}; i = 1, \dots, n; j = 1, \dots, m\}$.

Per trovare la soluzione occorre derivare questa espressione rispetto a $\underline{\beta}$ e $\underline{\varepsilon}$:

$$\frac{\partial}{\partial \underline{\beta}} J = -2W^T \underline{z} + 2W^T B \underline{\varepsilon} + 2W^T W \underline{\beta} ,$$

$$\frac{\partial}{\partial \underline{\varepsilon}} J = -2B^T \underline{z} + 2B^T W \underline{\beta} + 2(B^T B + P) \underline{\varepsilon} .$$

Imponendo che le derivate siano uguali a zero si ha:

$$\begin{cases} W^T W \hat{\underline{\beta}} = W^T (\underline{z} - B \hat{\underline{\varepsilon}}) \\ (B^T B + P) \hat{\underline{\varepsilon}} = B^T (\underline{z} - W \hat{\underline{\beta}}) \end{cases} .$$

Queste due equazioni ricordano quelle usate per la regressione lineare e per il modello senza covariate, con la differenza che in questo caso a \underline{z} è sottratta, in entrambi i casi, la parte spiegata dal termine di modello a cui non si riferiscono $\hat{\underline{\beta}}$ e $\hat{\underline{\varepsilon}}$ rispettivamente.

A questo punto si possono ricavare le stime dei parametri. Per i coefficienti della funzione si trova:

$$\begin{aligned} \hat{\underline{\varepsilon}} &= [B^T B + P + B^T W (W^T W)^{-1} W^T B]^{-1} B^T [I - W (W^T W)^{-1} W^T] \underline{z} \\ &= A Q \underline{z} \end{aligned} \quad (2.19)$$

con $A = [B^T B + P + B^T W (W^T W)^{-1} W^T B]^{-1} B^T$ e $Q = [I - W (W^T W)^{-1} W^T]$, matrice molto importante nel caso di regressione lineare, poiché essa proietta il vettore dei dati nel sottospazio ortogonale allo spazio generato dalle colonne della matrice disegno, ricavando così il vettore dei residui. Questa matrice si ritrova anche in questo caso e sono valide le sue proprietà:

- Q è idempotente, cioè $QQ = Q$;
- Q è simmetrica;

- a causa del fatto che proietta nel sottospazio ortogonale di $\text{Col}(W)$, QW risulta essere la matrice nulla di opportune dimensioni.

Infine, la stima di $\hat{\beta}$ si ottiene dalla stima ottenuta per $\underline{\beta}$:

$$\hat{\underline{\beta}} = (W^T W)^{-1} W^T (I - BAQ) \underline{z} \quad (2.20)$$

In modo analogo al caso senza covariate, è necessario ricavare la *smoothing matrix*, poiché sarà utile in seguito:

$$\hat{z} = B\underline{\beta} + W\hat{\underline{\beta}} = [BAQ + W(W^T W)^{-1} W^T (I - BAQ)] \underline{z} = S\underline{z}.$$

2.2.1 Proprietà statistiche degli stimatori

Anche in questo caso è possibile calcolare valore atteso e varianza degli stimatori ottenuti. Questo è particolarmente utile in presenza di covariate, poiché consente di calcolare intervalli di confidenza o effettuare test per verificarne la significatività. Per farlo, però, è necessario avere la forma matriciale del modello indicato in 2.18:

$$\underline{z} = B\underline{\beta} + W\underline{\varepsilon} + \underline{\varepsilon}. \quad (2.21)$$

Di nuovo si ha

$$\mathbb{E}[\underline{\varepsilon}] = \underline{0} \quad \text{Var}[\underline{\varepsilon}] = \sigma^2 I_{nm}$$

e di conseguenza

$$\mathbb{E}[\underline{z}] = B\underline{\beta} + W\underline{\beta} \quad \text{Var}[\underline{z}] = \sigma^2 I_{nm}.$$

Mediate questo risultato e le proprietà ricavate per la matrice Q è possibile ottenere che:

$$\mathbb{E}[\hat{\underline{\beta}}] = AQB\underline{\beta} \quad \text{Var}[\hat{\underline{\beta}}] = \sigma^2 AQA^T.$$

Per $\hat{\beta}$ i calcoli sono più complessi, ma si semplificano grazie alle proprietà indicate in precedenza per la matrice Q . Si ritrova:

$$\mathbb{E}[\hat{\beta}] = \underline{\beta} + (W^T W)^{-1} W^T (I - BAB) \underline{\beta}$$

$$\text{Var}[\hat{\beta}] = \sigma^2 (W^T W)^{-1} + \sigma^2 (W^T W)^{-1} W^T BAQ A^T B^T W (W^T W)^{-1}.$$

Come nel caso senza covariate, anche ora si avrebbe la gaussianità degli stimatori se fosse ipotizzata per $\underline{\varepsilon}$. Come si può notare da quanto ricavato, lo stimatore per $\hat{\beta}$ è distorto.

2.3 Stima della varianza di modello e scelta dei parametri di smoothing

Quanto riportato di seguito è valido indipendentemente dal fatto che siano inserite nel modello le covariate, quindi per entrambi i modelli proposti in precedenza.

2.3.1 Varianza di modello

Stimare la varianza dell'errore è necessario se si vuole fare inferenza sugli stimatori ed è molto semplice se si conoscono i gradi di libertà equivalenti del modello. Ma questi si ricavano dalla *smoothing matrix*:

$$\text{EDF} = \text{tr}(S) .$$

Con questo valore si calcola la stima della varianza, usando i residui e il numero totale di dati:

$$\hat{\sigma}^2 = \frac{1}{nm - \text{tr}(S)} (\underline{z} - \hat{\underline{z}})^T (\underline{z} - \hat{\underline{z}})$$

2.3.2 Parametri di smoothing

I parametri λ_S e λ_T hanno un ruolo rilevante nella stima della soluzione, poichè scelgono quanto peso dare alla regolarità della funzione in spazio e tempo. Quindi è opportuno che siano fissati accuratamente prima della stima della soluzione.

Secondo quanto indicato in [4], la scelta corretta si ha con il valore di $\underline{\lambda}$ che realizza il minimo dell'indice di *generalized cross validation*

$$\text{GCV}(\underline{\lambda}) = \frac{nm}{nm - \text{tr}(S)} D(\hat{\underline{c}}, \hat{\underline{\beta}}) ,$$

dove D è la devianza del modello. Essa è così definita:

$$D(\hat{\underline{c}}, \hat{\underline{\beta}}) = 2\sigma^2(l_{\text{sat}} - l(\hat{\underline{c}}, \hat{\underline{\beta}})) ,$$

dove l è la logverosimiglianza del modello, che si ipotizza gaussiano, valutata rispettivamente nel suo massimo (valore di saturazione) e in corrispondenza dei valori stimati. Non è difficile dimostrare che, sia nel caso con covariate che senza covariate, si ha:

$$D(\hat{\underline{c}}, \hat{\underline{\beta}}) = (\underline{z} - \hat{\underline{z}})^T (\underline{z} - \hat{\underline{z}})$$

Di conseguenza, il miglior $\underline{\lambda}$ può essere scelto come valore che minimizza

$$\text{GCV}(\underline{\lambda}) = \frac{nm}{nm - \text{tr}(S)} (\underline{z} - \hat{\underline{z}})^T (\underline{z} - \hat{\underline{z}}) . \quad (2.22)$$

CAPITOLO 3

Sviluppo del codice R

Il modello descritto nel capitolo precedente è stato associato ad un algoritmo e ha portato allo sviluppo di un codice R per l'analisi dei dati. In questo capitolo saranno spiegate alcune delle scelte adottate o imposte durante la fase di sviluppo computazionale.

Il linguaggio di programmazione adottato è R. Questa decisione ha permesso di sfruttare le numerose funzioni statistiche che in altri linguaggi non sarebbero disponibili. Ma, come si potrà notare in seguito, sono anche emersi i lati negativi di questo linguaggio. Durante l'implementazione si è fatto uso del pacchetto *fda* e di alcune funzioni già disponibili per il caso puramente spaziale descritto in [6].

3.1 Funzioni di base considerate

Sono stati implementati solo alcuni tipi di funzione di base in spazio e tempo, che si sono rivelate utili alle applicazioni che saranno riportate nei capitoli successivi.

Per quanto riguarda lo spazio, il codice è stato pensato per dati distribuiti su un dominio dalla complessa definizione (con bordi irregolari o buchi). Quindi un'ottima scelta di funzioni di base sono gli elementi finiti descritti in [8], analogamente a quanto fatto in [6]. Queste funzioni sono definite su Ω_τ , triangolazione di Delaunay del dominio Ω , costruita con i vertici del poligono che descrive la frontiera e con i punti interni (solitamente quelli in cui sono disponibili i dati). Ognuno di questi punti diventa un nodo della triangolazione e ad ogni nodo è associata una funzione lineare a tratti come quella in 3.1(a), che vale 1 sul nodo selezionato, decresce linearmente sui triangoli adiacenti e si annulla su tutti

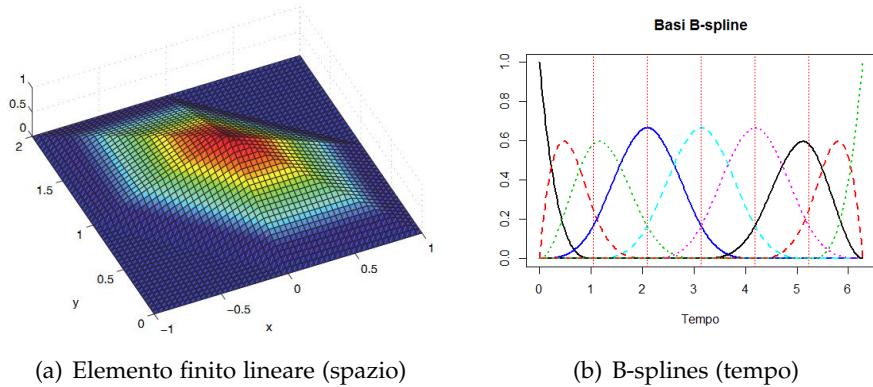


Figura 3.1: Funzioni di base implementate nel codice

gli altri nodi e triangoli. Per l'implementazione di queste funzioni è stato riutilizzata una parte del codice del caso puramente spaziale. Anche gli elementi finiti quadratici sono stati implementati, ma non sono stati scelti nelle applicazioni poiché rallentano l'esecuzione a causa della maggiore precisione che richiedono.

Le basi in tempo a disposizione sono le B-splines, funzioni che, fissato un intervallo temporale e alcuni nodi in esso contenuti (nel nostro caso equispaziati), sono usate per l'espansione in base delle *funzioni spline*. Il codice necessario è già stato implementato ed ottimizzato nel pacchetto *fda* ed è stato riciclato per il codice del modello ST-PDE. In fig. 3.1(b) sono riportate le 9 B-splines cubiche definite sull'intervallo $[0, 2\pi]$ (che sarà usato per l'applicazione al dominio a forma di C). L'unico vincolo imposto a queste funzioni di base è il grado, che deve essere almeno 3 per avere la derivata seconda delle basi necessaria per la penalizzazione.

3.2 Schematizzazione dell'algoritmo di stima

L'esecuzione del codice è stata divisa in alcuni passi per poter permettere all'utente di inserire gradualmente gli oggetti in ingresso e fissare i parametri che servono per ottenere la stima finale. In più punti si è cercato di nascondere all'utente le variabili temporanee e di includere in opportune classi di R tutte le informazioni riguardanti concetti tra di loro comuni.

Innanzitutto si ha l'acquisizione dei dati (e delle eventuali covariate) e la creazione della triangolazione. In questo caso non sono state implementate funzioni apposite poiché ci sono già pacchetti specializzati (nelle applicazioni che saranno presentate in seguito si è fatto uso di *RTriangle* per la triangolazione).

Dopo aver definito i punti, la triangolazione e gli istanti di tempo occorre fissare le funzioni di base. Sono state implementate due funzioni di R, una per lo spazio e una per il tempo, che in base ai parametri in ingresso e agli ordini scelti per le funzioni di base creano appositi oggetti che memorizzano tutte le informazioni necessarie.

A questo punto si ha la ricerca i migliori parametri di smoothing: definiti un insieme discreto di valori per λ_S ed uno analogo su λ_T , l'indice in eq. 2.22 è minimizzato sul prodotto cartesiano di tali insiemi. Per una corretta applicazione del modello questa operazione deve essere ripetuta più volte, fino al raggiungimento di una accettabile precisione nella stima di $\underline{\lambda}$. Sono state implementate due funzioni, una per il caso senza covariate e una per il caso con covariate, a causa della diversità della *smoothing matrix*. Questo è il passaggio più lento dell'algoritmo.

Dopo aver fissato i parametri di smoothing si arriva al cuore dell'algoritmo, poiché è calcolata la stima della soluzione. Ciò che eseguono le apposite funzioni è calcolare tutte le matrici di passaggio (P e B) nascondendole all'utente e risolvere gli appositi sistemi lineari (eq. 2.16, 2.19 e 2.20). In uscita è restituito un oggetto che racchiude i vettori ricavati e le funzioni di base, in modo da poter immagazzinare insieme tutti gli elementi necessari per tracciare grafici riguardanti la soluzione.

Terminata la stima, ciò che resta da fare è analizzare i risultati. Sono state create più funzioni, da scegliere in base a quello che si vuole visualizzare: calcolo della soluzione in punti ed istanti scelti, plot della funzione ad un istante fissato o ad un punto spaziale fissato, calcolo degli intervalli di confidenza approssimati (senza il termine di distorsione) per le componenti di β .

3.3 Strutture dati adottate

Al momento dell'implementazione del codice corrispondente alle eq. 2.22, 2.16, 2.19 e 2.20 è stato necessario decidere la corretta struttura dati per le matrici e una buona tecnica di inversione. In generale, quando si studiano dati distribuiti in spazio e tempo su domini complessi, le dimensioni delle matrici da invertire per calcolare $\hat{\beta}$ sono di grandi dimensioni. Inoltre, a causa delle funzioni di base scelte in precedenza, le matrici Ψ , Φ , P_S , R_0 e R_1 risulteranno sparse con molta facilità. Quindi la scelta di una struttura dati efficiente tra quelle disponibili in R è abbastanza delicata.

Sono stati analizzati quattro casi, in base al tipo di matrici (le matrici base di R o le sparse del pacchetto *Matrix*) o alla tecnica di inversione (classica di R o con fattorizzazione QR). Sono stati misurati i tempi di calcolo della stima con queste quattro modalità nell'applicazione del dominio a forma di C senza covariate (sarà discusso nel dettaglio nel prossimo capitolo) e in tab. 3.1 sono riportati i risultati. Sono stati eseguiti più tentativi,

al variare del numero di punti interni del dominio (e quindi della dimensione della matrice da invertire in 2.16), per poter controllare la velocità di esecuzione a difficoltà crescente.

Dimensione	Classico	Classico+QR	Sparse	Sparse+QR
1446	11.89	15.33	31.48	1508.63
2124	36.68	46.04	141.11	
3672	211.32	248.36	1369.21	
7086	1592.88	1840.64		

Tabella 3.1: Tempo di calcolo della stima di \hat{c} (in secondi) nelle simulazioni eseguite sul dominio a forma di C

Non sono state eseguite più misurazioni per caso perchè già da queste è chiaro quale sia la miglior scelta. L'uso delle matrici sparse è stato progressivamente abbandonato poichè nettamente più lento. Anche la fattorizzazione QR non ha portato ad un miglioramento, perciò è stato adottata l'inversione base di R.

Il motivo di questa lentezza è legato al linguaggio di programmazione. Non è una novità che R non sia uno dei linguaggi maggiormente efficienti. Inoltre, tra tutte le operazioni a disposizione, l'esecuzione dei cicli e di operazioni di accesso è un punto debole¹. Quindi l'uso di fattorizzazione QR (e della risoluzione di un sistema tramite *backward substitutions* che richiede) o di matrici sparse (implementate tramite tre vettori rispettivamente con indici di riga, colonna e valori non nulli) sono necessariamente lente per l'alto numero di cicli che richiedono. Le funzioni base di R, invece, sono certamente più ottimizzate. La conseguenza di questa analisi sarà riportata anche nel cap. 7, in cui si sottolinea che l'integrazione con linguaggi più efficienti nei colli di bottiglia del codice porterebbe a miglioramenti computazionali sicuri e alla possibilità di applicazione a dataset di grandi dimensioni.

¹si consultino le considerazioni in questa pagina: <http://adv-r.had.co.nz/Rcpp.html>

CAPITOLO 4

Simulazioni nel caso del dominio a forma di C

Prima di applicare il modello ST-PDE all'analisi della produzione di rifiuti nella provincia di Venezia sono state eseguite simulazioni su un caso noto e più semplice. Si è scelto di analizzare il dominio a forma di C e la corrispondente funzione spaziale $g(p)$ (riportata in fig. 4.1) descritti in [5], [6], [7] e implementata nel pacchetto R *mgcv*. La funzione $g(p)$ è solo spaziale, quindi è stata introdotta una variazione temporale deformando con il coseno:

$$f(\underline{p}, t) = g(\underline{p}) \cos(t)$$

Su questo semplice caso sono stati eseguiti i primi tentativi per il modello ST-PDE sia senza covariate che con una covariata generata.

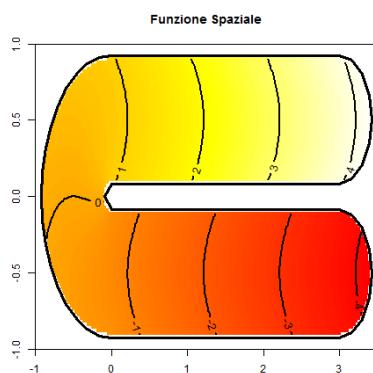


Figura 4.1: Funzione spaziale $g(\underline{p})$

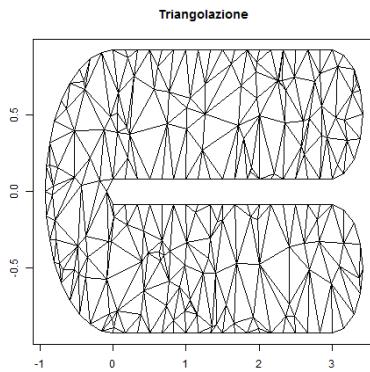


Figura 4.2: Triangolazione del dominio a forma di C

Dalla rappresentazione di $g(p)$ si può notare come sia fondamentale la geometria del problema, come per il caso della produzione dei rifiuti nella provincia di Venezia. Infatti, i massimi e i minimi si trovano sulle estremità del dominio, tra di loro vicine ma non collegate se non attraverso tutto il dominio.

4.1 Triangolazione e istanti temporali

In questo caso non sono presenti punti spaziali definiti dalla natura del problema (come possono essere i comuni per la provincia di Venezia), quindi è stato necessario ricavarli. Sono stati generati casualmente 150 punti all'interno del rettangolo $[-1, +3.5] \times [-1, +1]$ e di questi sono stati considerati validi solo quelli che ricadevano all'interno del dominio. Non è stata usata la descrizione della frontiera presente in *mgcv*, ma una versione diversa che permette di avere punti anche nella parte rettilinea del bordo. In fig. 4.2 è riportata la triangolazione di Delaunay ottenuta grazie al pacchetto R *RTriangle*. Come basi in spazio sono stati usati gli elementi finiti lineari definiti su questa triangolazione. In tutti gli esempi che seguiranno sarà considerata questa descrizione del dominio, che è formata da 241 punti (pari anche al numero di basi spaziali N). Di questi, 108 sono di frontiera e i restanti 133 corrispondono agli n punti sui quali saranno disponibili i dati.

Come intervallo temporale di variazione dei dati è stato scelto $[0, 2\pi]$ per sfruttare la periodicità del coseno. All'interno di questo intervallo sono stati ricavati 9 istanti temporali equidistanti tra di loro, quindi uno ogni $\frac{\pi}{4}$. Sono state fissate come basi in tempo le B-splines cubiche. Il numero di basi M è uguale al numero di istanti temporali a disposizione m , quindi 9.

4.2 Caso senza covariata

Nei punti e negli istanti temporali disponibili i dati sono stati ricavati dalla funzione esatta con l'aggiunta del rumore:

$$z_{ij} = g(\underline{p}_i) \cos(t_j) + \varepsilon_{ij} \quad \forall i \in 1 \dots n, \forall j \in 1 \dots m$$

dove

$$\varepsilon_{ij} \stackrel{\text{iid}}{\sim} N(0, 0.5^2) \quad \forall i \in 1 \dots n, \forall j \in 1 \dots m.$$

Per poter eseguire una analisi ottimale, come primo passo è necessario scegliere i valori dei parametri di smoothing ottimizzando l'indice GCV(λ) riportato in eq. 2.22. In queste analisi λ_S e λ_T saranno sempre espressi in potenze di 10. Per trovare dei buoni valori per i parametri sono stati eseguiti più tentativi, creando due insiemi discreti di variazione per $\log_{10} \lambda_S$ e $\log_{10} \lambda_T$ e minimizzando sui λ corrispondenti al prodotto cartesiano tra di essi. Il procedimento viene ripetuto qualche volta rendendo la griglia sempre più fitta. Dopo alcune iterazioni, è stato fissato $\lambda = (10^{-0.375}, 10^{-3.25})$ come valore definitivo per questa analisi ed è stata calcolata computazionalmente la stima.

In fig. 4.3 sono riportati i confronti tra funzione reale e stimata nei primi istanti di tempo (la scala di colori è stata resa uniforme tra tutti i grafici). Si può notare come la funzione stimata sia effettivamente molto simile a quella reale. La diversità tra le due estremità del dominio è stata colta, poiché il procedimento ha considerato correttamente la definizione della geometria spaziale.

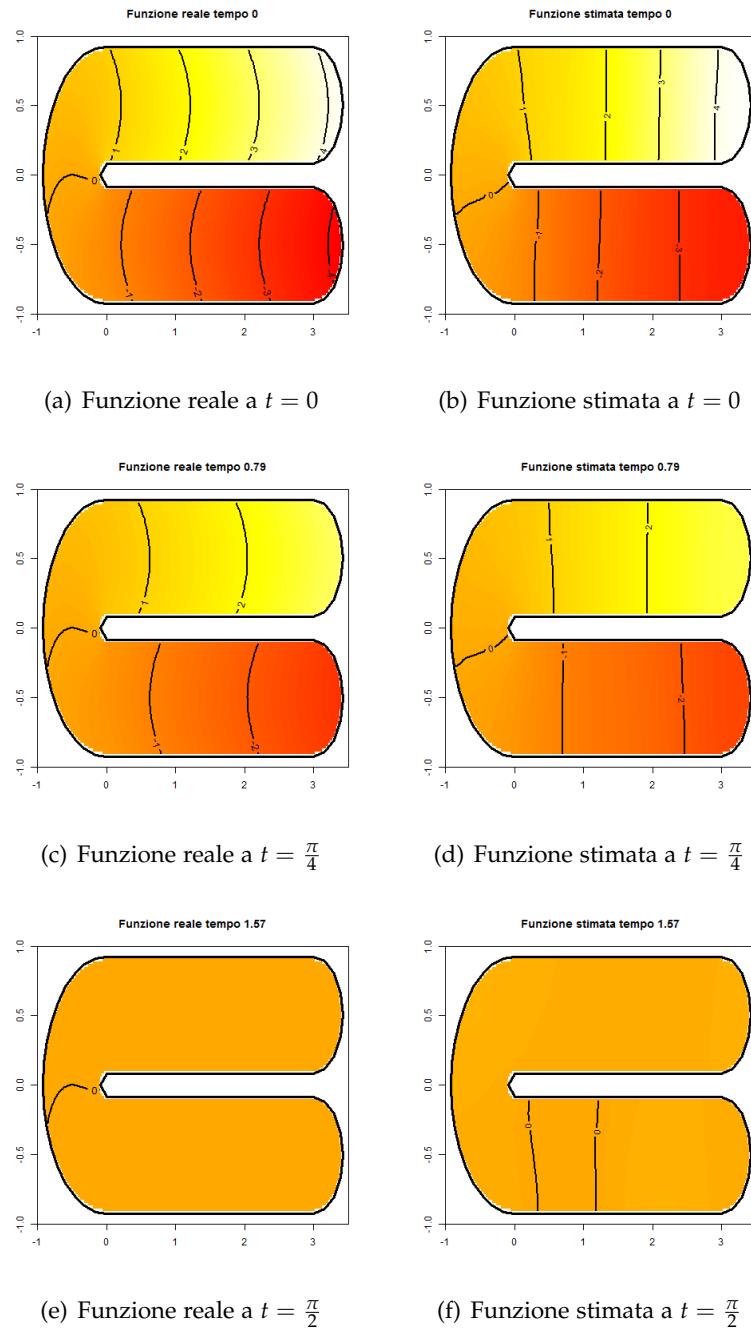


Figura 4.3: Stime della funzione $f(\underline{p}, t)$ ad alcuni istanti di tempo, caso senza covariata

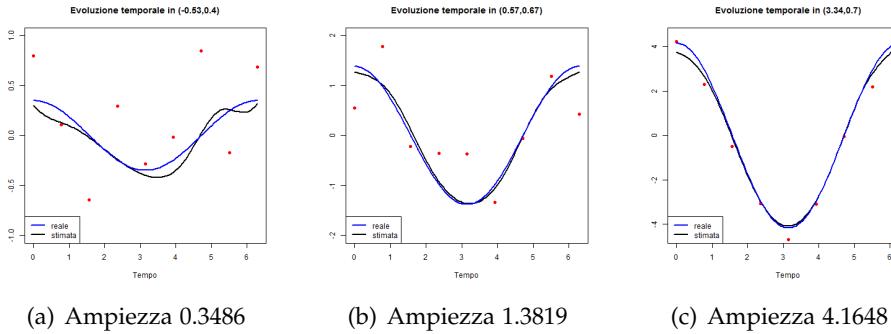


Figura 4.4: Evoluzione temporale in alcuni nodi della triangolazione, caso senza covariata

In fig. 4.4 si ha il confronto dell’evoluzione temporale in alcuni nodi della triangolazione. Oltre alla curva stimata è stata tracciata la reale, che è una cosinusoida di ampiezza nota (grazie alla perfetta conoscenza di $g(p)$) riportata con i grafici. I punti rossi corrispondono al dato generato (termine dovuto al rumore compreso). Tanto più è vicina a zero l’ampiezza della curva, tanto più il rumore influenza la stima poiché più rilevante (infatti, in fig. 4.3, le curve di livello della funzione stimata sono le più diverse da quelle reali quando la stima è vicina a zero). Tuttavia, anche nel caso di ampiezza vicina al valore massimo di $g(p)$ in fig. 4.4(c), l’andamento del fenomeno è stato riconosciuto e non è stata ottenuta una stima troppo interpolante dei dati. Si può concludere che la stima è molto buona, sebbene leggermente più confusa nella parte centrale del dominio.

4.3 Caso con covariata

Nel problema della stima della funzione $f(\underline{p}, t) = g(\underline{p})\cos(t)$ non sono presenti covariate. Quindi per poter testare il modello anche in questo caso, è stato necessario generare valori da assumere come covariata in ogni punto spaziale ed istante temporale in cui si ha la risposta. In definitiva i dati sono così formati:

$$z_{ij} = g(\underline{p}_i)\cos(t_j) + \beta w_{ij} + \varepsilon_{ij} \quad \forall i \in 1 \dots n, \forall j \in 1 \dots m$$

dove covariata e rumore sono generate da due normali tra loro indipendenti:

$$w_{ij} \stackrel{\text{iid}}{\sim} N(0, 1) \quad \forall i \in 1 \dots n, \forall j \in 1 \dots m$$

$$\varepsilon_{ij} \stackrel{\text{iid}}{\sim} N(0, 0.5^2) \quad \forall i \in 1 \dots n, \forall j \in 1 \dots m$$

e β è fissato a 1. Se il modello è buono, c'è da aspettarsi che la parte di funzione stimata senza covariata sia vicina a $f(\underline{p}, t)$ e che $\hat{\beta}$ si avvicini a 1.

Anche in questo caso è necessaria una analisi preliminare per fissare i valori di $\underline{\lambda}$ ottimizzando l'indice GCV($\underline{\lambda}$), che nel caso con covariata si differenzia dal precedente solo per la forma della *smoothing matrix*. Dopo alcuni tentativi su griglie discrete (esattamente come nel caso precedente) è stato ottenuto un minimo in corrispondenza di $\underline{\lambda} = (10^{-0.125}, 10^{-3.25})$.

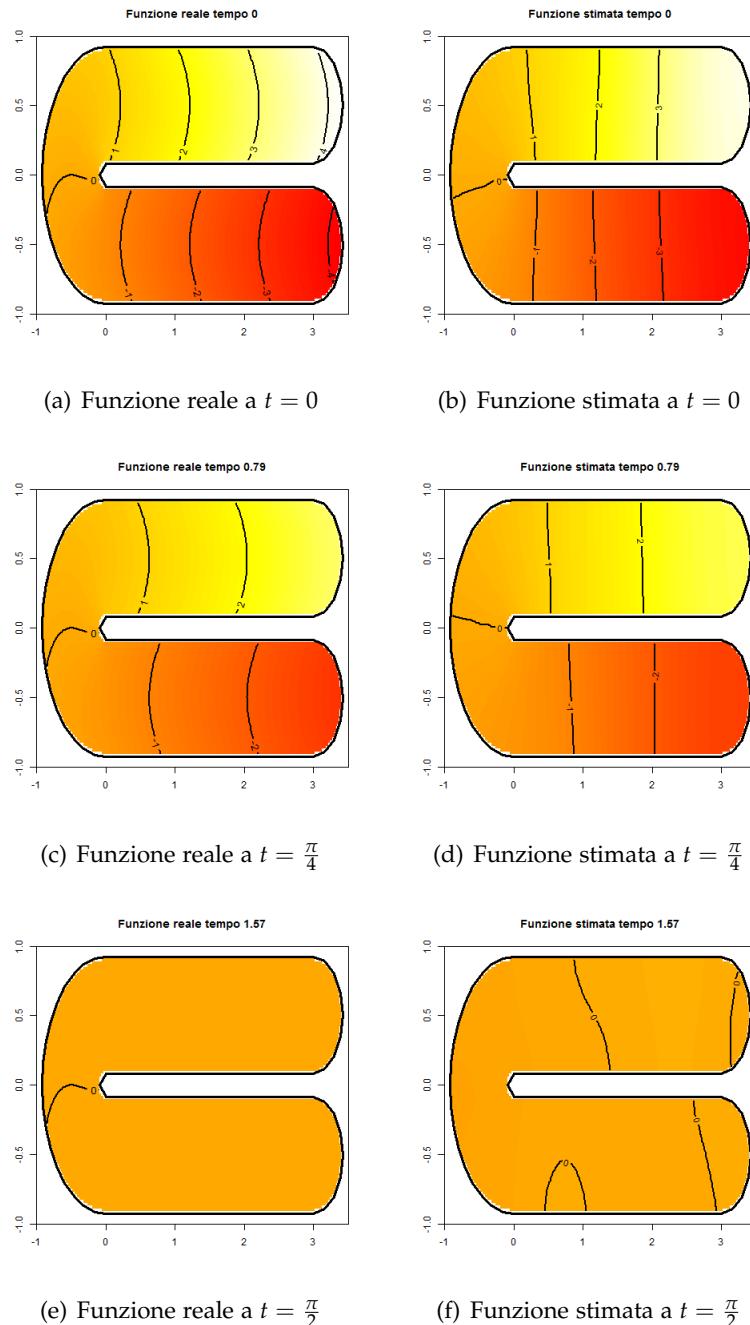


Figura 4.5: Stime della funzione $f(\underline{p}, t)$ ad alcuni istanti di tempo, caso con covariata

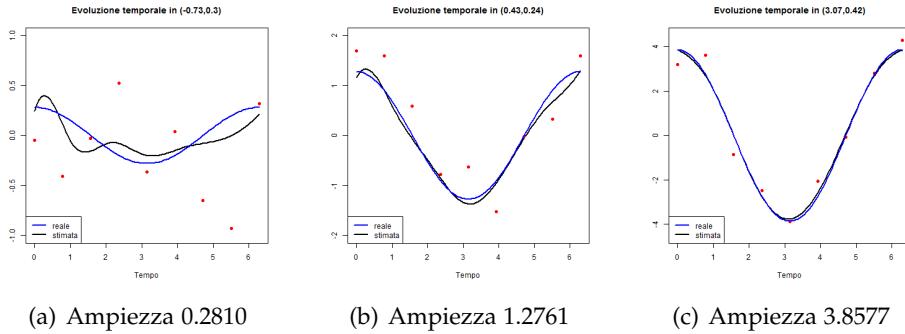


Figura 4.6: Evoluzione temporale in alcuni nodi della triangolazione, caso con covariata

In fig. 4.6, analogamente a quanto fatto nel caso senza covariata, si hanno i grafici dell'evoluzione temporale della funzione in alcuni punti spaziali. I punti rossi tracciati corrispondono alla parte di dato senza il termine dovuto alla covariata (ma con il rumore). Le conclusioni sono le stesse del caso senza covariata: la stima è ben riuscita e la vera variazione temporale è stata colta dal modello. Tuttavia, avvicinandosi alla parte centrale del dominio, si ha una maggiore influenza del rumore, in questo caso meno evidente a causa della presenza della covariata.

Dai grafici precedenti si può concludere che la stima della parte funzionale della risposta sia effettivamente una buona approssimazione della reale. Occorre verificare, però, anche che il contributo della covariata sia ben riconosciuto dal modello, controllando il valore stimato di β . Si ha:

$$\hat{\beta} \approx 1.005 ,$$

valore vicinissimo al reale. Pertanto è interessante validare questo risultato attraverso un test del tipo

$$\begin{cases} H_0 : \beta = 1 \\ H_1 : \beta \neq 1 \end{cases} ,$$

per poter verificare con una data significatività se il valore stimato corrisponde a quello reale. Per poter fare ciò si costruisce un intervallo di confidenza approssimato al 95% per β (eliminando il termine di distorsione) con quanto ricavato in sez. 2.2.1. Ne risulta:

$$\beta \in [0.9843; 1.0259]$$

che contiene 1. L'ipotesi H_0 può essere accettata.

Concludendo, la simulazione esposta in questo capitolo ha permesso di verificare, in un caso dalle caratteristiche totalmente note e con una forte dipendenza dalla geometria spaziale, l'appropriata stima del campo spazio-temporale e del termine con covariata. Grazie a questi risultati (e alla possibilità di conoscere i valori reali su tutto il dominio) si può dedurre che questo esempio sia adatto per il confronto delle stime del modello ST-PDE con le altre tecniche disponibili in letteratura.

CAPITOLO 5

Confronto con altri metodi

Il modello ST-PDE, come è già stato evidenziato nel capitolo 1, non è l'unico modello disponibile per l'analisi funzionale di dati distribuiti sia in spazio che in tempo. Pertanto è necessario che sia confrontato con le altre principali metodologie presenti in letteratura, al fine di poter dire se e quanto tale modello possa rappresentare un miglioramento in questo campo. I confronti saranno eseguiti nel caso della simulazione sul dominio a forma di C, poiché la conoscenza della funzione reale permette il calcolo dell'errore di stima.

Una tecnica da confrontare è senza dubbio il kriging spazio-temporale (KRIG). Le stime sono ottenute fissando un variogramma separabile e marginalmente esponenziale in spazio e tempo. I parametri del variogramma sono stimati dall'empirico e, successivamente, è possibile calcolare la stima grazie alle funzioni del pacchetto R *gstat*. Per alcuni problemi con l'uso del pacchetto il kriging non sarà analizzato nel caso con covariata. Come è già stato evidenziato precedentemente, in questa tecnica non è preso in considerazione il dominio spaziale. Pertanto ci si aspettano stime peggiori rispetto ai metodi che possiedono questa caratteristica.

Assieme al modello costruito, in [1] è possibile consultare il codice per il calcolo della stima, implementato nel pacchetto R *mgcv*. Per un corretto confronto con il modello ST-PDE sarà considerato un solo termine funzionale per la risposta, costituito dal prodotto tensoriale dei modelli marginali in spazio e tempo (definiti dalle funzioni di base e dai termini di penalizzazione differenziali, analoghi a quelli proposti per ST-PDE). Saranno sempre considerate come basi in tempo le *Regression splines* cubiche, mentre in spazio saranno

analizzati due casi:

- basi spaziali per *thin plate splines*, riportate in [16], che sono basi radiali non in grado di considerare la geometria spaziale (TPS);
- basi spaziali per *soap film smoothing*, analizzate nel dettaglio in [7] e utilizzate in [4] e [1] e costituite da due insiemi di funzioni (rispettivamente per la parte interna del dominio e per la frontiera) appositamente pensate per casi con dominio spaziale complesso (SOAP).

La triangolazione e i dati sono gli stessi che sono stati usati nel capitolo 4. In aggiunta è stata costruita una griglia spazio-temporale di punti per la validazione, attraverso l'uso di 80 punti equispaziati in $[-1, +3.5]$, 40 punti in equispaziati $[-1, +1]$ e 20 istanti di tempo equispaziati in $[0, 2\pi]$. Ovviamente l'errore è stato valutato soltanto sui punti che ricadevano all'interno del dominio a forma di C.

I modelli sono stati confrontati attraverso il Root Mean Square Error (RMSE) calcolato nei punti di validazione. Quindi se se V è l'insieme dei punti della griglia interni al dominio, fissato un modello si avrà:

$$\text{RMSE}_V = \sqrt{\frac{\sum_{(p_i, t_i) \in V} (\hat{z}_{ij} - g(\underline{p}_i) \cos(t_i))^2}{\text{card}(V)}}$$

Il procedimento è stato iterato 50 volte, per poter escludere possibili effetti particolari dovuti alla generazione del rumore.

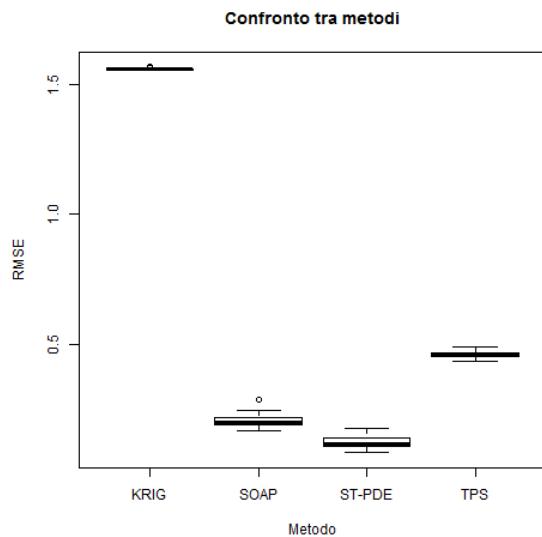


Figura 5.1: Confronto tra i metodi, caso senza covariata

5.1 Caso senza covariata

Nel caso senza covariata si hanno i risultati riportati in fig. 5.1, in cui sono stati tracciati i boxplot dei valori di RMSE ottenuti nelle 50 iterazioni per ogni metodo. Subito si nota che l'errore commesso è minore nel caso di ST-PDE, e quindi la stima ottenuta con il modello proposto è la migliore.

I risultati riflettono quanto è stato ipotizzato in precedenza: KRIG e TPS, che non considerano la geometria del problema, hanno prodotto gli errori più alti. Invece SOAP e ST-PDE commettono errori minori, ma tra i due il migliore è ST-PDE. Come ulteriore conferma di tutto ciò, in fig. 5.2 sono riportate le stime ottenute dai differenti metodi per i primi istanti di tempo.

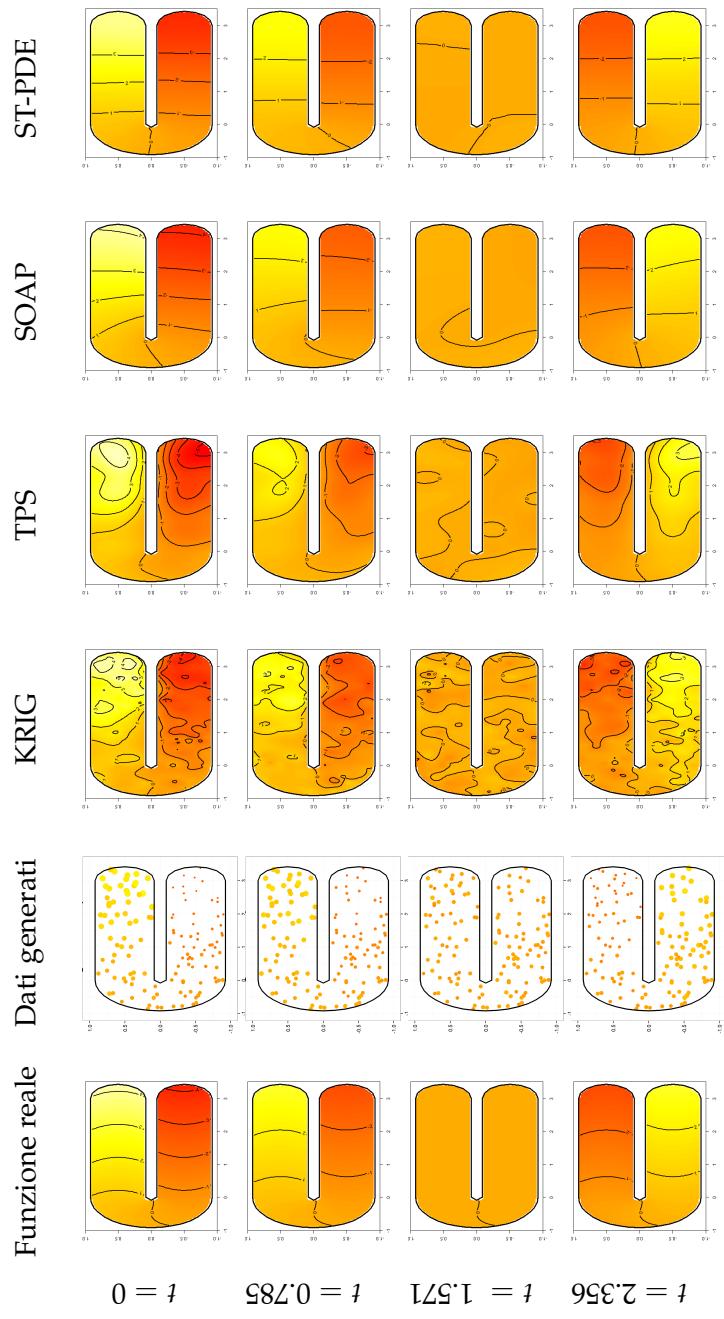


Figura 5.2: Per alcuni istanti di tempo, funzione test $f(\underline{p}, t)$ reale, dati simulati, stime ottenute rispettivamente con kriging spazio-temporale, soap film smoothing, thin plate splines e ST-PDE, caso senza covariata

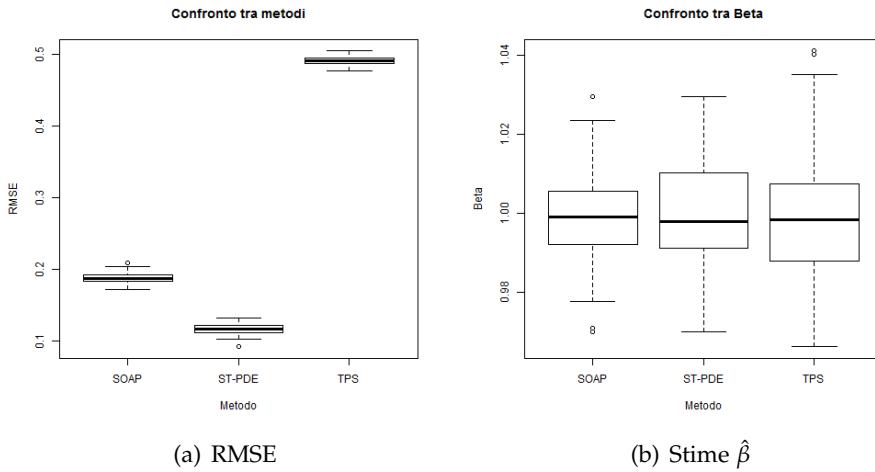


Figura 5.3: Confronto tra i metodi, caso con covariata

5.2 Caso con covariata

La stessa analisi è stata eseguita nel caso con covariata, generata in tutti i punti esattamente come fatto nel capitolo 4. Nella calcolo del RMSE è stato considerato solo la parte di risposta spiegata dalla funzione $f(p, t)$, poiché non è opportuno generare nuovamente valori per la covariata nei punti di validazione. I boxplot riportati in fig. 5.3(a) possono quindi essere considerati come valutazione della bontà della stima della parte funzionale del modello. Per confrontare la parte spiegata dalla covariata sono stati tracciati i boxplot in fig. 5.3(b), con le stime di $\hat{\beta}$ calcolate durante le 50 iterazioni. Come è già stato riportato in precedenza, il kriging non è stato analizzato per alcuni problemi riguardanti l'uso del pacchetto *gstat* in presenza di covariate.

Le conclusioni sono perfettamente analoghe al caso precedente. La stima di β non presenta differenze, ma nella parte funzionale il caso ST-PDE è nuovamente il migliore. Dai plot della funzione stimata ad alcuni istanti di tempo fissati in fig. 5.4 si possono trarre le stesse conclusioni dal caso senza covariata. Il modello ST-PDE è quello che più si avvicina alla funzione reale.

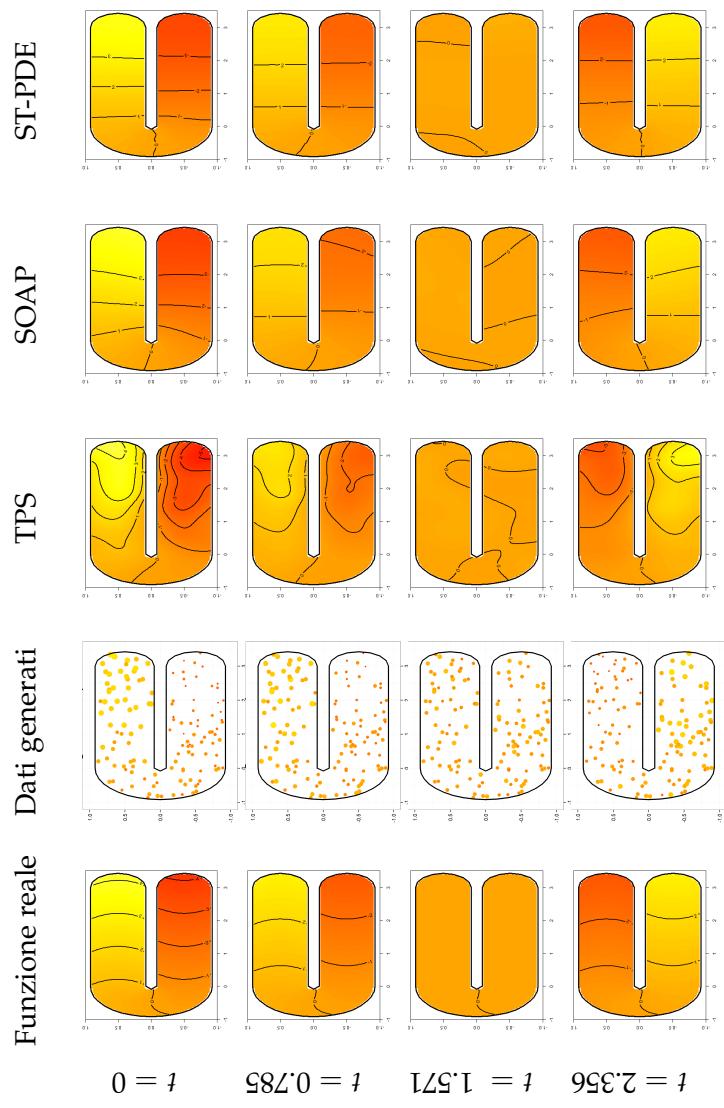


Figura 5.4: Per alcuni istanti di tempo, funzione test $f(p, t)$ reale, dati simulati, stime ottenute rispettivamente con soap film smoothing, thin plate splines e ST-PDE, caso con covariata

CAPITOLO 6

Applicazione alla produzione di rifiuti urbani nella provincia di Venezia

L'applicazione principale per il modello ST-PDE in questo lavoro di tesi riguarda i dati della produzione di rifiuti urbani nel periodo di anni dal 1997 al 2011 nella provincia di Venezia. Per rifiuti urbani si intendono rifiuti domestici, rifiuti prodotti in locali, aree pubbliche, parchi, giardini o spiagge, rifiuti provenienti dalla pulizia delle strade o di altri luoghi pubblici. Non sono conteggiati i rifiuti speciali (tra cui ad esempio quelli industriali, agricoli o provenienti da attività commerciali o di costruzione) o pericolosi (per i quali esistono programmi di smaltimento particolari).

In realtà i dati raccolti dall'Agenzia Regionale per la Prevenzione e Protezione Ambientale del Veneto (Arpav) e condivisi sul sito Open Data Veneto riguardano tutta la regione. Tuttavia è stata analizzata solo la provincia di Venezia per due motivi. Innanzitutto l'interesse per la zona della laguna veneta e per le particolarità del dominio spaziale da cui essa è descritta. Inoltre considerare tutto il Veneto aumenterebbe notevolmente le dimensioni delle matrici in gioco causando una grossa spesa computazionale per la ricerca della soluzione. Quindi è stato scelto di concentrarsi su un dominio più piccolo ma nel quale è possibile notare più facilmente le caratteristiche dell'andamento del fenomeno della produzione dei rifiuti urbani e le proprietà di stima del modello.

Per ogni comune della provincia di Venezia e per ogni anno è disponibile il numero di rifiuti totali raccolti in tonnellate e la popolazione residente. La popolazione è certamente un valore influente per la produzione di rifiuti, perciò la quantità di riferimento non sarà il valore dei rifiuti totali raccolti in ogni anno per comune, ma il valore pro capite.

Le coordinate spaziali dei comuni sono la longitudine e la latitudine, disponibili on line¹. Nel caso dei comuni con dato replicato (sez. 6.2.2), le coordinate sono state ottenute da Google Maps.

La più importante ipotesi iniziale in questa applicazione è la scelta di riferire i dati riguardanti la raccolta dei rifiuti di tutto il territorio comunale in un unico punto, corrispondente al capoluogo. Potrebbe essere più appropriato riferire i dati all'intero territorio comunale o alla zona urbanizzata del comune (non su tutto il territorio sono prodotti rifiuti urbani). Ma per semplicità lo studio è eseguito con dati puntiformi normalizzati rispetto alla popolazione.

6.1 Il turismo come possibile covariata

L'inclusione della popolazione residente nella risposta tramite la scelta di usare i valori pro capite è necessaria, poiché permette di depurare la risposta da una variabile che per sua natura la influenzerebbe. Ma sarebbe un errore fermarsi solo alla popolazione residente, poiché anche i turisti rappresentano una componente non trascurabile di produzione di rifiuti urbani.

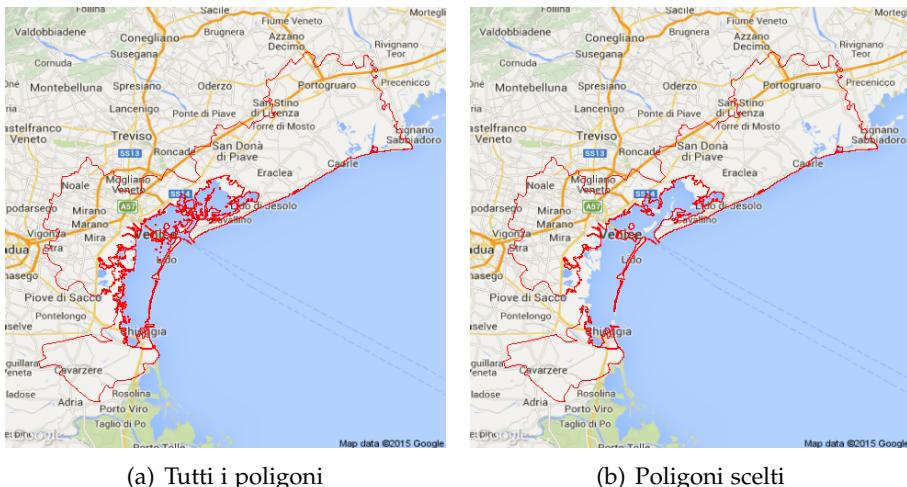
Nella provincia di Venezia sono presenti molte zone di elevata attrazione turistica. Grande importanza è da attribuire a Venezia, ma si hanno anche zone balneari (come Lido di Venezia, Cavallino-Treporti, Jesolo, San Michele al Tagliamento, Bibione, ecc...). L'informazione scelta per sintetizzare l'attività turistica è il numero di posti letto totali del comune, valore disponibile grazie all'applicativo dell'Istat *Atlante Statistico dei Comuni*² per ogni anno. Il totale dei posti letto per comune è la somma di vari tipi di attività non solamente alberghiere in senso stretto (ad esempio sono conteggiati anche esercizi complementari, bed & breakfast, campeggi) e sarà considerato normalizzato per la popolazione residente per uniformità con la risposta. I valori ricavati saranno inseriti nel modello come possibile covariata.

6.2 Trattamento del dominio

Per poter studiare il problema a livello computazionale occorre avere una buona approssimazione della frontiera della regione. Questa è disponibile nel pacchetto R *raster* che descrive dati geografici di moltissime zone del mondo sia a livello nazionale che locale (nel caso italiano province e comuni) tramite poligoni molto precisi.

¹<http://www.dossier.net/utilities/coordinate-geografiche/>

²<http://www.istat.it/it/archivio/113712>

Figura 6.1: Poligoni disponibili nel pacchetto *raster*

Una volta scaricata la provincia di Venezia da *raster*, si è riscontrato subito un problema: la regione è composta da un insieme di 101 poligoni distinti (a causa delle numerose isole di cui è composta la laguna) e ogni poligono ha un alto numero di vertici (ad esempio, la prima delle due regioni corrispondenti alla parte interna della provincia ha 10538 vertici). Non è possibile analizzare il problema su un territorio così descritto, perciò è stata necessaria una analisi iniziale della frontiera per ridurne la complessità.

Oltre all'entroterra (composto da due poligoni) saranno considerate solo le più più rilevanti isole della laguna a livello di popolazione e turismo: Venezia, Murano, Lido di Venezia, Pellestrina e Chioggia (composta da due poligoni). In fig. 6.1 (tracciata, come per le successive, con il pacchetto R *RgoogleMaps*) sono visualizzati i poligoni disponibili in *raster* e quelli scelti per l'analisi. Come sarà indicato nel paragrafo successivo, tutte queste isole sono state trattate per la riduzione dei vertici e, per creare un poligono unico, sono state unite tra loro con ponti dove era possibile. Tra le isole collegate solamente via mare con il resto del territorio sono stati simulati ponti in corrispondenza delle trafficate linee di trasporto pubblico con traghetto.

6.2.1 Regression splines

Per ridurre l'elevato numero di vertici di ognuno dei poligoni considerati (e la conseguente complessità della stima) è stato necessario ricorrere a tecniche di smoothing di dati funzionali. Ad ogni poligono è possibile associare una coppia di funzioni: la latitudine e la longitudine rispet-

to all'ascissa curvilinea (disponibili per punti, corrispondenti ai vertici di *raster*) da rappresentare in funzioni di base. Successivamente, le nuove funzioni stimate sono state valutate in un numero molto inferiore di valori dell'ascissa curvilinea. Con le nuove coordinate ottenute dopo questo procedimento è stata costruita la nuova definizione della regione.

Per avere una rappresentazione ottimale tramite funzioni di base sono state provate più tecniche. La scelta definitiva è ricaduta sull'uso di *Regression Splines* cubiche senza penalizzazione della derivata seconda. Infatti i risultati non sono stati migliori con le altre tecniche provate a causa della zona interna alla laguna di Venezia, fortemente frastagliata: ad esempio penalizzare la derivata seconda ha eliminato troppe asperità presenti sulle coste del territorio, mentre l'utilizzo di *Kernel Smoothing* ha portato alla definizione di regioni che, dopo la triangolazione, presentavano un maggior numero di triangoli composti solamente da punti di frontiera (e quindi senza dati) rispetto agli altri metodi.

Una volta fissato un ragionevole numero di punti con cui descrivere la regione sono stati eseguiti più tentativi per decidere il miglior numero di funzioni di base. Il criterio di scelta è stato complesso, poiché sono stati esclusi i valori che generavano intersezioni nella nuova descrizione della regione e comuni esterni alla frontiera. Il miglior valore per *Regression Splines* è stato quello che, una volta eseguito lo smoothing della regione, ha causato la minor distanza tra i nuovi punti della regione e il poligono iniziale di *raster*. In fig. 6.2 è riportato il risultato dello smoothing sul primo poligono che descrive l'entroterra della provincia di Venezia (nella versione definitiva ha 100 punti, molti meno dei 10538 iniziali).

Dopo aver ripetuto l'analisi per ognuna delle isole elencate precedentemente (eccetto Chioggia, aggregata al dominio in modo molto più semplificato), la descrizione finale è stata ricavata unendo tra loro tutti i nuovi poligoni. In seguito è stata eliminata una zona costiera dell'entroterra della laguna di Venezia che, sebbene presente sia in *raster* che nei grafici di Google Maps, corrisponde ad una parte fangosa e paludosa e quindi disabitata. Non essendo possibile che su di essa siano prodotti rifiuti, è stata tagliata dalla regione. Per questo motivo si troverà sempre una zona non analizzata sui grafici con mappe di Google Maps. In fig. 6.3 è riportata la descrizione finale del dominio con i punti spaziali considerati.

6.2.2 Replica dei dati

L'uso di valori pro capite per rifiuti e posti letto consente di replicare il dato del comune anche su altri punti in cui risulta necessario. Ad esempio, le isole di Murano, Lido di Venezia e Pellestrina non sono sedi di comune, ma si riferiscono a Venezia. Quindi il dato di Venezia è stato replicato in queste isole ad ogni anno, per avere un valore di riferimento anche nelle isole. I dati replicati dal comune di Venezia sono visualizzati in

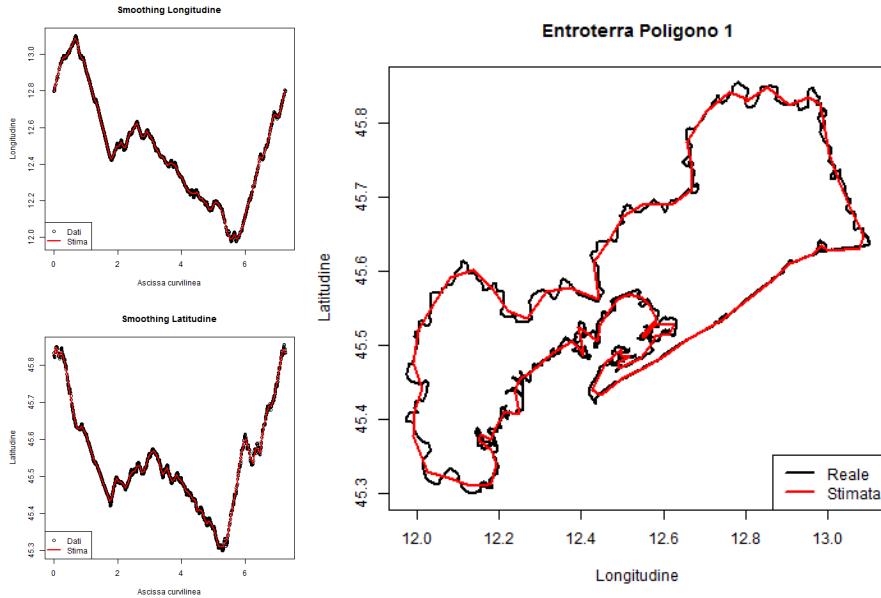


Figura 6.2: Smoothing con *Regression Splines* cubiche per il primo poligono dell'entroterra della provincia di Venezia

rosso in fig. 6.4(a). Come si può notare, anche nell'isola di Venezia è stato duplicato il dato, per poter ottenere una triangolazione senza troppi triangoli composti solo da punti di frontiera in una zona di particolare rilevanza.

Un caso particolare riguarda il comune di Cavallino-Treporti, che è stato istituito nel 1999 con una parte dei territori del comune di Venezia. La separazione all'interno dei dati, però, è presente dal 2002. Di conseguenza prima di questo anno il dato in Cavallino-Treporti è una replica del dato di Venezia. Cavallino-Treporti corrisponde al punto blu in fig. 6.4(b).

Un altro punto di replica del dato è Bibione, località balneare vicina a Lignano Sabbiadoro (che però non è in Veneto) ed interna alla regione (punto verde in fig. 6.4(b)). Il comune di riferimento di Bibione (San Michele al Tagliamento) è lontano, pertanto si genera una parte di territorio senza dati molto ampia nell'estremità orientale della provincia. Quindi la replica della misurazione in Bibione consente di avere un dato in una zona rilevante (già dai grafici in fig. 6.6 e 6.7 si potrà capire l'importanza della zona balneare in questo studio) e di coprire anche un'estremità del dominio altrimenti priva di riferimenti.

6.2.3 Triangolazione del dominio

La triangolazione è stata prodotta tramite il pacchetto R *RTriangle*. Poiché nella zona ad est il numero di capoluoghi di comune (e quindi di nodi

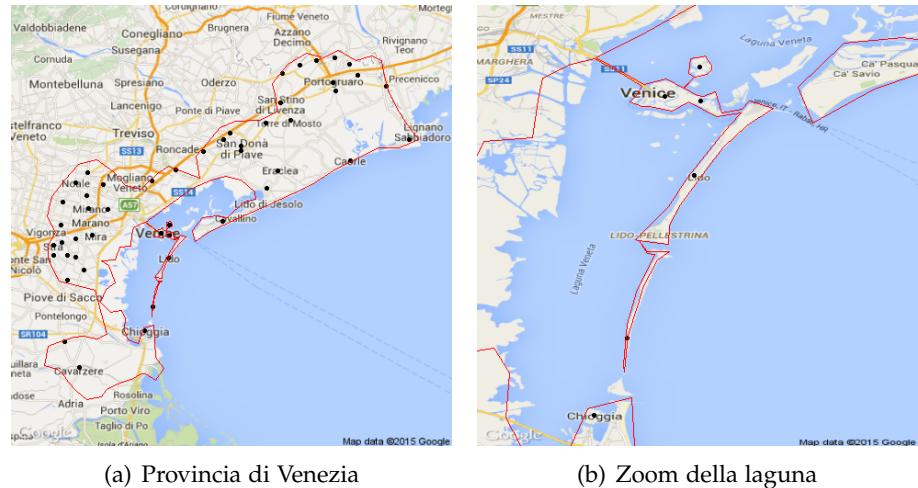


Figura 6.3: Frontiera e punti spaziali per la provincia di Venezia

della triangolazione) è minore rispetto al resto della regione, è stata fissata un'area massima per i triangoli generati da *RTriangle*. Questo ha reso la triangolazione più fitta anche dove non lo sarebbe stata, e garantirà una stima della risposta più precisa nella zona balneare che, come si potrà notare in seguito, ha una grande importanza per la distribuzione dei rifiuti. Affinchè questo sia possibile sono stati aggiunti nuovi punti spaziali all'interno della regione, che resteranno senza dato per tutta l'analisi ma saranno utili per la definizione dei triangoli. In fig. 6.5 si ha la triangolazione finale che sarà usata da ora in avanti.

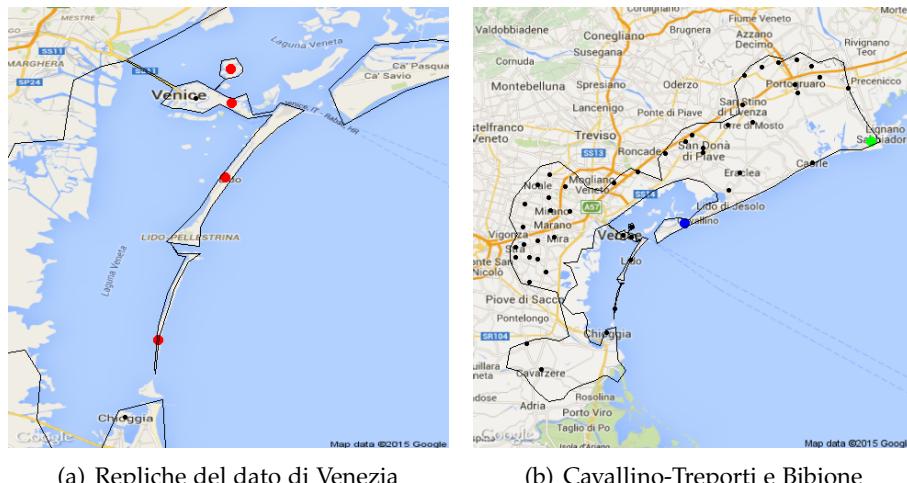
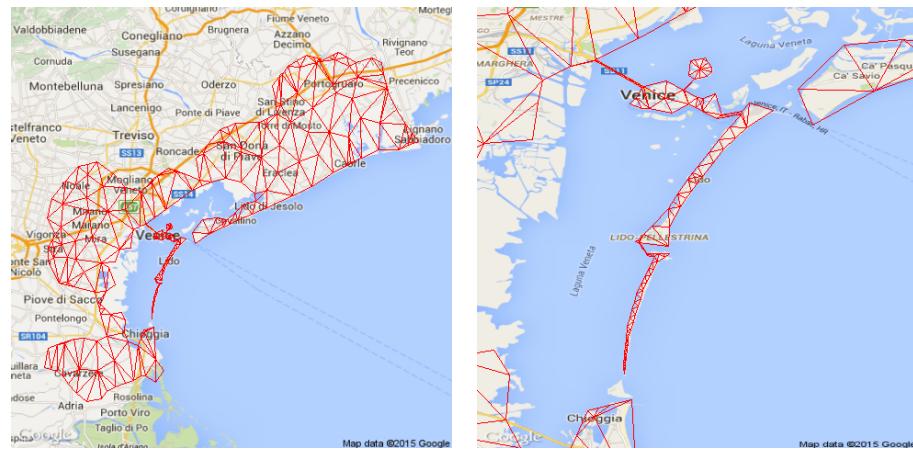


Figura 6.4: Dati replicati

In conclusione, il dominio è descritto da 418 punti (49 con dati, 369 di frontiera o aggiunti durante la creazione della triangolazione) e da 480 triangoli.

6.3 Analisi preliminare dei dati

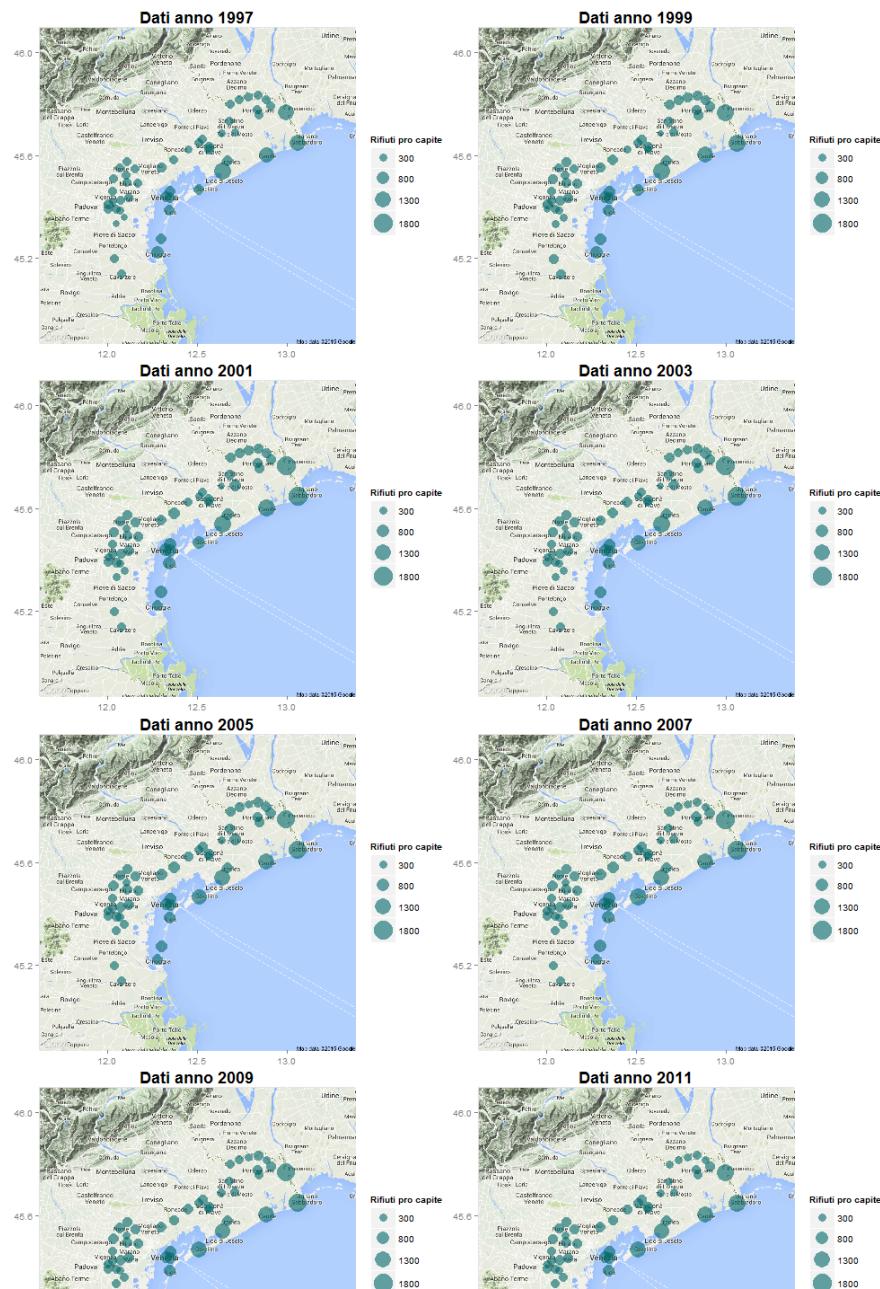
Prima di eseguire l'analisi, è opportuno avere una visualizzazione dei dati sul dominio della provincia di Venezia. In fig. 6.6 e 6.7 sono visualizzati, rispettivamente, i valori dei rifiuti e dei posti letto pro capite (realizzati grazie all'aiuto del pacchetto R *ggmap*), che permettono di avere un'idea della loro distribuzione in spazio e tempo.



(a) Provincia di Venezia

(b) Zoom della laguna

Figura 6.5: Triangolazione della provincia di Venezia



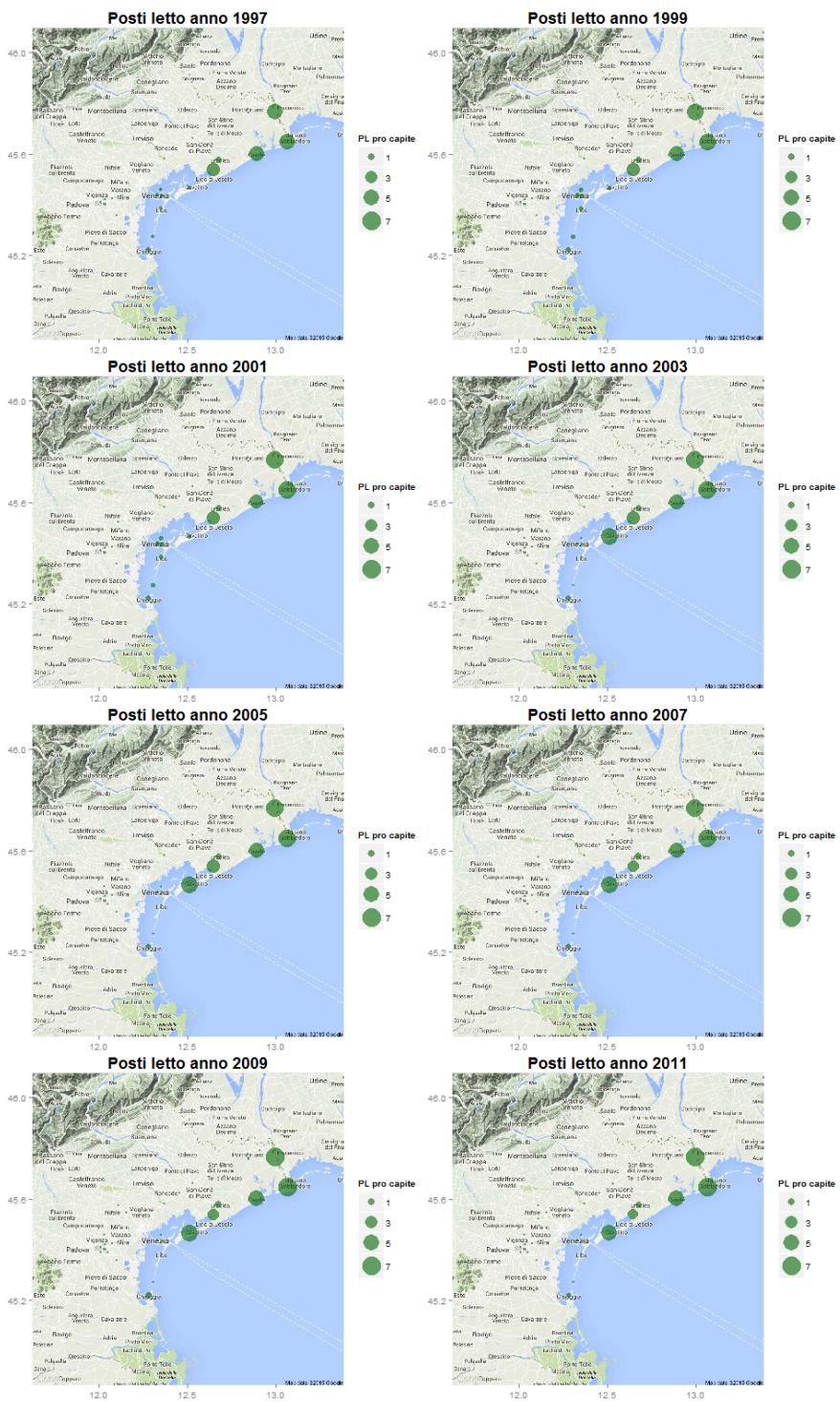


Figura 6.7: Posti letto pro capite ogni due anni dal 1997 al 2011

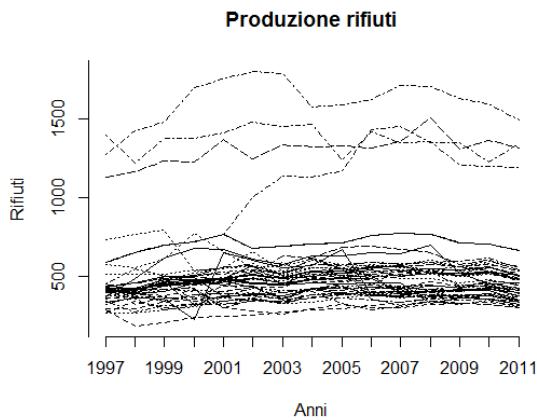


Figura 6.8: Andamento temporale della produzione dei rifiuti urbani nei comuni

Da questa analisi iniziale si può già notare come la produzione di rifiuti sia più alta nella zona balneare della regione e come non presenti grosse variazioni nel tempo in quasi tutti i comuni, poiché i dati sono più o meno simili negli anni. Questo è evidenziato anche in fig. 6.8, dove sono stati collegati tra loro i dati al variare del tempo per ogni comune. Si distinguono subito due gruppi (valori più alti per la zona balneare e più bassi negli altri comuni) totalmente distinti, eccetto per un comune (Cavallino-Treporti) che subisce un forte innalzamento nel tempo.

Dai grafici in fig. 6.7 per il numero di posti letto pro capite si può notare che la covariata assume valori decisamente più alti nella zona balneare rispetto al resto della regione. Contrariamente a quello che si potrebbe pensare, non si ha un elevato valore di posti letto nell'isola di Venezia. Invece si ha una grossa variazione nel comune di Cavallino-Treporti dopo la separazione dal comune di Venezia (si nota che il raggio della bolla aumenta notevolmente dopo il 2002). Questo indica che i posti letto sono in realtà più densi in Cavallino-Treporti, ma quando il dato era unito al comune di Venezia questo effetto non poteva essere colto in modo così dettagliato. Ci si può quindi aspettare che la funzione stimata presenti un comportamento differente in Cavallino-Treporti nelle due situazioni.

6.3.1 Funzioni di base

Le basi in spazio scelte per l'applicazione del modello sono gli elementi finiti lineari anche in questo caso. Ad ognuno dei punti spaziali (interni o di frontiera) è associata una funzione di base, coerentemente con la triangolazione prodotta. Di conseguenza si ha $N = 418$ mentre il numero di punti con dati n è minore ed è pari a 49.

In tempo, esattamente come nel caso del dominio a forma di C, saranno adottate come funzioni di base le B-splines cubiche. L'intervallo temporale per la descrizione del dominio è [1997, 2011] e i dati sono disponibili con cadenza annuale. Anche in questo caso si assume che il numero di basi sia pari al numero di istanti temporali a disposizione, quindi $M = m = 15$.

6.4 Applicazione del modello senza covariata

Prima di calcolare i risultati dell'analisi occorre fissare i parametri λ_S e λ_T . Il procedimento è perfettamente analogo a quello ricavato nel caso del dominio a forma di C per ricavare una buona approssimazione del minimo di $GCV(\lambda)$ in eq. 2.22. A seguito di alcuni tentativi su griglie discrete, la produzione dei rifiuti è stata analizzata con $\lambda = (10^{-9}, 10^{-0.125})$. In fig. 6.10 sono riportati i risultati ottenuti nei 15 anni a disposizione. Questi grafici permettono di studiare il profilo della funzione nei vari istanti temporali in modo molto accurato, perché la scala cromatica è stata resa uniforme tra tutti gli anni.

Successivamente sarà rappresentata anche la stima dell'evoluzione temporale della funzione in alcuni comuni, riportati in fig. 6.9.

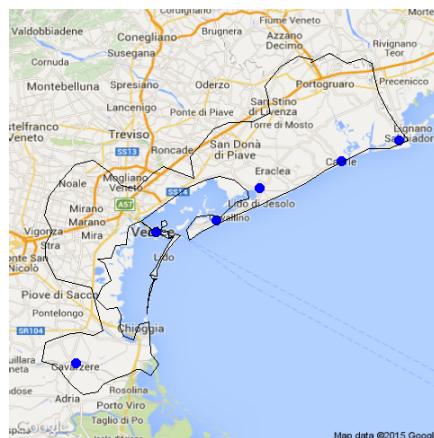
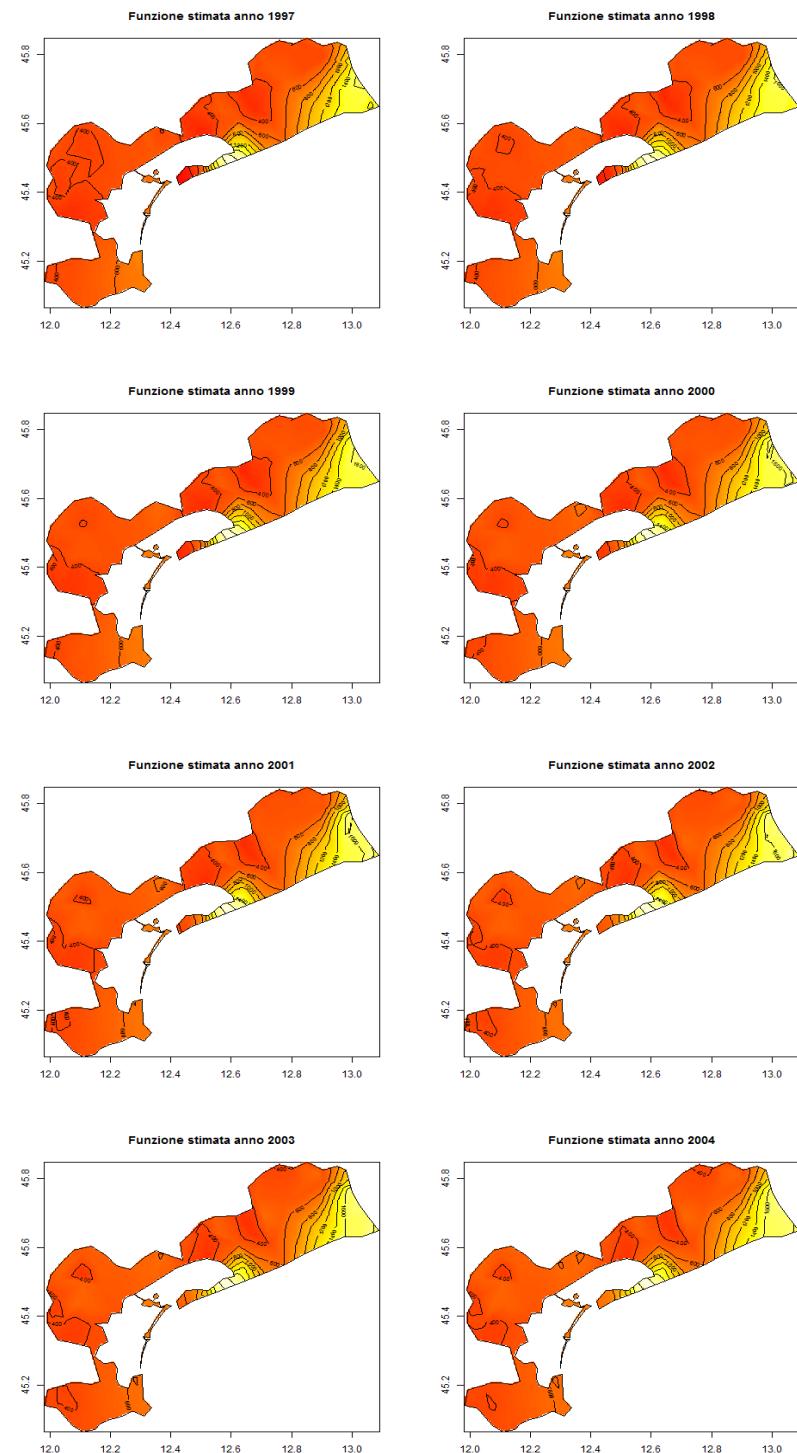


Figura 6.9: Comuni selezionati per l'analisi dei risultati



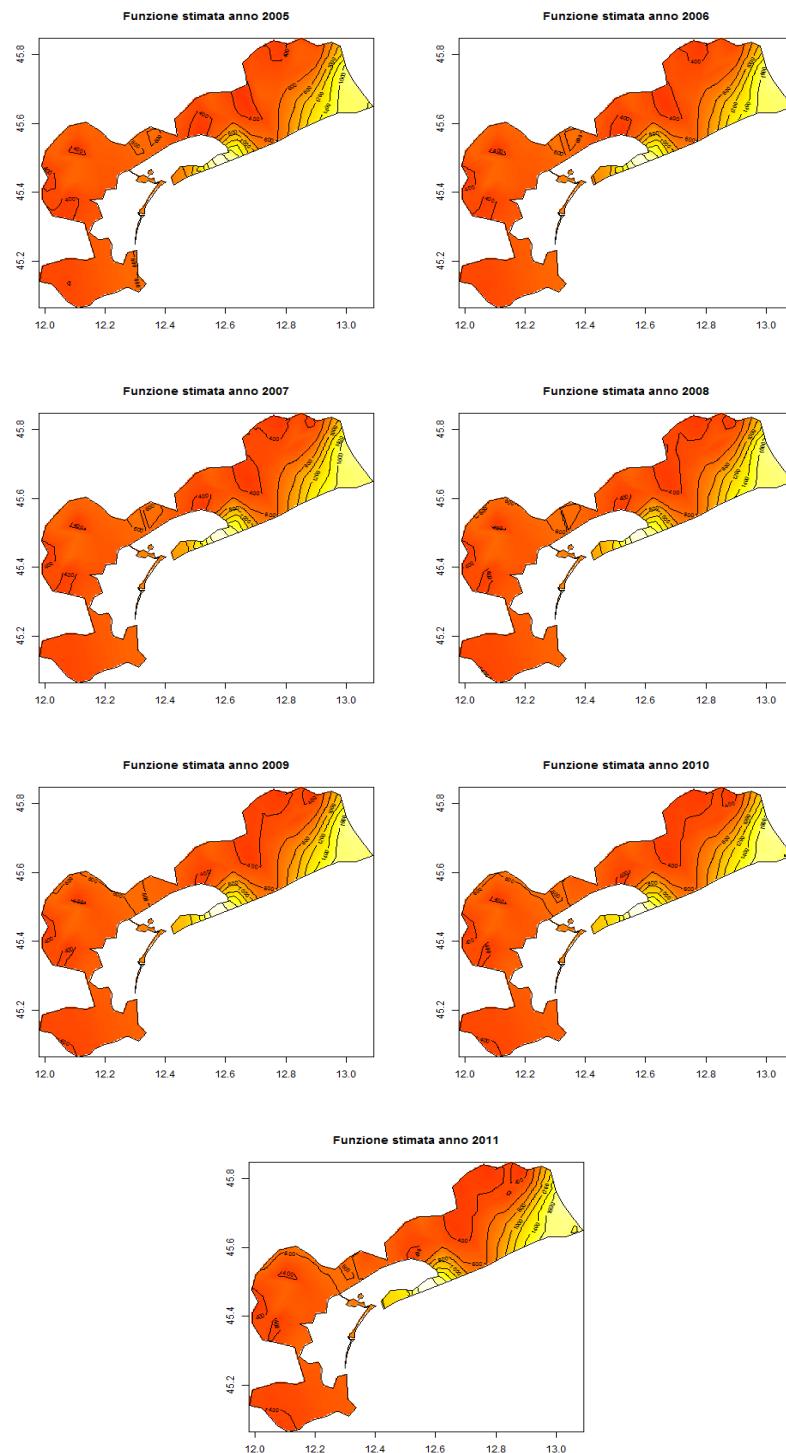


Figura 6.10: Stima della funzione spazio-temporale della produzione di rifiuti urbani pro capite nella provincia di Venezia a tempi fissati dal 1997 al 2011, caso senza covariata

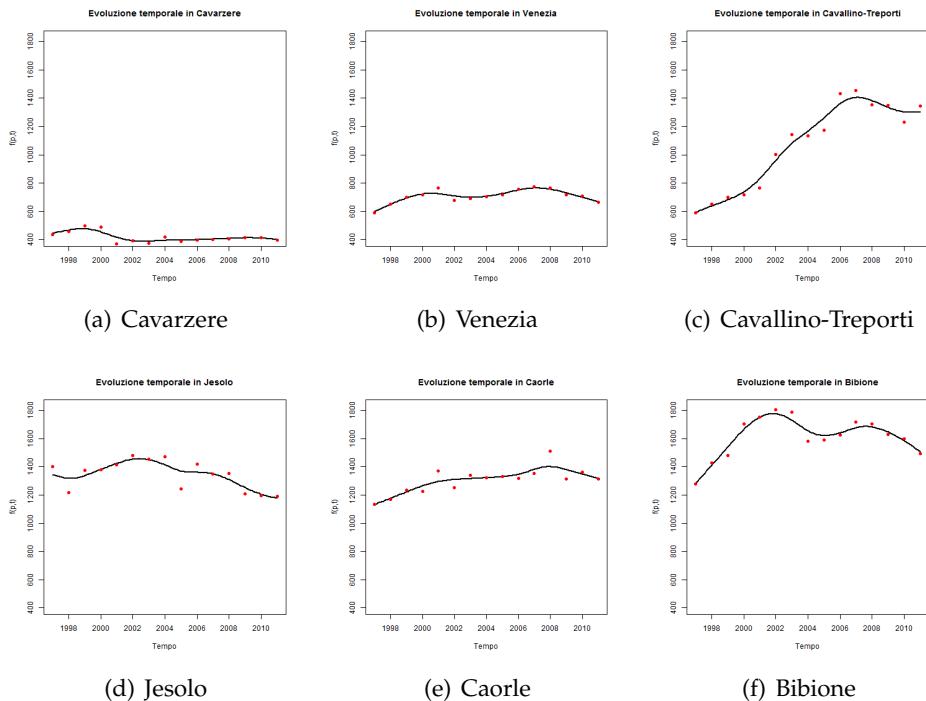


Figura 6.11: Stima della produzione di rifiuti urbani pro capite in alcuni comuni, caso senza covariata

Da tutti questi grafici è confermata la deduzione iniziale: i rifiuti sono massimi nella parte balneare della regione e minimi nella parte interna. I valori più alti si registrano in Bibione, ad est della regione. Questo conferma l'intuizione della replica del dato in questo punto. Contrariamente a quanto poteva essere supposto inizialmente, Venezia (meta turistica di fama internazionale) non è la località con la produzione più alta. Come già si notava dai grafici in fig. 6.7, il turismo sarà rilevante nell'analisi della produzione di rifiuti e la causa non sarà Venezia (dove la produzione non è eccessivamente diversa dagli altri comuni) ma le località balneari.

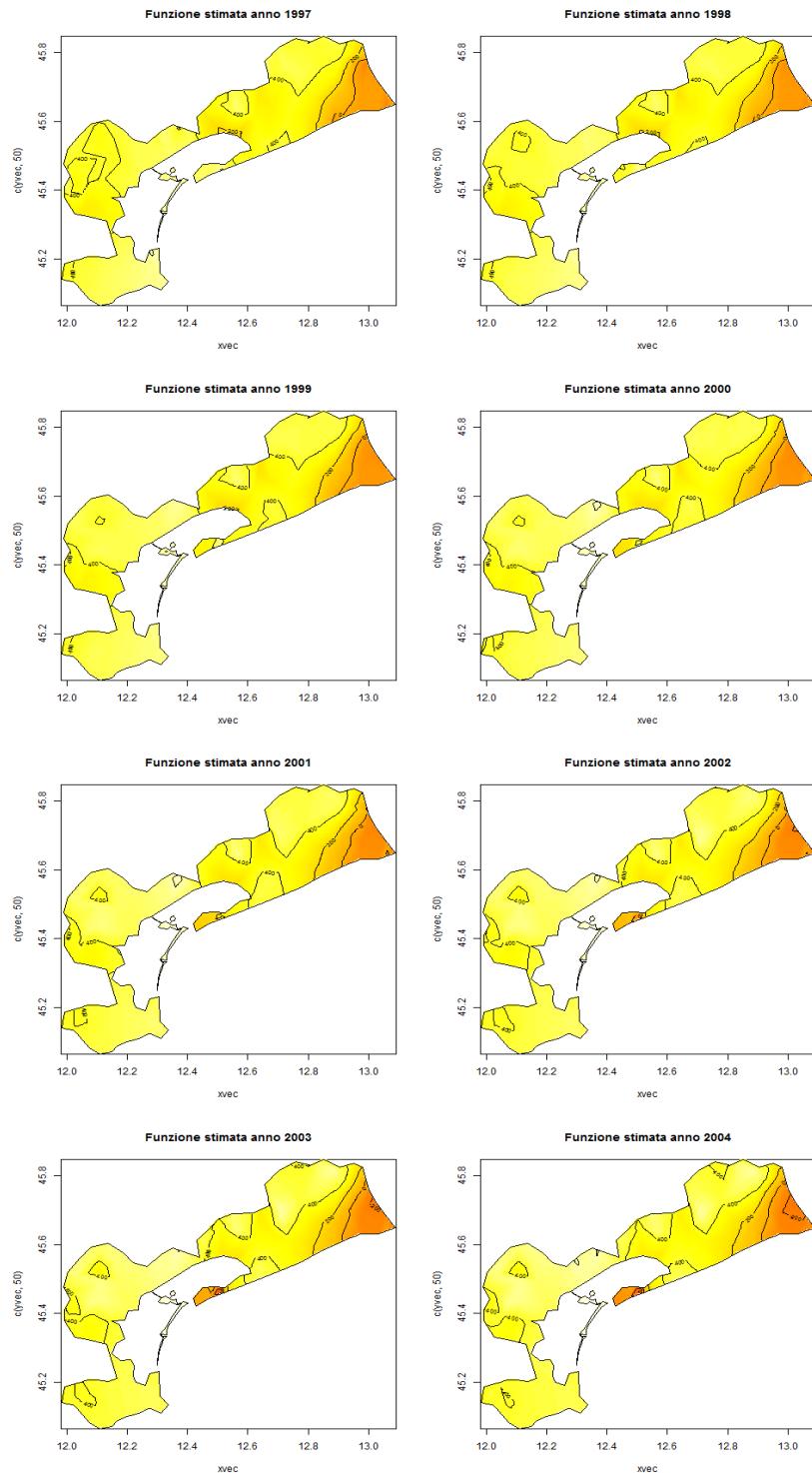
Le variazioni temporali della funzione in generale non sono marcate (i grafici in fig. 6.10 sono molto simili) ma si possono notare alcune particolarità. In fig. 6.11 sono tracciati gli sviluppi temporali stimati in alcuni comuni. I punti rossi corrispondono al dato misurato. La funzione stimata spiega bene l'andamento tracciato dai dati iniziali senza cadere nell'eccessiva interpolazione. Infatti, lo smoothing imposto dal modello tramite la penalizzazione della derivata seconda in tempo non permette di avere variazioni repentine poiché λ_T non è eccessivamente basso. Ad esempio, nei grafici di Venezia e Cavallino-Treporti, i cambiamenti dovuti alla separazione del comune dal 2002 hanno provocato un mutamento dello sviluppo

temporale, ma l'andamento generale è comunque colto dal modello.

Si noti come la risposta cresca velocemente in Cavallino-Treporti (già si vedeva in fig. 6.8). In fig. 6.10 il cambiamento più marcato nel corso degli anni è proprio in questa località. La separazione dal comune di Venezia del 2002 rende il dato più preciso in Cavallino-Treporti, che successivamente cresce negli anni. Quindi l'utilizzo di questo metodo, che considera accuratamente la geometria spaziale del problema, si conferma appropriato. Se non fosse ritenuta esclusa la zona marittima della laguna, il dato di Cavallino-Treporti influenzerebbe i comuni circostanti, Venezia compresa. Invece l'effetto dell'innalzamento del dato si propaga solo verso sud-ovest, fino al termine della piccola parte di terra prima del mare.

6.5 Applicazione del modello con covariata

Le analisi sono state eseguite con $\lambda = (10^{-7.625}, 10^{-0.125})$, valore ricavato come nei casi precedenti. Come si potrà notare dai prossimi grafici, dove è riportata la stima della funzione $f(p, t)$ in tutti i punti senza l'aggiunta della parte spiegata dalla covariata, nelle zone dove la produzione di rifiuti pro capite era massima nel caso senza covariate ora risultano i valori più bassi.



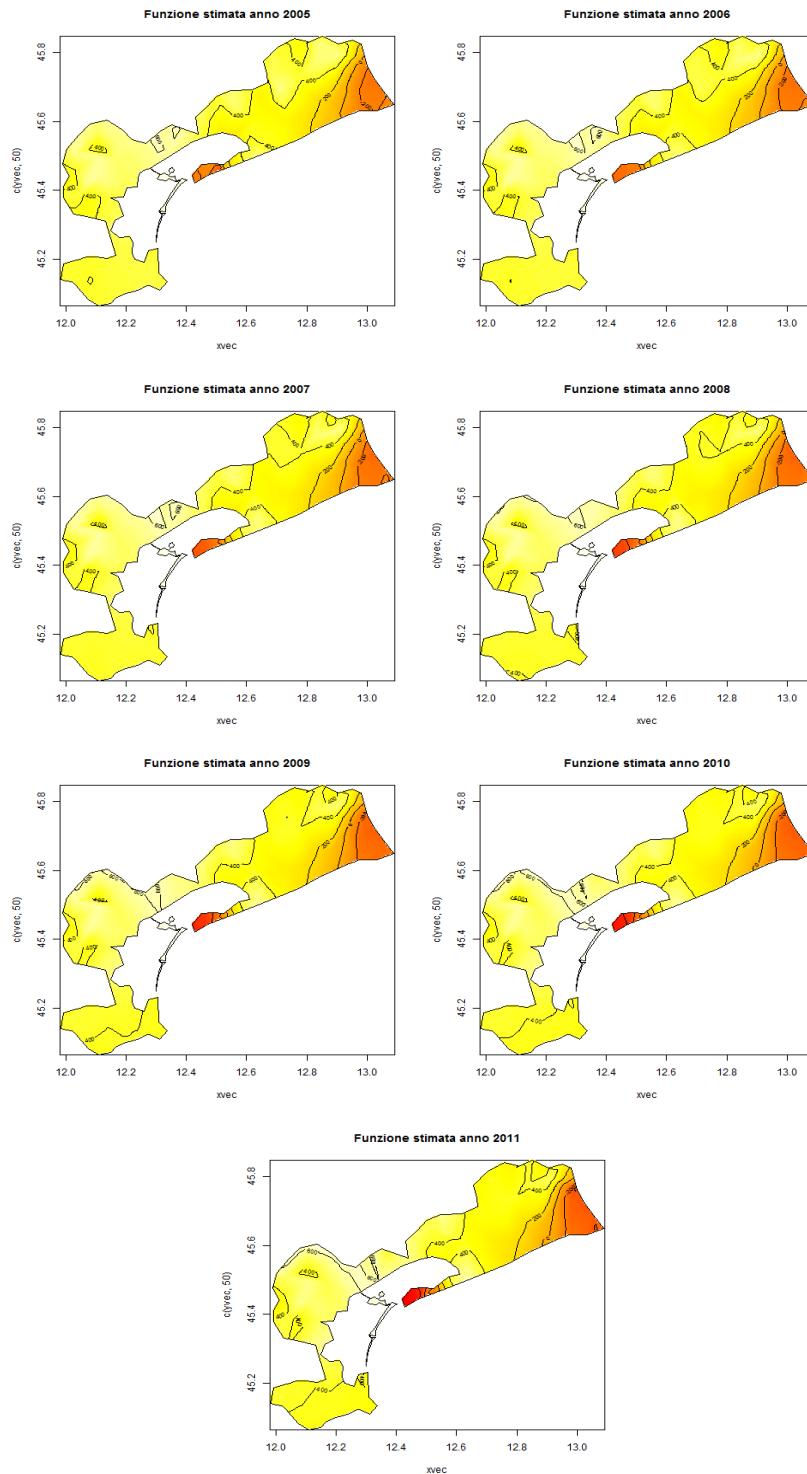


Figura 6.12: Stima della parte funzionale della produzione di rifiuti urbani pro capite nella provincia di Venezia a tempi fissati dal 1997 al 2011, caso con covariata

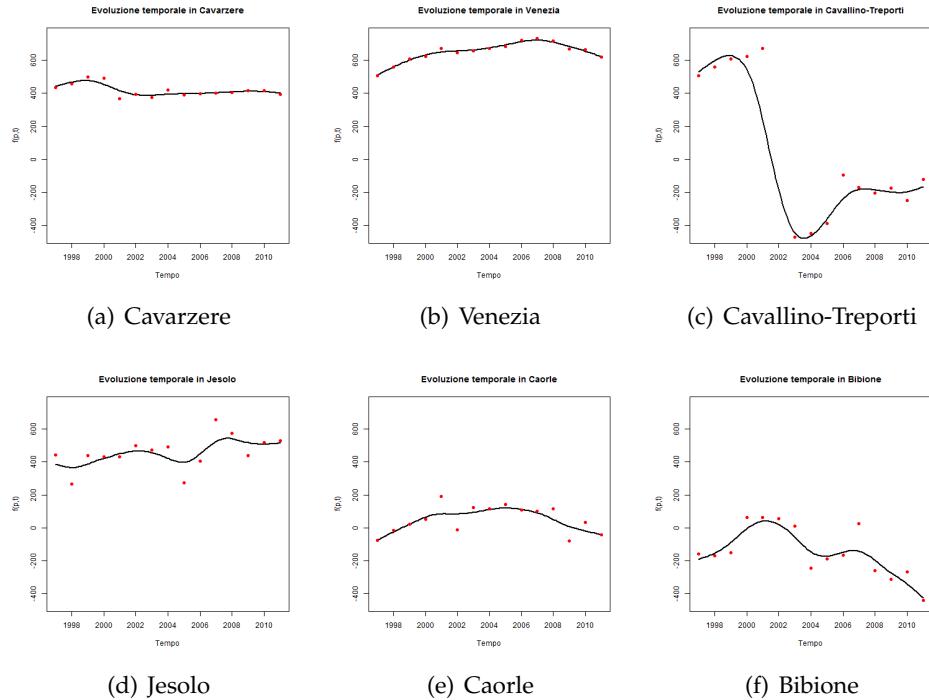


Figura 6.13: Stima della parte funzionale produzione della produzione di rifiuti urbani pro capite in alcuni comuni, caso con covariata

In fig. 6.13 sono tracciati i grafici della funzione stimata (senza il termine di covariata) negli stessi comuni che sono stati scelti per il caso senza covariata. I punti tracciati in rosso rappresentano il dato senza i termine della covariata, sottratto attraverso la moltiplicazione per $\hat{\beta}$ stimato. Non possono essere considerati come la parte di dati generati esattamente dalla funzione, ma come una buona approssimazione (ma sono comunque ciò che è analizzato dal modello per la stime di \hat{c} , come già evidenziato in sez. 2.2).

L'interpretazione del risultato in questo caso è legata all'alta affluenza turistica nelle zone balneari. Come già si notava dai grafici iniziali in fig. 6.6 e 6.7, su tutta la zona costiera non si hanno solo i massimi dei rifiuti pro capite ma anche del numero di posti letto pro capite. I valori della covariata nella parte interna della regione sono quasi trascurabili rispetto alle località balneari. Quindi non solo la produzione di rifiuti è massima sulla parte settentrionale della costa, ma ancora di più, in proporzione, lo è la covariata.

Il modello stima

$$\hat{\beta} \approx 274.034 .$$

Più la covariata è alta, più è sottratta informazione a $\hat{f}(p, t)$ dal termine

con covariata, poiché $\hat{\beta}$ è costante in spazio e tempo. Perciò la funzione ha un profilo quasi simile ad una traslazione del caso senza covariate nella parte interna del dominio (in questa zona il numero di posti letto pro capite, come era stato evidenziato in fig. 6.7, è quasi nullo) e fortemente ribassato (talvolta fino a valori negativi) nella zona balneare a causa dell'alto contributo della covariata. Dall'intervallo di confidenza per β

$$\beta \in [271.923; 276.148]$$

si può dedurre che la covariata non può essere esclusa dal modello.

Di nuovo è interessante il comportamento nel comune di Cavallino-Treporti, dove la funzione subisce una grossa variazione dopo la divisione dal comune di Venezia per l'aumento del numero dei posti letto. Attorno all'anno 2002 si ha un aumento molto forte della covariata (si veda fig. 6.7), perciò la funzione si abbassa di colpo. Tuttavia, visto che questo è dovuto alla maggior precisione del dato in Cavallino-Treporti, è un cambiamento utile a spiegar meglio la risposta. Gli effetti dovuti alle proprietà di smoothing poste dal modello (non troppo interpolanti, poiché anche in questo caso λ_T non ha un valore eccessivamente basso) portano ad un cambiamento smussato attorno all'anno della separazione del dato.

CAPITOLO 7

Conclusioni e sviluppi futuri

In questo lavoro di tesi è stato analizzato nel dettaglio il modello ST-PDE nell'ambito della stima funzionale per dati varianti all'interno di un dominio spaziale e di un intervallo temporale. L'algoritmo di stima, particolarmente adatto a domini spaziali complessi, è stato implementato in codice R. Dal confronto con gli altri metodi e da quanto ricavato con le stime, soprattutto sul dominio a forma di C in cui è possibile conoscere il valore reale della funzione, si può concludere che i risultati prodotti sono molto buoni.

Il modello è stato proposto a seguito della necessità di dover analizzare la produzione dei rifiuti urbani tra il 1997 e il 2011 nella provincia di Venezia. I dati, disponibili per comune, sono stati localizzati in punti corrispondenti al capoluogo. Tuttavia potrebbero essere considerati in modo più preciso (ma anche più complesso) come dati areali definiti sul territorio dei comuni o sulla zona urbanizzata del comune (visto che i rifiuti urbani non sono prodotti fuori da questa zona). Quindi un possibile sviluppo futuro di questo lavoro può essere sicuramente l'estensione del modello a dati non puntiformi e la conseguente applicazione a questo problema. I risultati ottenuti dalla stima con dati puntiformi sono comunque buoni, perché spiegano appropriatamente il modello. Una conferma da un eventuale modello per dati areali, però, sarebbe senza dubbio interessante.

Diversa è la valutazione conclusiva sulle prestazioni computazionali del codice. Per semplicità le basi degli elementi finiti sono state utilizzate soltanto lineari e la produzione dei rifiuti è stata analizzata solamente nella provincia di Venezia, pur avendo a disposizione i dati di tutto il Veneto. Inoltre, durante l'esecuzione del codice, si è potuto notare come alcune funzioni come la minimizzazione di $GCV(\lambda)$ o il calcolo dei valori stimati

ad un istante di tempo fissato (usato per tracciare il profilo della funzione ad un certo anno) siano molto lente. Ovviamente per analisi di dataset di grosse dimensioni deve essere messa in conto una spesa di tempo elevata, ma R certamente non ha aiutato. Infatti, è noto che R non sia un linguaggio di programmazione fortemente efficiente e questo ha prodotto la lentezza di esecuzione. Un chiaro sviluppo futuro può essere l'uso del codice come base per la creazione di un algoritmo più veloce attraverso l'integrazione con un linguaggio di programmazione più efficiente (come il C++) o dell'eventuale parallelizzazione nei colli di bottiglia più evidenti. Dopo che sarà stata sviluppata l'integrazione del codice sarà possibile garantire un'analisi più agile anche per dataset di dimensioni più elevate o per elementi finiti di ordine maggiore. In questo modo si avrà a disposizione uno strumento di analisi statistica buono non solo dal punto di vista dei risultati, ma anche in termini di efficienza computazionale.

Bibliografia

- [1] Nicole H. Augustin, Verena M. Trenkel, Simon N. Wood, Pascal Lorange, *Space-time modelling of blue ling for fisheries stock management*, Environmetrics, 24, 109–119, (2013)
- [2] Laura Azzimonti, Laura M. Sangalli, Piercesare Secchi, Maurizio Domanin, Fabio Nobile, *Blood flow velocity field estimation via spatial regression with PDE penalization*, Journal of the American Statistical Association, (2015)
- [3] Peter Craven, Grace Wahba, *Smoothing noisy data with spline functions: estimating the correct degree of smoothing by the method of generalized cross-validation*, Numerische Mathematik, 31, 377–403, (1979)
- [4] Giampiero Marra, David L. Miller, Luca Zanin, *Modelling the spatiotemporal distribution of the incidence of resident foreign population*, Statistica Neerlandica, 66, 133–160, (2012)
- [5] Timothy O. Ramsay, *Spline smoothing over difficult regions*, Journal of the Royal Statistical Society: Series B, 64, 307–319, (2002)
- [6] Laura M. Sangalli, James O. Ramsay, Timothy O. Ramsay, *Spatial spline regression models*, Journal of the Royal Statistical Society: Series B, 75, 681–703, (2013)
- [7] Simon N. Wood, Mark W. Bravington, Sharon L. Hedley, *Soap film smoothing*, Journal of the Royal Statistical Society: Series B, 70, 931–955, (2008)
- [8] A. Quarteroni, *Modellistica numerica per problemi differenziali*, Springer-Verlag Italia, (2012)

- [9] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, 2013, <http://www.R-project.org/>
- [10] J. O. Ramsay, Hadley Wickham, Spencer Graves and Giles Hooker (2014). *fda: Functional Data Analysis*. R package version 2.4.3. <http://CRAN.R-project.org/package=fda>
- [11] Douglas Bates and Martin Maechler (2015). *Matrix: Sparse and Dense Matrix Classes and Methods*. R package version 1.1-5. <http://CRAN.R-project.org/package=Matrix>
- [12] Shewchuk JR (1996). *Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator*. In Lin MC and Manocha D (eds.), Applied Computational Geometry: Towards Geometric Engineering, volume 1148 series Lecture Notes in Computer Science, pp. 203-222. Springer-Verlag. From the First ACM Workshop on Applied Computational Geometry.
- [13] Wood, S.N. (2011) *Fast stable restricted maximum likelihood and marginal likelihood estimation of semiparametric generalized linear models*. Journal of the Royal Statistical Society (B) 73(1):3-36
- [14] Wood, S.N. (2004) *Stable and efficient multiple smoothing parameter estimation for generalized additive models*. Journal of the American Statistical Association. 99:673-686.
- [15] Wood, S.N. (2006) *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC.
- [16] Wood, S.N. (2003) *Thin-plate regression splines*. Journal of the Royal Statistical Society (B) 65(1):95-114.
- [17] Wood, S.N. (2000) *Modelling and smoothing parameter estimation with multiple quadratic penalties*. Journal of the Royal Statistical Society (B) 62(2):413-428.
- [18] Pebesma, E.J., 2004. *Multivariable geostatistics in S: the gstat package*. Computers & Geosciences, 30: 683-691.
- [19] Robert J. Hijmans (2015). *raster: Geographic data analysis and modeling*. R package version 2.3-24. <http://CRAN.R-project.org/package=raster>
- [20] Markus Loecher (2014). *RgoogleMaps: Overlays on Google map tiles in R*. R package version 1.2.0.6. <http://CRAN.R-project.org/package=RgoogleMaps>

- [21] H. Wickham. *ggplot2: elegant graphics for data analysis*. Springer New York, 2009.
- [22] D. Kahle and H. Wickham. *ggmap: Spatial Visualization with ggplot2*. The R Journal, 5(1), 144-161. <http://journal.r-project.org/archive/2013-1/kahle-wickham.pdf>