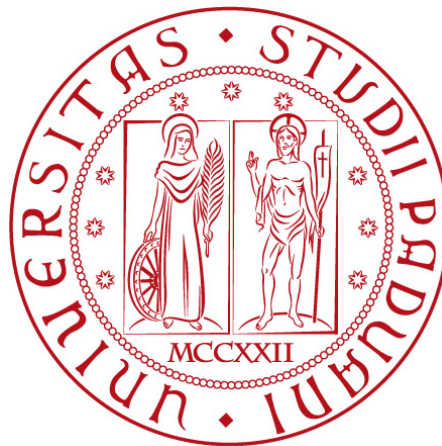


Università degli Studi di Padova
Dipartimento di Scienze Statistiche
Corso di Laurea Triennale in
Statistica per le Tecnologie e le Scienze



RELAZIONE FINALE

Modelli statistici per la diffusione delle Innovazioni
DIMORA: Un pacchetto statistico

Relatore: Prof. Bruno Scarpa
Dipartimento di Scienze Statistiche

Laureando: Federico Zanghi
Matricola: 1124703

Anno Accademico 2017/2018

*"In the history of diffusion of many innovations,
one cannot help being struck by two characteristics
of the diffusion process: its apparent overall slowness on the
one hand, and the wide variations in the rates of acceptance
of different inventions, on the other."*

(N. Rosenberg, 1976:191)

Alla mia famiglia,

Introduzione

Un'innovazione è un'applicazione di un'invenzione tale da conferire a quest'ultima un significato economico. Il termine si può riferire a tutti i componenti di un sistema e al sistema stesso; si può dunque parlare di innovazione di prodotto, di processo, di sistema produttivo ecc.

”La diffusione di una innovazione in un mercato o in un sistema sociale è una tematica di grande interesse nella gestione delle imprese o nel governo di istituzioni pubbliche. La conoscenza precoce delle dimensioni qualitative e quantitative di tali processi consente l'introduzione di opportune scelte strategiche il cui orizzonte operativo dipende molto dalla velocità di penetrazione e dall'ampiezza del mercato o dei gruppi di cittadini raggiungibili”. (*L'equazione di Riccati, il modello di Bass e i processi di adozione delle innovazioni, R. Guseo, 2002*)

Per questo motivo vengono utilizzati modelli di previsione adatti a descrivere il possibile ciclo di vita del nuovo prodotto: *i modelli di diffusione delle innovazioni*. Il riferimento principale è rivolto al modello di Bass (1969), modello che incorpora nella stima delle vendite il tasso d'innovazione e il tasso d'imitazione, rendendolo un modello completo sotto ogni punto di vista. L'obiettivo di questo lavoro è sviluppare nell'ambiente R una serie di funzioni che permettano un utilizzo di tali modelli anche agli utenti meno esperti del software. In particolare i modelli considerati saranno : *il modello di Bass standard, il modello di Bass Generalizzato* ed *il modello con mercato potenziale variabile*, ponendo una particolare attenzione alla formulazione proposta da *Guseo-Guidolin* nel 2009.

Per questo motivo, nel Capitolo 1 verrà trattato il concetto di diffusione di un prodotto, lasciando spazio alla definizione e descrizione dei modelli sopracitati, nel Capitolo 2 propone una panoramica generale del pacchetto sviluppato, illustrandone sue funzionalità e, per ciascuna funzione, viene richiamata la misura o l'aspetto che questa intende implementare.

Nel capitolo 3 infine verrà presentato un breve esempio applicativo del pacchetto a dati reali, in cui verranno applicati il modello di Bass standard, il modello di Bass generalizzato con uno shock rettangolare, il modello con potenziale variabile con $m(t)$ definito secondo il modello di Guseo-Guidolin ed utilizzando una diversa specificazione.

Indice

Introduzione	i
1 Modelli di diffusione	1
1.1 Ciclo di vita di un prodotto	1
1.1.1 Introduzione	1
1.1.2 Crescita	2
1.1.3 Maturità	2
1.1.4 Declino	2
1.2 Innovatori ed Imitatori	3
1.3 Il modello di Bass standard	3
1.4 Il modello di Bass Generalizzato	7
1.4.1 Impulsi Rettangolari	8
1.4.2 Impulsi Esponenziali	9
1.4.3 Impulsi Misti	11
1.4.4 Impulsi Armonici	12
1.5 Il modello Guseo-Guidolin	13
1.5.1 Confronto Guseo Guidolin e Bass Standard	15
1.6 Stima dei parametri	16
2 Il pacchetto di funzioni sviluppate	19
2.1 Generalità del pacchetto	19
2.2 Gestione dei dati	20
2.3 Il modello di Bass Standard	21
2.4 Il modello di Bass Generalizzato	22

2.5	Modello con mercato potenziale dinamico	23
3	Il caso di studio LYRICA	25
	Conclusioni e possibili estensioni	33
A	L'equazione di Riccati	35
B	DIMORA package	37
	B.0.1 Gestione dati:	37
	B.0.2 Modello di Bass Standard:	37
	B.0.3 Modello di Bass Generalizzato:	39
	B.0.4 Modello con mercato potenziale variabile:	44
	Bibliografia	47

Elenco delle figure

1.1	Ciclo di vita di un prodotto	2
1.2	Innovatori ed Imitatori	5
1.3	Modello di Bass standard	6
1.4	Modello di Bass Generalizzato con shock rettangolare	9
1.5	Modello di Bass Generalizzato con shock Esponenziale	10
1.6	Modello di Bass Generalizzato con shock misto	11
1.7	Modello di Guseo-Guidolin e modello di Bass standard	15
3.1	Serie storica settimanale <i>'lyrica'</i>	26
3.2	Modello di Bass standard su dati <i>'lyrica'</i>	27
3.3	Modello di Bass Generalizzato con uno shock esponenziale su dati <i>'lyrica'</i>	29
3.4	Modello di Guseo-Guidolin su i dati <i>'lyrica'</i>	30
3.5	Modello con mercato potenziale variabile su i dati <i>'lyrica'</i>	32

Capitolo 1

Modelli di diffusione

”Il processo di diffusione è costituito da quattro elementi chiave: l’innovazione, il sistema sociale sul quale impatta tale innovazione, i canali di comunicazione del sistema ed il tempo”.(Rogers, 1983).

1.1 Ciclo di vita di un prodotto

Il ciclo di vita del prodotto è un modello previsionale basato sull’andamento delle vendite nel periodo compreso dall’ingresso nel mercato a quello della sua rimozione.

Le quattro fasi del ciclo di vita del prodotto sono: introduzione, crescita, maturità e declino.

1.1.1 Introduzione

La fase iniziale di introduzione di un prodotto sul mercato comincia con il lancio del prodotto stesso. La prima fase del ciclo di vita del prodotto si caratterizza per una lenta crescita delle vendite dovuta a una domanda inesistente che deve essere ancora creata dall’azienda.

1.1.2 Crescita

Quando la domanda comincia a crescere rapidamente il prodotto entra nella fase di crescita del ciclo di vita. Le vendite prendono slancio e i profitti aumentano a tassi molto elevati per poi cominciare a decrescere verso la fine di questo stadio man mano che nuovi concorrenti entrano nel mercato.

1.1.3 Maturità

Nella fase di maturità la curva delle vendite presenta un flesso per poi raggiungere il punto di massimo profitto, oltre il quale il tasso comincia a decrescere. Questa fase del ciclo di vita del prodotto, a cui corrisponde un periodo di vendite a livelli elevati e più o meno costanti attorno al punto di massimo profitto.

1.1.4 Declino

Quando la domanda comincia a decrescere rapidamente, il prodotto raggiunge la fase di declino venendo via via espulso dal mercato.

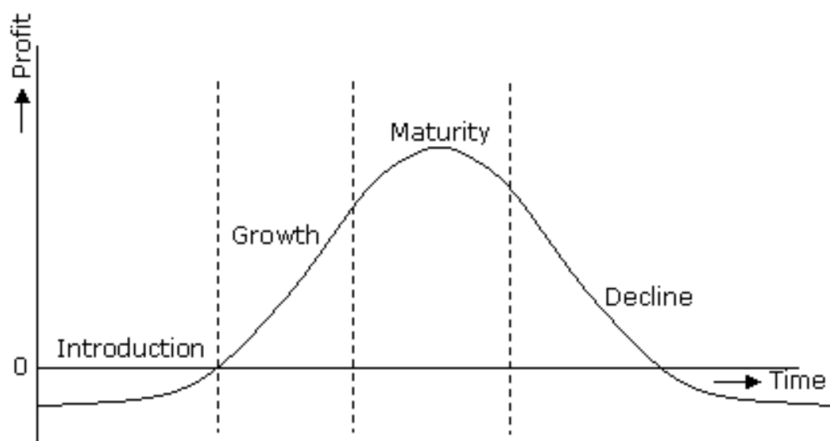


Figura 1.1: Ciclo di vita di un prodotto

1.2 Innovatori ed Imitatori

Per innovazione si intende la "fase conclusiva e fondamentale del processo di crescita economica e tecnologica dell'impresa avviato da invenzioni o scoperte. Le modalità con le quali si presenta l'**innovazione**. sono varie: un'idea, un prodotto, un nuovo processo produttivo, [...]". (Treccani, 2018)

Nella diffusione di un'innovazione l'individuo, ed in particolare la sua reazione, ne determinano il successo, il comportamento degli acquirenti, per quanto variabile potrebbe essere raggruppato in diverse categorie, Roger nel 1965 ne aveva definite ben 5, noi tuttavia ci limitiamo ad analizzare le due macrocategorie definite da Bass nel 1969 ovvero: *innovatori* ed *imitatori*.

Gli innovatori, come suggerisce il nome, rappresentano la porzione di popolazione che acquista un prodotto durante la fase iniziale del processo diffusivo, ovvero durante la fase di lancio.

Gli imitatori, invece, procedono con l'acquisto conseguentemente al giudizio che gli innovatori conferiscono al prodotto, questi ricoprono infatti la seconda e terza parte del ciclo di vita di un prodotto.

Sono fondamentali entrambe le categorie per garantire il successo di un'innovazione, in quanto innovatori ed imitatori sono strettamente correlati, il giudizio di uno influenza l'altro, il quale rappresenta la maggior parte del mercato.

1.3 Il modello di Bass standard

Il Modello di Bass (1969) è formato da un'equazione differenziale che descrive il processo di adozione delle innovazioni in un sistema economico. L'equazione di base del modello si ottiene come caso particolare dell'equazione di Riccati ¹ a coefficienti costanti e assume la forma,

$$y' = (p + qy)(1 - y), \quad (1.1)$$

¹Si veda appendice A

con $p, q \geq 0$ e y è una funzione di ripartizione mentre y' la corrispondente densità. La ripartizione è usualmente riferita all'indice temporale t ed esprime quindi la probabilità che l'evento si sia verificato non oltre t .

Se volessimo esprimere il modello in funzione delle vendite cumulate, sostanzialmente un conteggio degli eventi di interesse, la formuazione sarebbe,

$$z = my, \quad (1.2)$$

con m costante positiva, detta *mercato potenziale*. Nel caso di modello cumulato, m rappresenta l'asintoto superiore, ovvero il massimo che la numerosità degli eventi può raggiungere. Nel caso della presentazione prodotto m rappresenta appunto il mercato totale raggiungibile; il modello di Bass risulta essere parzialmente lineare nel parametro m che, come già detto, corrisponde all'asintoto superiore.

Possiamo esprimere il modello (1.1) considerando anche l'effetto del mercato ovvero invertendo la (1.2), così facendo otteniamo,

$$z' = m\left(p + q\frac{z}{m}\right)\left(1 - \frac{z}{m}\right) = \left(p + q\frac{z}{m}\right)(m - z), \quad (1.3)$$

dove risultano evidenti e ben distinti l'effetto *innovativo* ed *imitativo*, in particolare:

- $p(m - z)$ esprime l'effetto innovativo, ed infatti risulta essere direttamente proporzionale al mercato residuo, chiaramente l'effetto degli *innovatori* cala col passare del tempo, tuttavia risulta in ogni caso molto utile in quanto permette di capire come il prodotto inizialmente verrà diffuso.
- $q\frac{z}{m}(m - z)$ esprime l'effetto imitativo del processo che, pur essendo ancora proporzionale al mercato residuo, viene corretto da una penalizzazione decrescente.

Possiamo dunque notare che il modello di Bass somma due sottomodelli: il modello a componente *innovativa*² e il modello a componente *imitativa*³.

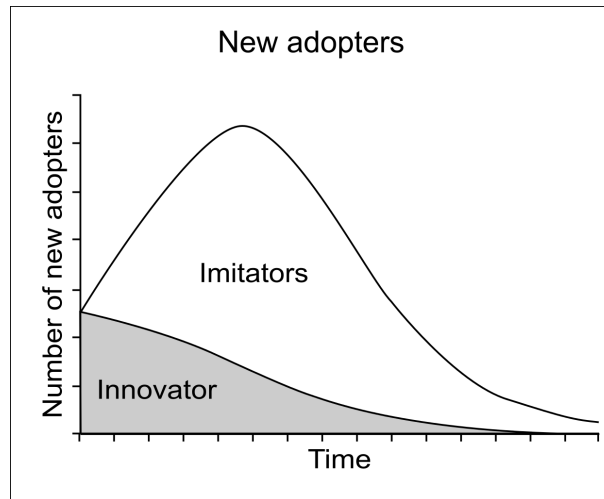


Figura 1.2: Innovatori ed Imitatori

Possiamo quindi riformulare l'equazione (1.1) esplicitando la componente *imitativa* ed *innovativa*:

$$y' + qy^2 + (p + q)y - p = 0, \quad (1.4)$$

Le radici dell'equazione caratteristica sono $r_1 = -\frac{p}{q}$ e $r_2 = 1$, per cui, $a(r_2 - r_1) = p + q$. Solitamente, $1 - y$ e y' sono termini positivi, e dunque anche p e q risulteranno positivi. In questo avremo che $a = q$ e $r_1 < 0 < r_2$. Effettuando qualche sostituzione otteniamo,

$$y(t) = \frac{1 - e^{-(p+q)t}}{1 + \frac{q}{p}e^{-(p+q)t}}, \quad (1.5)$$

²Il modello monomolecolare per la componente innovativa

³Il modello logistico di Verhulst per la componente imitativa (1983)

Ricordando la formula (1.2) ne consegue che,

$$z(t) = m \frac{1 - e^{-(p+q)t}}{1 + \frac{q}{p} e^{-(p+q)t}}, \quad t > 0; p, q > 0 \quad (1.6)$$

Il Modello di Bass standard appena descritto risulta efficace nel modellare le serie di vendite cumulate per la maggior parte delle innovazioni. Tuttavia per serie reali, spesso c'è bisogno di un modello in grado di adattarsi ad andamenti non regolari, quali shock positivi e negativi nelle curve di diffusione. Proprio per questo tipo di problemi entra in gioco il modello di Bass Generalizzato.

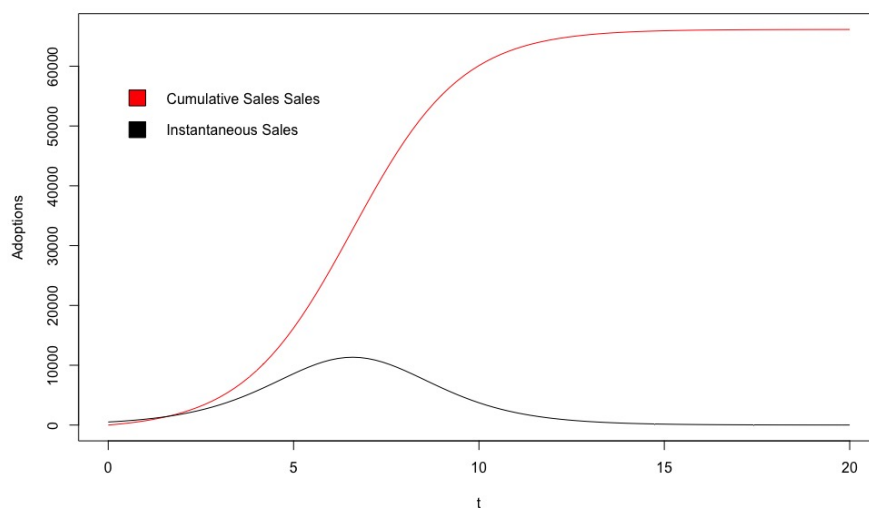


Figura 1.3: Modello di Bass standard

1.4 Il modello di Bass Generalizzato

Nel modello di Bass Generalizzato (introdotto nel 1994 da Bass, Krishnan e Jain) si introduce una funzione *portante*, $x(t)$, che permette di modificare l'andamento del modello, in intervalli opportuni. L'equazione del modello di Bass generalizzato risulta essere dunque:

$$z' = m\left(p + q\frac{z}{m}\right)\left(1 - \frac{z}{m}\right)x(t) = m\frac{p(p+q)^2x(t)e^{-(p+q)\int_0^t x(\tau)d\tau}}{(pe^{(p+q)\int_0^t x(\tau)d\tau})^2}, \quad (1.7)$$

dove $x(t)$ è una funzione *integrabile* in *dominii limitati* e *non negativa*. Il Bass standard (1.1) non è altro che un caso particolare di (1.7), infatti nel caso di Bass standard tale funzione assume valore unitario $x(t) = 1$.

Il valore di $x(t)$ determina il rallentamento o l'accelerazione della diffusione, infatti se $x(t) \in [0, 1]$ avremo un rallentamento viceversa quando $x(t) \in [1, +\infty)$ saremo in presenza di un'accelerazione. La soluzione in forma chiusa del modello di Bass Generalizzato esplicitando innovatori (p) ed imitatori (q) risulta essere:

$$z(t) = m\frac{1 - e^{-(p+q)\int_0^t x(\tau)d\tau}}{1 + \frac{q}{p}e^{-(p+q)\int_0^t x(\tau)d\tau}}, \quad t > 0; p, q > 0 \quad (1.8)$$

Questa, a meno di una costante moltiplicativa m , è una funzione di ripartizione definita nel dominio temporale $t \in [0, +\infty)$.

È interessante notare che la funzione portante $x(t)$ pur modificando il processo di adozione (accelerandolo o rallentandolo) non interviene in nessuna maniera sul mercato potenziale m , in altre parole, $x(t)$ altera la densità⁴ lasciando però inalterata la funzione di ripartizione $y(t)$.

⁴ $y'(t) = \frac{z'(t)}{m}$

Naturalmente $x(t)$ può essere definita a proprio piacimento, purchè soddisfisi le condizioni di integrabilità e non negatività; ci sono però alcune definizioni di $x(t)$ che meritano attenzione in quanto spesso, si adattano molto bene a dati reali.

1.4.1 Impulsi Rettangolari

Una semplice, ma spesso molto efficace formulazione di $x(t)$ è rappresentata dagli impulsi rettangolari, infatti si tratta di una perturbazione più stabile che agisce sul processo diffusivo per un determinato periodo di tempo, viene definita come:

$$x(t) = 1 + c_1 I_{t \geq a_1} I_{t \leq b_1} + c_2 I_{t \geq a_2} I_{t \leq b_2} + c_3 I_{t \geq a_3} I_{t \leq b_3} \quad (1.9)$$

dove i parametri $[a_i, b_i]$ con $i \in [1, 2, 3]$ indicano l'intervallo in cui applicare lo shock, mentre c_i con $i \in [1, 2, 3]$ ne indica l'intensità. Questo tipo di formulazione si traduce in un comportamento temporaneo stazionario.

Per gli impulsi rettangolari, abbiamo una soluzione in forma chiusa dell'integrale definito, presente in (1.8):

$$\begin{aligned} \int_0^t x(\tau) d\tau = & t + c_1(t - a_1) I_{t \geq a_1} I_{t \leq b_1} + c_1(b_1 - a_1) I_{t > b_1} + \\ & + c_2(t - a_2) I_{t \geq a_2} I_{t \leq b_2} + c_2(b_2 - a_2) I_{t > b_2} + \\ & + c_3(t - a_3) I_{t \geq a_3} I_{t \leq b_3} + c_3(b_3 - a_3) I_{t > b_3}. \quad (1.10) \end{aligned}$$

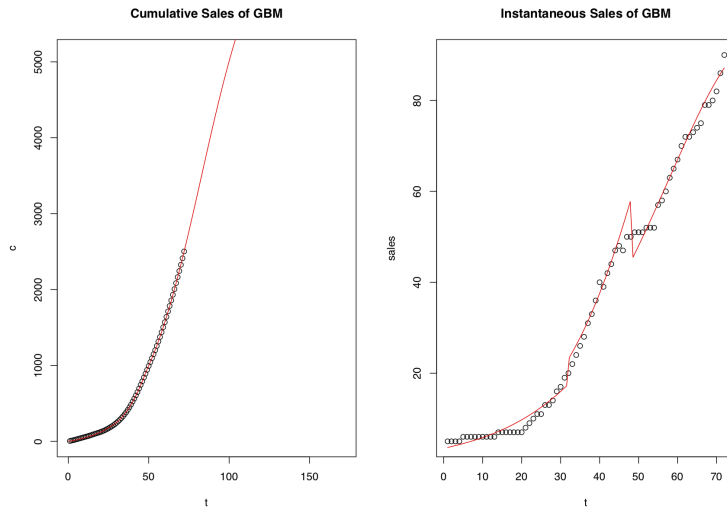


Figura 1.4: Modello di Bass Generalizzato con shock rettangolare ⁵

1.4.2 Impulsi Esponenziali

In alcuni casi è necessaria una perturbazione molto intensa e allo stesso tempo istantanea, tale situazione può essere descritta da una funzione esponenziale del tipo:

$$x(t) = 1 + c_1 e^{b_1(t-a_1)} I_{t \geq a_1} + c_2 e^{b_2(t-a_2)} I_{t \geq a_2} + c_3 e^{b_3(t-a_3)} I_{t \geq a_3}. \quad (1.11)$$

dove i parametri a_i con $i \in [1, 2, 3]$ rappresentano i tempi di insorgenza degli shock, i parametri b_i con $i \in [1, 2, 3]$ rappresentano la velocità di decadimento dello shock, mentre c_i con $i \in [1, 2, 3]$ rappresentano i valori iniziali dello shock, possono assumere valori positivi o negativi a seconda del tipo di

⁵I dati provengono da uno studio di Campbell (2002) condotto sui consumi annui mondiali di Gas Metano in Tcf (Trillion of cubic feet).

perturbazione. Anche in questo caso abbiamo la soluzione in forma chiusa dell'integrale in (1.8):

$$\int_0^t x(\tau) d\tau = t + c_1 \frac{1}{b_1} (e^{b_1(t-a_1)} - 1) I_{t \geq a_1} + \\ + c_2 \frac{1}{b_2} (e^{b_2(t-a_2)} - 1) I_{t \geq a_2} + \\ + c_3 \frac{1}{b_3} (e^{b_3(t-a_3)} - 1) I_{t \geq a_3}. \quad (1.12)$$

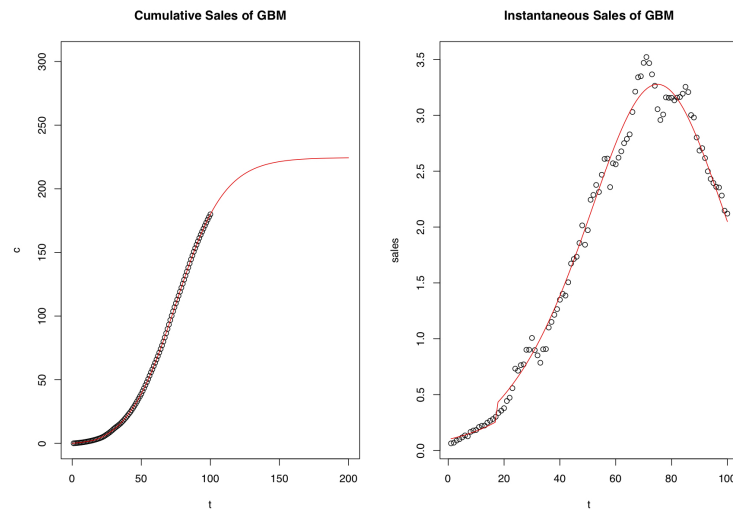


Figura 1.5: Modello di Bass Generalizzato con shock esponenziale ⁶

⁶Nell'esempio sono stati utilizzati i volumi di estrazione del crude oil degli U.S.A. i dati sono espressi in miliardi di barili annui. La fonte indiretta è Industriedatenbank con una serie che copre il periodo che va dall'anno 1900 fino al 2000. Per ulteriori approfondimenti si veda Morreale (2004).

1.4.3 Impulsi Misti

In alcuni casi, la funzione portante $x(t)$ può dover interpretare perturbazioni di diverso tipo in intervalli temporali diversi, dovuti a cause differenti. Un caso molto semplice e molto utilizzato è costituito da una coppia di impulsi, il primo esponenziale, il secondo rettangolare.

$$x(t) = 1 + c_1 e^{b_1(t-a_1)} I_{t \geq a_1} + c_2 I_{t \geq a_2} I_{t \leq b_2}, \quad (1.13)$$

dove i parametri assumono lo stesso significato di quanto spiegato in precedenza. Anche in questo caso abbiamo la soluzione in forma chiusa dell'integrale:

$$\int_0^t x(\tau) d\tau = t + c_1 \frac{1}{b_1} (e^{b_1(t-a_1)} - 1) I_{t \geq a_1} + c_2 (t - a_2) I_{t \geq a_2} I_{t \leq b_2} + c_2 (b_2 - a_2) I_{t > b_2}. \quad (1.14)$$

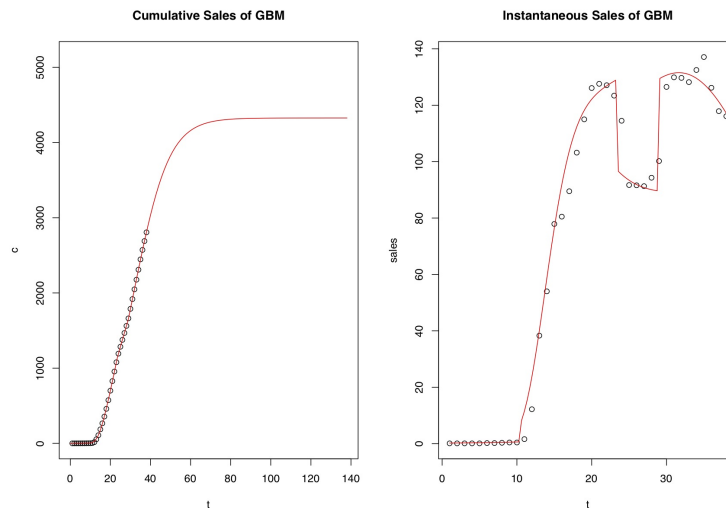


Figura 1.6: Modello di Bass Generalizzato con shock misto ⁷

⁷I dati si riferiscono alle produzioni di crude oil della Gran Bretagna. La serie copre il periodo che va dall'anno 1965 al 2002,

1.4.4 Impulsi Armonici

In alcuni casi il comportamento di $z(t)$ può essere alterato da una perturbazione simmetrica, ovvero un susseguirsi di impulso positivo e negativo o viceversa. Nel caso in questione avremo:

$$\begin{aligned}
 x(t) = 1 + c_1 \cos\left(2\pi \frac{(t - a_1)}{(b_1 - a_1)}\right) I_{t \geq a_1} I_{t \leq b_1} + \\
 + c_2 \cos\left(2\pi \frac{(t - a_2)}{(b_2 - a_2)}\right) I_{t \geq a_2} I_{t \leq b_2} + \\
 + c_3 \cos\left(2\pi \frac{(t - a_3)}{(b_3 - a_3)}\right) I_{t \geq a_3} I_{t \leq b_3}, \quad (1.15)
 \end{aligned}$$

dove i parametri a_i con $i \in [1, 2, 3]$ rappresentano i tempi di insorgenza degli shock, i parametri b_i con $i \in [1, 2, 3]$ rappresentano i tempi di interruzione, mentre c_i con $i \in [1, 2, 3]$ rappresentano i valori iniziali degli shock ed anche in questo caso possono assumere valori positivi o negativi a seconda del tipo. La soluzione forma chiusa dell'integrale è:

$$\begin{aligned}
 \int_0^t x(\tau) d\tau = t + c_1 \frac{b_1 - a_1}{2\pi} \sin\left(2\pi \frac{(t - a_1)}{(b_1 - a_1)}\right) I_{t \geq a_1} I_{t \leq b_1} + \\
 + c_2 \frac{b_2 - a_2}{2\pi} \sin\left(2\pi \frac{(t - a_2)}{(b_2 - a_2)}\right) I_{t \geq a_2} I_{t \leq b_2} + \\
 + c_3 \frac{b_3 - a_3}{2\pi} \sin\left(2\pi \frac{(t - a_3)}{(b_3 - a_3)}\right) I_{t \geq a_3} I_{t \leq b_3}. \quad (1.16)
 \end{aligned}$$

1.5 Il modello Guseo-Guidolin

Il modello Guseo-Guidolin (2009) nasce dall'idea che l'assunzione fatta sul mercato potenziale, ovvero che rimanga costante nel tempo, sia troppo stringente e spesso non rappresentativa della realtà. Il modello infatti si presenta come un'estensione del modello di Bass Generalizzato in cui però il mercato potenziale entra in gioco come variabile dipendente dal tempo.

Il modello di Guseo-Guidolin non è altro che una specificazione del modello di diffusione con mercato potenziale variabile, infatti $m(t)$ può essere una qualsiasi funzione che soddisfi specifiche condizioni, in linea di principio, il mercato potenziale può essere una qualsiasi funzione $m(t)$ che definisce un limite superiore per le vendite cumulate $z(t)$, ovvero $z(t) \leq m \forall t$.

Nel modello di Guseo e Guidolin, $m(t)$ viene specificato basandosi su una descrizione formale delle dinamiche della conoscenza riguardanti un'innovazione specifica; se gli effetti di comunicazione sono persistenti, cioè senza effetto di decadimento e nessun passaparola negativo il mercato potenziale viene definito come:

$$m(t) = KG(t) = K \sqrt{\frac{1 - e^{-(p_c+q_c)t}}{1 + \frac{q_c}{p_c} e^{-(p_c+q_c)t}}}. \quad (1.17)$$

Il comportamento limite, ovvero nel caso in cui tutti coloro che sono informati divengono potenziali acquirenti, avremo che $m(t)$ è uguale alla capacità di carico costante K :

$$\lim_{t \rightarrow +\infty} m(t) = K$$

Il modello Guseo-Guidolin risulterà dunque essere:

$$z(t) = KG(t)F(t) = K \sqrt{\frac{1 - e^{-(p_c+q_c)t}}{1 + \frac{q_c}{p_c} e^{-(p_c+q_c)t}} \frac{1 - e^{-(p_s+q_s)t}}{1 + \frac{q_s}{p_s} e^{-(p_s+q_s)t}}}. \quad (1.18)$$

Dove i parametri p_c e q_c descrivono il processo diffusivo, che può avvenire per mezzo di comportamenti innovativi o imitativi mentre i parametri p_s e q_s descrivono il normale processo di adozione di un prodotto (come nel modello di Bass). In questo modello il mercato potenziale viene creato dalla diffusione dell'informazione relativa al prodotto: in altre parole, diviene potenziale adottante del prodotto solo chi ne è informato. Una buona diffusione dell'informazione, implica un rapido sviluppo del mercato potenziale $m(t)$.

Per quanto riguarda i dati istantanei l'equazione del modello diventa:

$$z'(t) = m(t) \left[\left(p + q \frac{z(t)}{m(t)} \right) \left(1 - \frac{z(t)}{m(t)} \right) \right] + z(t) \frac{m'(t)}{m(t)}. \quad (1.19)$$

dove $m'(t)$ indica la derivata prima del mercato potenziale, ovvero dell'equazione (1.17).

1.5.1 Confronto Guseo Guidolin e Bass Standard

Un esempio applicativo di come funziona il modello viene riportato in seguito; i dati utilizzati riguardano il ciclo di vita di un farmaco (*Fertifol*) in Italia. Proveremo prima a modellare i dati con il modello di Bass ed in seguito con il modello Guseo-Guidolin. I parametri di partenza per ottenere le stime di m , p e q per il modello di Bass sono rispettivamente: 1174953 , 0.01 , 0.1 , mentre per il modello di Guseo-Guidolin useremo per m , p_c e q_c le stime ottenute con il modello di Bass standard mentre per p_s e q_s useremo rispettivamente 0.001 e 0.1 .

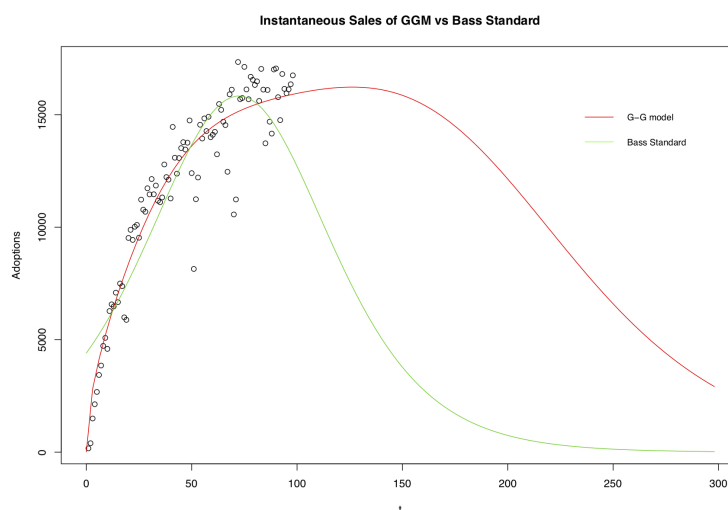


Figura 1.7: Modello di Guseo-Guidolin e modello di Bass standard

Come possiamo notare, sia dal grafico, che da i valori dei coefficienti di correlazione ($R^2 = 0.9995977$ e $R^2 Adj = 0.9995849$ per il modello di Bass Standard, mentre $R^2 = 0.9999453$ e $R^2 Adj = 0.9999423$ per il modello G-G) è evidente che il modello con il mercato potenziale variabile risulta più efficace.

1.6 Stima dei parametri

L'utilizzo del modello di Bass e delle sue più complesse estensioni richiede la stima di diversi parametri; studi empirici hanno confermato che l'utilizzo dei minimi quadrati lineari (*OLS*) produce delle stime non affidabili per i parametri in questione. Le stime dunque vengono tipicamente effettuate utilizzando i minimi quadrati non lineari (*NLS*) ritenendo questo metodo di stima molto più efficace, pur producendo delle stime distorte.

Una complicazione che si riscontra nell'utilizzo dei minimi quadrati non lineari consiste nel dover passare esplicitamente dei valori iniziali per la stima di ogni parametro, questo potrebbe causare problemi in quanto i valori iniziali che solitamente vengono utilizzati potrebbero rivelarsi non applicabili non portando l'algoritmo a convergenza, costringendoci dunque a cercare dei parametri ad hoc per i nostri dati, inoltre valori di partenza lontani dalle stime potrebbero portare a stime non ottime. Questo problema si riscontra soprattutto nel modello di Bass Generalizzato (1.7) con più di uno shock in quanto il numero di parametri da stimare cresce notevolmente. Un metodo che è risultato ancora più efficiente, pur non rimuovendo tutti i problemi sopracitati, è quello dei minimi quadrati smorzati (*DLS*); per calcolarli si sfrutta l'algoritmo di Levenberg-Marquardt, che essendo più robusto dell'algoritmo di Gauss-Newton in molti casi riesce a trovare una soluzione anche se i parametri sono lontani dal minimo finale. Affinchè funzioni bene è comunque importante che i parametri di partenza siano ragionevoli. Come altri algoritmi di minimizzazione numerica, l'algoritmo Levenberg-Marquardt è una procedura iterativa. Per iniziare una minimizzazione, l'utente deve fornire un'ipotesi iniziale per il vettore di parametri β . Nei casi con un solo minimo i parametri iniziali hanno poca importanza mentre nei casi con minimi multipli, l'algoritmo converge al minimo globale solo se l'ipotesi iniziale è già un po' vicina alla soluzione finale. L'algoritmo si può vedere come composto da una fase di discesa del gradiente, lenta ma che converge, seguita da un risolutore di tipo Gauss-Newton, più veloce.

In ogni fase di iterazione, il vettore di parametro β viene sostituito da

una nuova stima $\beta + \delta$. Per determinare δ si utilizza la funzione:

$$f(x_i, \beta + \delta) \approx f(x_i, \beta) + J_i \delta \quad (1.20)$$

dove J_i rappresenta il gradiente, ed è pari a:

$$J_i = \frac{\partial f(x_i, \beta)}{\partial \beta} \quad (1.21)$$

Sapendo che la somma dei quadrati delle deviazioni $S(\beta)$ ha il suo minimo in corrispondenza del gradiente nullo, dall'approssimazione di primo ordine di $f(x_i, \beta + \delta)$ otteniamo:

$$S(\beta + \delta) \approx \sum_{i=1}^m [y_i - f(x_i, \beta) - J_i \delta]^2 \quad (1.22)$$

A questo punto prendendo la derivata prima di $S(\beta + \delta)$ rispetto a δ e uguagliando a 0 il risultato, otteniamo:

$$(J^T J) \delta = J^T [y - f(\beta)] \quad (1.23)$$

Dove J rappresenta la matrice Jacobiana, le cui righe sono rappresentate da J_i e dove f e y sono i vettori le cui componenti sono rispettivamente $f(x_i, \beta)$ e y_i . Quello che otteniamo è dunque un problema lineare che può essere facilmente risolto per δ .

Capitolo 2

Il pacchetto di funzioni sviluppate

In questo capitolo verrà presentato il pacchetto che permette di utilizzare tutti i modelli spiegati nel capitolo precedente.

2.1 Generalità del pacchetto

Al momento non esistono pacchetti R che permettano di analizzare facilmente i dati di diffusione sfruttando tutti i modelli spiegati nel capitolo precedente. Quello che si propone con DIMORA (*diffusion models R analysis*) è un pacchetto che in qualsiasi situazione riesca a stimare correttamente i parametri dei modelli senza perdita di generalità e con un *output* facilmente interpretabile ed il più informativo possibile. I modelli presentati nel capitolo precedente utilizzano i dati cumulati sia per quanto riguarda le stime che per il calcolo dei vari indici. Tuttavia solitamente i dati che abbiamo a disposizione sono dati istantanei, quindi ho ritenuto più opportuno passare alle funzioni del pacchetto quest'ultimi. Le funzioni poi calcoleranno in automatico i dati cumulati per procedere con le stime.

Le funzioni presenti nel pacchetto sono:

<i>Funzionalità</i>	<i>Nome funzione</i>
Gestione dati	<code>make.instantaneous</code>
Bass Standard	<code>BASS.plot</code>
	<code>BASS.standard</code>
	<code>BASS.standard.generator</code>
Bass Generalizzato	<code>BASS.generalized</code>
Modello con potenziale variabile	<code>GG.model</code>

2.2 Gestione dei dati

Questa è una funzione molto semplice che però può rivelarsi utile in quanto tutte le funzioni presenti nel pacchetto richiedono in *input* i dati istantanei. Questa infatti trasforma i dati cumulati in istantanei.

```
make.instantaneous(cumulate.data)
```

come detto in precedenza questa funzione riceve in *input* i dati cumulati e restituisce i dati istantanei.

2.3 Il modello di Bass Standard

La funzione più importante per quanto riguarda il modello di Bass standard (1.3 e 1.6) è la seguente:

```
BASS.standard( sales , method = 'nls',
               prelimestimates = c(sum(sales)+100,
                                   0.01, 0.1),display=T)
```

Questa funzione permette di stimare i parametri del modello in due metodi diversi, i minimi quadrati non lineari (*nls*) oppure minimizzano la somma del quadrato dei residui (*optim*), in particolare alla funzione bisognerà fornire in *input*:

- **sales**: Fa riferimento ad un vettore contenente i dati istantanei sui quali vogliamo stimare il modello di Bass Standard.
- **method**: I valori possibili sono 'nls' e 'optim' che identificano rispettivamente il metodo di stima dei parametri con i minimi quadrati non lineari e attraverso la minimizzazione della somma dei quadrati dei residui. Il metodo predefinito è quello dei minimi quadrati non lineari.
- **prelimestimates**: Con *prelimestimates*, come si evince anche dal nome, si fa riferimento alle stime preliminari, necessarie come punto di partenza per poter stimare i parametri, sia per 'nlm' che per 'optim'. Dei valori di partenza troppo lontani dalle vere stime potrebbero portare alla non convergenza dell'algoritmo oppure all'identificazione di stime rappresentanti dei massimi locali e non globali. I valori predefiniti sono: un numero di poco maggiore dal massimo valore raggiungibile dai dati cumulati (l'asintoto superiore) , 0.01, e 0.1.
- **alpha**: Il livello di significatività desiderato per gli intervalli di confidenza. Il valore predefinito è 0.05.
- **display**: Con *display*, le cui possibili opzioni sono 'T' e 'F', si può decidere se ricevere in output le stime e i grafici o solamente le stime.

La funzione restituisce le *stime* dei parametri con i relativi *standard error*, un intervallo di confidenza con livello di significatività α ed il *p-value*. Inoltre vengono restituiti *R-squared* e *Adjusted R-squared*.

La seconda funzione che consideriamo per il modello di Bass Standard è la seguente:

```
BASS.standard.generator(m,p,q,tstart=1,n=50)
```

Questa funzione semplicemente genera dei dati per il modello di Bass con m , p e q passati dall'utente. I dati partono di default dal tempo $t=1$ e la lunghezza di default del dataframe che viene restituita è 50.

Strettamente connessa a questa funzione c'è *BASS.plot* che dato un dataframe (come quello che restituisce `BASS.standard.generator`) restituisce i grafici dei dati cumulati ed istantanei.

```
BASS.plot(data)
```

2.4 Il modello di Bass Generalizzato

Questa funzione permette di analizzare i dati utilizzando il modello di Bass generalizzato (1.7 e 1.8), con diversi tipi di shock. La funzione viene richiamata nel seguente modo:

```
BASS.generalized( sales, shock = c('exp','rett','mixed',  
                                'armonic'), nshock, prelimestimates  
                alpha = 0.05)
```

Gli argomenti da passare alla funzione sono:

- **sales**: Fa riferimento ad un vettore contenente i dati istantanei sui quali vogliamo stimare il modello di Bass Generalizzato

- **shock**: Le possibili opzioni sono 'exp', 'rett', 'mixed' e 'armonic' che rappresentano il tipo di shock e sono rispettivamente *rettangolare*, *esponenziale*, *misto* ed *armonico*. Questi, escluso lo shock misto nel caso $nshock=3$ che non è stato implementato, sono tutti utilizzabili nei contemplati in questa funzione, ovvero $nshock \in [1, 2, 3]$.
- **nshock**: Rappresenta il numero di perturbazioni (*shock*) necessarie nella funzione.
- **prelimestimates**: Indicano i valori iniziali che 'nls' utilizzerà per le stime dei parametri, è importante notare che anche in questo caso stime preliminari molto lontane dalle stime reali potrebbero portare a stime non rappresentative o alla non convergenza dell'algoritmo. In questo caso risulta difficile assegnare dei valori di default a *prelimestimates*, dunque si è preferito lasciare all'utente il compito di definire i valori di partenza.
- **alpha**: Anche in questo caso indica il livello di significatività desiderato per gli intervalli di confidenza.

La funzione restituirà in *output* le *stime* dei parametri con i relativi *standard error*, gli *intervalli di confidenza* di livello fissato ed i *p-value*. Inoltre verranno restituiti i grafici dei dati istantanei e cumulati e le relative curve dei modelli stimati, l'*R-squared* e l'*Adjusted R-squared*.

2.5 Modello con mercato potenziale dinamico

La funzione *GG.model* è stata implementata proprio per considerare i modelli con mercato potenziale variabile:

```
GG.model( sales, prelimestimates = NULL, mt = 'base' ,  
          alpha = 0.05, ... )
```

Gli argomenti da passare alla funzione sono:

- **sales**: Vettore contenente i dati istantanei sui quali vogliamo stimare il modello.
- **prelimestimates**: Indica sempre le stime preliminari per i parametri k, p_c, q_c, p_s e q_s , anche in questo caso stime lontane dal vero valore possono portare alla non convergenza dell'algoritmo. In questa funzione il valore predefinito per le stime (*NULL*) indica il caso particolare in cui le stime per k, p_c e q_c vengono stimate sfruttando il modello di Bass standard (*la funzione BASS.standard*) mentre per p_s e q_s abbiamo rispettivamente 0.001 e 0.1.
- **mt**: Rappresenta il mercato potenziale variabile, questo può assumere valore 'base' nel caso in cui si voglia il modello di Guseo-Guidolin (1.17) oppure è possibile specificare una funzione
- **alpha**: Anche in questo caso indica il livello di significatività desiderato per gli intervalli di confidenza.
- ...: Parametri grafici.

Questa funzione permette di analizzare dati di diffusione non soltanto con il classico modello di Guseo-Guidolin ma con un qualsiasi modello di diffusione con mercato potenziale variabile, semplicemente specificando la propria funzione $m(t)$ ¹.

Anche in questo caso, quello che la funzione restituirà saranno le *stime* con *standard error*, *intervalli di confidenza* ed i *p-value*. Anche in questo caso avremo i grafici dei dati istantanei e cumulati e le relative curve dei modelli stimati, l'*R-squared* e l'*Adjusted R-squared*.

¹È importante notare che le possibili funzioni specificabili per $m(t)$ devono soddisfare seguenti condizioni: la funzione deve essere monotona crescente con dominio in $[0, 1]$

Capitolo 3

Il caso di studio LYRICA

Il dataset *Lyrice.csv* rappresenta la serie storica settimanale, che va dal 05 Settembre 2005 al 9 Luglio 2007, delle vendite di un farmaco, il *lyrica*. Su questa si è deciso di applicare come prima cosa il modello di Bass standard (1.3) per poi provare qualche modello più complicato, dal modello di Bass generalizzato con uno shock rettangolare (1.9) al modello con mercato potenziale variabile (1.17). Come risulterà evidente, l'utilizzo del pacchetto *DIMORA*, risulta molto semplice.

Procediamo dunque con l'analisi, come prima cosa carichiamo il pacchetto e naturalmente i dati:

```
1 library(DIMORA)
2 lyrice <- read.csv('~/Desktop/Lyrice.csv',header = F)
3 lyrice <- lyrice$V1
```

Risulta opportuno visualizzare i dati per verificare, graficamente se si tratta di dati cumulati o istantanei, ricordando che tutte le funzioni presenti nel pacchetto richiedono l'utilizzo di dati istantanei. Dunque:

```
1 plot(lyrice,xlab = 'Time')
```

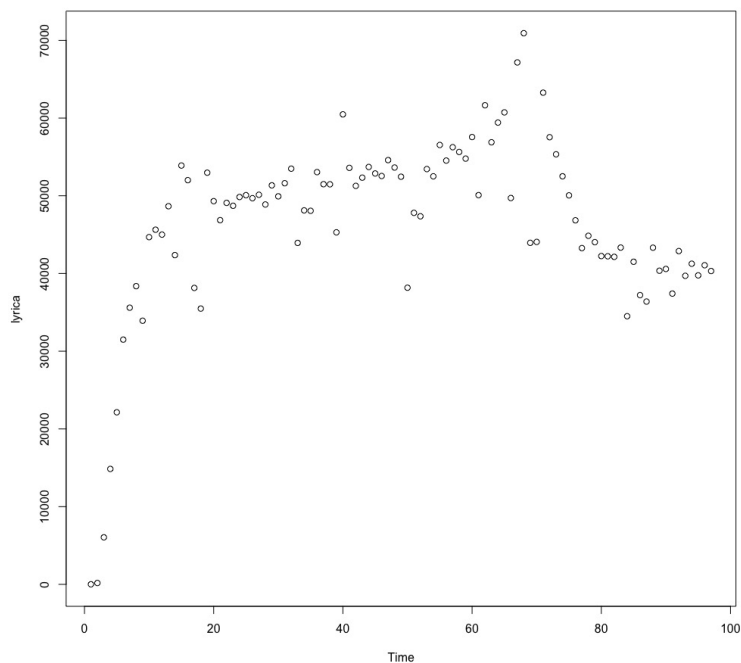


Figura 3.1: Serie storica settimanale 'lyrica'

Come possiamo notare, i dati risultano essere una serie storica istantanea, dunque possiamo procedere con l'utilizzo dei modelli, senza dover ricorrere alla funzione `make.instantaneous`.

Applichiamo dunque il modello di Bass standard, con la funzione `BASS.standard`, non specificando né il metodo di stima né le stime preliminari, ricadendo dunque nel default, ovvero *minimi quadrati non lineari* e $c(\text{sum}(\text{sales})+100, 0.01, 0.1)$.

```
1 BASS.standard(sales = lyrica)
```

Come possiamo notare il modello si adatta abbastanza bene e i suoi coefficienti risultano tutti ampiamente significativi.

```
## $Estimate
##      Estimate      Std.Error      Lower      Upper  p-value
## m : 5.668671e+06 7.802125e+04 5.515752e+06 5.821590e+06 2.17e-84
## p : 4.881109e-03 4.760005e-05 4.787815e-03 4.974403e-03 2.79e-98
## q : 2.908790e-02 7.909659e-04 2.753764e-02 3.063817e-02 1.35e-57
##
## $Rsquared
## [1] 0.9997941
##
## $RsquaredAdj
## [1] 0.9997874
```

Graficamente possiamo però vedere che l'adattamento non è perfetto, questo lo notiamo soprattutto nei dati istantanei.

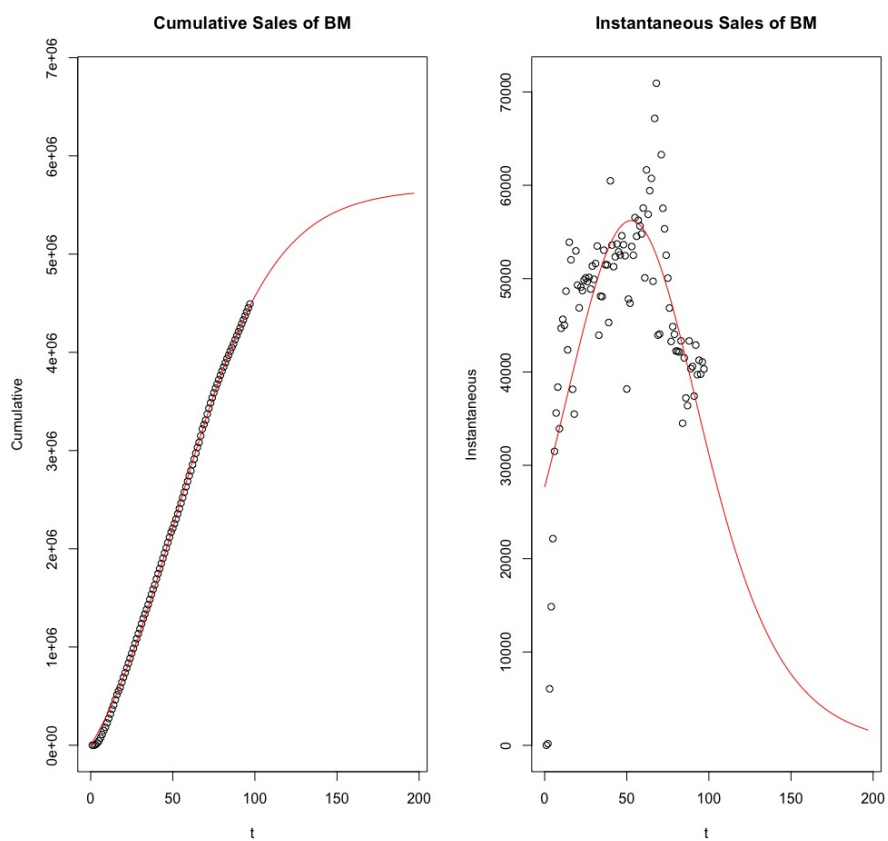


Figura 3.2: Modello di Bass standard su dati *'lyrica'*

Proviamo ora ad applicare il modello di Bass generalizzato con uno shock rettangolare, per vedere se l'adattamento migliora, sfruttiamo la funzione `BASS.generalized`, dove come stime preliminari per m, p e q abbiamo utilizzato le stime ottenute con il modello di Bass standard, mentre per a_1, b_1 e c_1 il valore '10':

```
1 BASS.generalized(sales = lyrica, shock = 'rett', nshock = 1, prelinestimates = c(1.69062e+06, 2.60513e-03, 3.20522e-02, 10, 10, 10), alpha = 0.05)
```

Quello che abbiamo ottenuto dalla stima del modello è:

```
## $Estimate
##           Estimate   Std.Error      Lower      Upper  P-value
## m :  6.138301e+06  2.157821e+05  5.715375e+06  6.561226e+06  4.54e-47
## p :  5.800940e-03  6.583520e-04  4.510594e-03  7.091286e-03  7.84e-14
## q :  2.262883e-02  4.704113e-03  1.340894e-02  3.184873e-02  5.94e-06
## a1 : -1.100654e+03  1.245692e+06  -2.442613e+06  2.440411e+06  9.99e-01
## b1 : -1.101425e+03  1.245613e+06  -2.442458e+06  2.440255e+06  9.99e-01
## c1 :  4.065694e+00  3.662721e+05  -7.178761e+05  7.178843e+05  1.00e+00
##
## $Rsquared
## [1] 0.9999256
##
## $RsquaredAdj
## [1] 0.9999206
```

In questo caso notiamo un miglioramento nell'adattamento, tuttavia non tutti i parametri risultano significativi, quindi non si può ritenere affidabile il valore di R_{Adj}^2 .

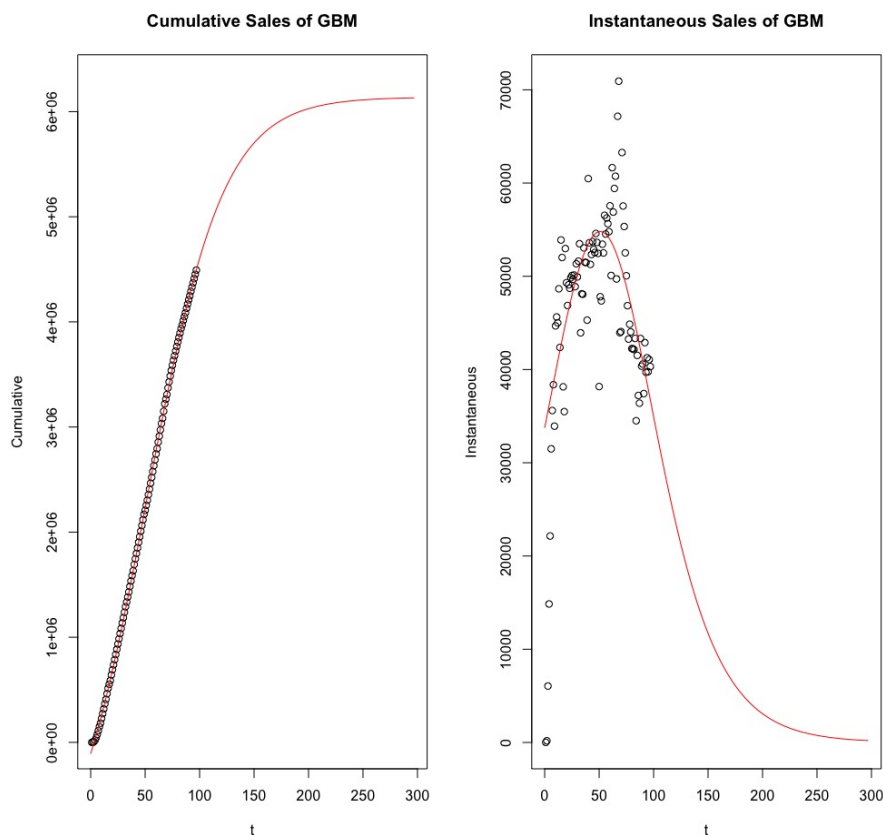


Figura 3.3: Modello di Bass Generalizzato con uno shock esponenziale su dati *'lyrica'*

Applichiamo a questo punto il modello di Guseo-Guidolin, sfruttando dunque la funzione per il mercato (m_t) canonica ovvero quella specificata in 1.17, inoltre non specificando le stime preliminari sfrutteremo quelle ottenute con il modello di Bass standard per k , p_c e q_c mentre per p_s e q_s avremo rispettivamente 0.001 e 0.1.

```
1 GG.model(sales = lyrica,col=2)
```

Quello che otteniamo in output non è molto soddisfacente nemmeno in questo caso in quanto non tutti i parametri risultano significativi nel modello, infatti:

```

## $Estimate
##           Estimate      Std.Error      Lower      Upper P-value
## k : 2.570420e+07 6.763645e+07 -1.068608e+08 1.582692e+08 7.05e-01
## pc : 1.838145e-03 1.096282e-03 -3.105285e-04 3.986818e-03 9.70e-02
## qc : 3.806059e-02 6.155440e-03 2.599614e-02 5.012503e-02 1.70e-08
## ps : 8.785713e-03 2.157545e-02 -3.350139e-02 5.107282e-02 6.85e-01
## qs : -4.007051e-02 1.257112e-02 -6.470945e-02 -1.543158e-02 1.96e-03
##
## $Rsquared
## [1] 0.9999218
##
## $RsquaredAdj
## [1] 0.9999175

```

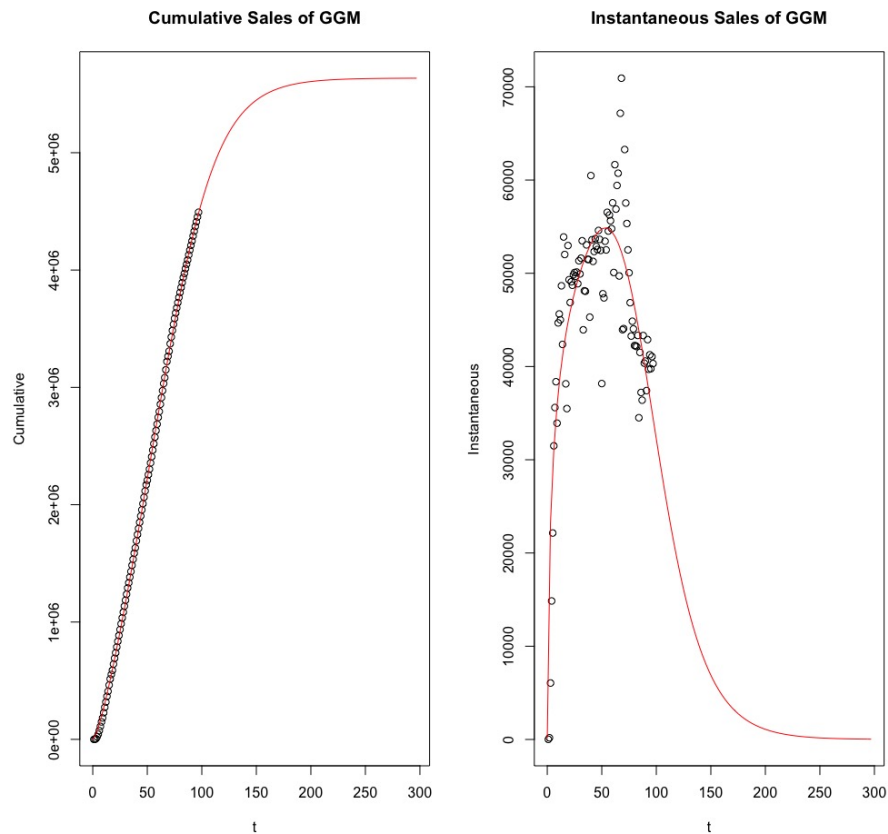


Figura 3.4: Modello di Guseo-Guidolin su i dati *'lyrica'*

Proviamo ad applicare a questo punto un modello con potenziale varibi-

le, non limitandoci a considerare la forma canonica per $m(t)$, sempre però ricordando quali sono le condizioni che questa funzione deve soddisfare, decidiamo di sfruttare la funzione di ripartizione di un $\chi^2(10)$ ottenendo dunque un modello così definito:

$$z(t) = Km(t) \frac{1 - e^{-(p_s+q_s)t}}{1 + \frac{q_s}{p_s} e^{-(p_s+q_s)t}}. \quad (3.1)$$

Dove $k = Gdl$ e $m(t)$:

$$m(t) = \frac{1}{\Gamma(\frac{k}{2})} \gamma\left(\frac{k}{2}, \frac{x}{2}\right) \quad (3.2)$$

Avremo dunque:

```
1 GG.model(sales = lyrica, mt = function(x) pchisq(x,10), col=2)
```

Da cui otteniamo:

```
## $Estimate
##      Estimate  Std.Error      Lower      Upper  P-value
## k : 5.968413e+06 5.760627e+04 5.855507e+06 6.081320e+06 1.07e-98
## ps : 4.987358e-03 2.769400e-05 4.933079e-03 5.041638e-03 3.78e-121
## qs : 2.579339e-02 4.682720e-04 2.487559e-02 2.671118e-02 2.40e-73
##
## $Rsquared
## [1] 0.9999376
##
## $RsquaredAdj
## [1] 0.9999356
```

che è un risultato molto soddisfacente in quanto l' R^2 ha un valore molto elevato, e naturalmente maggiore di quello dei modelli precedenti, inoltre i parametri risultano essere tutti ampiamente significativi.

Possiamo inoltre notare dal grafico dei dati istantanei come il modello appena stimato segua bene i dati.

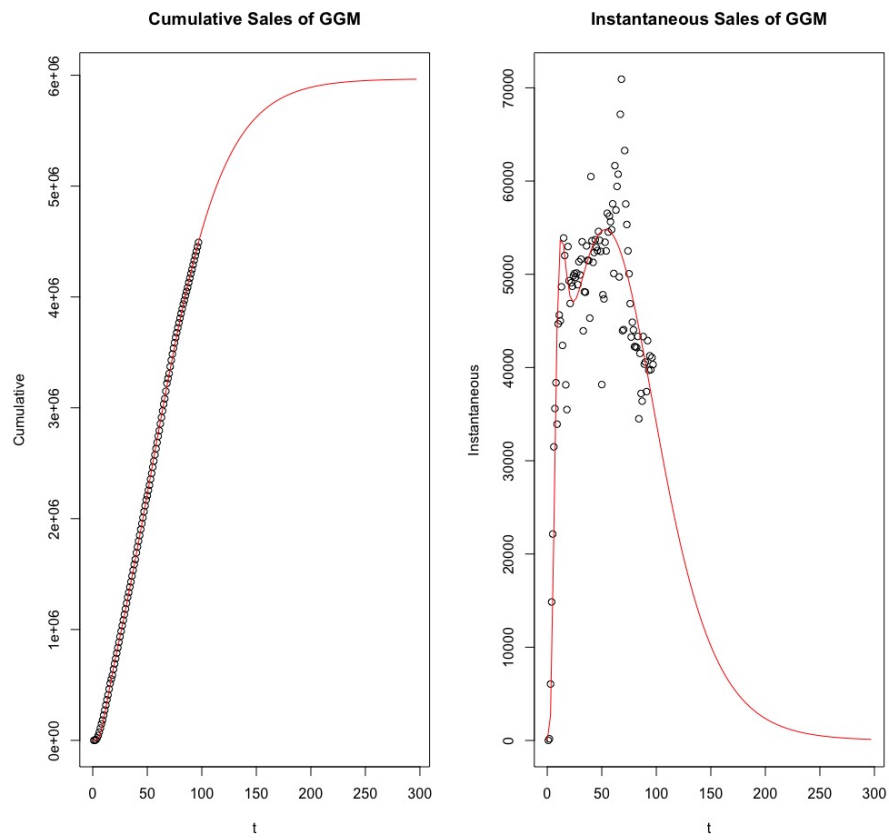


Figura 3.5: Modello con mercato potenziale variabile su i dati 'lyrica'

Conclusioni e possibili estensioni

Il lavoro che ho svolto all'interno di questa tesi è stato quello di illustrare una precisa formalizzazione matematica per gestire i dati di diffusione, concentrando l'attenzione sul *Modello di Bass*, sulla sua versione *generalizzata* in particolare considerando perturbazioni *esponenziali, rettangolari, miste ed armoniche* ed infine su *Modelli con mercato potenziale variabile* non limitandosi alla formulazione proposta da Guseo-Guidolin (2009).

Ho inoltre sviluppato un pacchetto di funzioni, per il software *R*, di facile utilizzo ed interpretazione con la speranza che sia il più possibile generale ed applicabile in qualunque situazione.

Malgrado il pacchetto presenti dei limiti, legati alle mie conoscenze, questo permette comunque di effettuare un'analisi dei dati, probabilmente non approfondita come quelle effettuabili con altri software (come ad esempio *Statgraphics*), fornendo stime, standard error, intervalli di confidenza e p-value per i parametri, viene inoltre fornito un indice di bontà del modello (il coefficiente di determinazione, R^2) infine in output avremo la rappresentazione grafica dei dati istantanei e cumulati, con le relative curve teoriche.

Il pacchetto sviluppato può certamente essere esteso e migliorato in varie direzioni. Le estensioni che personalmente ritengo più importanti sono: in primo luogo l'aggiunta di una funzione che permetta di stimare il modello di Bass generalizzato (1.8) sfruttando una funzione di shock definita dall'utente, che calcoli però la curva in modo teorico e non interpolando semplicemente

alcuni punti. Una seconda opzione, secondo me di maggiore interesse, è quella di proporre un algoritmo in grado di calcolare in ogni situazione le stime preliminari. Questo permetterebbe di eliminare completamente i problemi di non convergenza e non affidabilità delle stime in alcune situazioni e ridurrebbe di molto il carico di lavoro dell'utente, rendendo così il pacchetto di più semplice e comodo utilizzo.

Appendice A

L'equazione di Riccati

Consideriamo anzitutto l'equazione di Riccati a coefficienti costanti come base di partenza da cui in seguito deriveremo il modello di Bass standard, che è definita come:

$$y' + ay^2 + by + c = 0 \quad (\text{A.1})$$

dove a, b e $c \in R$.

Siano r_1 e r_2 le soluzioni, anche complesse, dell'equazione caratteristica, possiamo dunque rappresentare la (A.1) in funzione delle sue radici:

$$y' + a(y - r_1)(y - r_2) = 0 \quad (\text{A.2})$$

Se ipotizziamo $y \neq r_2$ l'equazione (A.2) può essere riscritta come:

$$-\frac{y'}{(y - r_2)^2} = \frac{a(y - r_1)}{(y - r_2)} \quad (\text{A.3})$$

se deriviamo il secondo membro della (A.3):

$$\left(\frac{a(y - r_1)}{(y - r_2)} \right)' = -\frac{ay'(y - r_2) - y'(y - r_1)a}{(y - r_2)^2} = -\frac{-y'a(r_2 - r_1)}{(y - r_2)^2} \quad (\text{A.4})$$

quello che otteniamo è un'equazione uguale, a meno di una costante moltiplicativa $a(r_2 - r_1)$, al primo membro dell'equazione (A.3). L'unica funzione $y(x)$ che soddisfa tale proprietà è l'esponenziale, $z = \alpha e^{\beta x}$, a cui corrisponde

$z' = \beta \alpha e^{\beta x} = \beta z$. A questo punto possiamo riscrivere la soluzione generale dell'equazione (A.2) ottenendo:

$$\frac{(y - r_1)}{(y - r_2)} = C_0 e^{a(r_2 - r_1)t} \quad (\text{A.5})$$

esplicitando la y si ottiene:

$$y = \frac{r_1 - r_2 C_0 e^{a(r_2 - r_1)t}}{1 - C_0 e^{a(r_2 - r_1)t}} \quad (\text{A.6})$$

La derivata prima dell'equazione (A.6) rispetto a t risulta essere:

$$y' = \frac{-a C_0 e^{a(r_2 - r_1)t} (r_2^2 - 2r_1 r_1 + r_1^2)}{(C_0 e^{a(r_2 - r_1)t} - 1)^2} \quad (\text{A.7})$$

dove $\lim_{t \rightarrow \pm\infty} y'(t) = 0$.

Per $a > 0$ l'equazione (A.6) è dotata di due asintoti, precisamente: $\lim_{t \rightarrow -\infty} y(t) = r_1$ e $\lim_{t \rightarrow +\infty} y(t) = r_2$, nel caso in cui $a < 0$ risulterebbero invertiti.

Possiamo applicare ancora qualche trasformazione per rendere ancora più immediata la somiglianza con il modello di Bass, infatti se imponiamo $y(0) = 0$ avremo che $C_0 = \frac{r_1}{r_2}$ (posti $r_1 \neq 0$ e $r_2 \neq 0$) e dunque, semplicemente moltiplicando numeratore e denominatore per $-e^{-a(r_2 - r_1)t}$ otteniamo:

$$y = \frac{1 - e^{-a(r_2 - r_1)t}}{\frac{1}{r_2} - \frac{1}{r_1} e^{-a(r_2 - r_1)t}} \quad (\text{A.8})$$

Appendice B

DIMORA package

B.0.1 Gestione dati:

```
1  make.instantaneous <- function(cumulate.data){
2  firstdiff <- function(x) {
3  shifted <- c(0,x[1:(length(x)-1)])
4  x-shifted
5  }
6  return(firstdiff(cumulate.data))
7  }
```

B.0.2 Modello di Bass Standard:

```
1  BASS.standard.generator<-function(m,p,q,tstart=1,n=50){
2  data=data.frame(tstart,0,0)
3  names(data)=c("time","sales","cumulative sales")
4  i=1
5  a=q/p
6  b=p+q
7  x=exp(-b*i)
8  y=1+(a*x)
9  adoption=m*((b^2)*x)/(p*(y^2))
10 for( i in 1:n){
11 data[i+1,1]=tstart+i
12 data[i+1,2]=adoption
13 data[i+1,3]=data[i,3]+data[i+1,2]
14 i=i+1
15 a=q/p
16 b=p+q
17 x=exp(-b*i)
18 y=1+(a*x)
19 adoption=m*((b^2)*x)/(p*(y^2))
20 }
21 return(data)
22 }
```

```

1  BASS.plot <-function(data){
2  par(mfrow=c(1,2))
3  plot(data[,1],data[,2],type="o",lwd=2,col="dark blue",xlab="[t]",ylab="Sales")
4  #par(new=TRUE)
5  plot(data[,1],data[,3],type="o",lwd=2,col="dark green",xlab="[t]",ylab="Cumulative sales")
6  par(mfrow=c(1,1))
7  }

```

```

1  BASS.standard <- function(sales,method='nls',prelimestimates=c(sum(sales)+100, 0.01, 0.1),alpha
   =0.05,display=T){
2  require(minpack.lm)
3  library(minpack.lm)
4  #alpha=0.05
5  t <- seq(1,length(sales),by=1)
6  s <- sales
7  c <- cumsum(s)
8  ff <- function(t,m,p,q) { m * (1 - exp(- (p+q)*t )) / (1 + q/p * exp(- (p+q)*t )) }
9  ff1 <- function(t,par) { c - ff(t,par[1],par[2],par[3]) }
10 ff2 <- function(t,par) { ff(t,par[1],par[2],par[3]) }
11 zprimo <- function(t,m,p,q){ m * ( p+q*(ff(t,m,p,q)/m) )*( 1 - (ff(t,m,p,q)/m) ) }
12 if(method=='nls'){
13 stime <- nls.lm(par=prelimestimates, fn=ff1, t=t)$par
14 sssss <- signif(summary(nls.lm(par=prelimestimates, fn=ff1,t=t))$coefficients,digits=3)
15 #print(sssss)
16 aa <- data.frame(summary(nls.lm(par=prelimestimates, fn=ff1,t=t))$coefficients[,c(1,2)],0,0,0)
17 names(aa) <- c('Estimate','Std.Error','Lower','Upper','p-value')
18 row.names(aa) <- c('m :','p :','q :')
19 for(i in 1:NROW(aa)){
20 aa[i,c(3,4)] <- aa[i,1]+c(-1,1)*qnorm(1-alpha/2)*aa[i,2]
21 }
22 aa[,5] <- sssss[,4]
23 }
24 else if(method=='optim'){
25 mass <- sum(s)+1000
26 ff3 <- function(par) { ff(t,par[1],par[2],par[3]) }
27 max <- sum(s)+10000
28 stima_optim <- function(c){
29 f <- function(p) { sum( (c - ff3(p) )^2 )}
30 optim(par = prelimestimates, fn = f ,method = "L-BFGS-B",lower = c
      (0.0000000001,0.000000001,0.000000001) , upper = c(mass,1,1))$par
31 }
32 stime <- stima_optim(c)
33 aa <- stime
34 }
35 if(display==T){
36 par(mfrow=c(1,2),mar=c(5, 4, 4, 2))
37 plot(t,c,main='Cumulative Sales of BM',xlim = c(0,max(t)+100),ylim = c(0,sum(s)+sum(s)*50/100),ylab
   = 'Cumulative')
38 curve(ff2(x,stime),add=T,col=2,xlim = c(0,max(t)+100))
39 plot(t,sales,main='Instantaneous Sales of BM',xlim = c(0,max(t)+100),ylab = 'Instantaneous')
40 curve(zprimo(x,stime[1],stime[2],stime[3]), col=2,add=T,xlim = c(0,max(t)+100))
41 par(mfrow=c(1,1),mar=c(5, 4, 4, 2))
42 }
43 s.hat <- ff2(t,stime)
44 tss <- sum( (c - mean(s))^2 )
45 rss <- sum( (c - s.hat)^2 )
46 r.squared <- 1-rss/tss
47 r.squared.adj <- 1 - ( (1- r.squared)*(length(s)-1) )/( length(s) - 1 - NROW(aa))
48
49 ##### restituisco i valori
50 list(Estimate=aa , Rsquared=r.squared , RsquaredAdj=r.squared.adj)
51 }

```


B.0.3 Modello di Bass Generalizzato:

```

1  BASS.generalized <- function(sales,shock=c('exp','rett','mixed','armonic'),nshock,prelimestimates
    ,alpha=0.05){
2  require(minpack.lm)
3  library(minpack.lm)
4  t <- seq(1,length(sales),by=1)
5  s <- sales
6  c <- cumsum(sales)
7  if(nshock==1){
8  if(shock=='exp'){
9  ##### definisco la funzione: integrale di x(t)
10 intx <- function(t,a1,b1,c1) {
11   ( t + c1*(1/b1)*(exp(b1*(t-a1)) - 1 )*(t>=a1) )
12 }
13 cat('##### Esponential shock ##### \n')
14
15 ##
16 xt <- function(t,a1,b1,c1){ 1 + (c1* exp(b1*(t-a1)))*(t>=a1) }
17 ##
18
19 }
20 else if (shock=='rett') {
21 ##### definisco la funzione: integrale di x(t)
22 intx <- function(t,a1,b1,c1) {
23   ( t + c1 * (t-a1) *(a1<=t)*(t<=b1) + c1*(b1-a1)*(b1<t) )
24 }
25 cat('##### Rectangular shock ##### \n')
26
27 ##
28 xt <- function(t,a1,b1,c1){ 1 + c1*(t>=a1)*(t<=b1)}
29 ##
30
31 }
32 else if (shock=='mixed'){
33 ##### definisco la funzione: integrale di x(t)
34 intx <- function(t,a1,b1,c1) {
35   t + c1*(1/b1)*(exp(b1*(t-a1)) - 1 )*(t>=a1)
36 }
37 cat('##### Mixed shock ##### \n')
38
39 ##
40 xt <- function(t,a1,b1,c1){ 1 + (c1* exp(b1*(t-a1)))*(t>=a1) }
41 ##
42
43 }
44 else if (shock=='armonic'){
45 ##### definisco la funzione: integrale di x(t)
46
47 intx <- function(t,a1,b1,c1) {
48   t + c1*( (b1-a1)/(2*pi) ) *cos( 2*pi*( (t-a1)/(b1-a1) ) )*(t>=a1)*(b<=b1)
49 }
50
51 cat('##### Armonic shock ##### \n')
52
53 ##
54 xt <- function(t,a1,b1,c1){
55   1 + c1*cos( 2*pi*( (t-a1)/(b1-a1) ) )*(t>=a1)*(b<=b1)
56 }
57 ##
58 }
59 ##### stimo e ritorno i risultati
60 ### !!!! cambia s in c quando passo i dati istantanei

```

```

61 ff <- function(t,m,p,q,a1,b1,c1) { m*(1 - exp(- (p+q)* intx(t,a1,b1,c1))) / (1 + (q/p) * exp(- (p
    +q) * intx(t,a1,b1,c1))) }
62 ff1 <- function(t,par) {c - ff(t,par[1],par[2],par[3],par[4],par[5],par[6])}
63 ff2 <- function(t,par) {ff(t,par[1],par[2],par[3],par[4],par[5],par[6])}
64 zprimo <- function(t,m,p,q,a1,b1,c1) {
65 m*(p+q*(ff(t,m,p,q,a1,b1,c1)/m))*(1-(ff(t,m,p,q,a1,b1,c1)/m))*xt(t,a1,b1,c1)
66 }
67 ##### ottengo stime e std.Error e IC
68
69 stime <- nls.lm(par=prelimestimates, fn=ff1, t=t)$par
70 aa <- data.frame(summary(nls.lm(par=prelimestimates, fn=ff1,t=t))$coefficients[,c(1,2)],0,0,0)
71 names(aa) <- c('Estimate','Std.Error','Lower','Upper','P-value')
72 row.names(aa) <- c('m :','p :','q :','a1 :','b1 :','c1 :')
73 for(i in 1:NROW(aa)){
74 aa[i,c(3,4)] <- aa[i,1]+c(-1,1)*qnorm(1-alpha/2)*aa[i,2]
75 }
76 sssss <- signif(summary(nls.lm(par=prelimestimates, fn=ff1,t=t))$coefficients,digits=3)
77 aa[,5] <- sssss[,4]
78
79 ##### plotto grafico dati e curva teorica
80
81 par(mfrow=c(1,2))
82 plot(t,c,xlim=c(0,length(t)+200),ylim = c(0,sum(s)+sum(s)*0.4),main='Cumulative Sales of GBM',
    ylab = 'Cumulative')
83 curve(ff2(x,stime),add=T,col=2)
84 plot(t,sales,main='Instantaneous Sales of GBM',xlim=c(0,length(t)+200),ylab = 'Instantaneous')
85 curve(zprimo(x,stime[1],stime[2],stime[3],stime[4],stime[5],stime[6]), col=2,add=T)
86 par(mfrow=c(1,1))
87
88 ##### mi calcolo R^2
89
90 s.hat <- ff2(t,stime)
91 tss <- sum( (c - mean(s))^2 )
92 rss <- sum( (c - s.hat)^2 )
93 r.squared <- 1-rss/tss
94 r.squared.adj <- 1 - ( (1- r.squared)*(length(s)-1) ) / ( length(s) - 1 - NROW(aa) )
95
96 ##### restituisco i valori
97
98
99 list(Estimate=aa , Rsquared=r.squared , RsquaredAdj=r.squared.adj , RSS=rss )
100 }
101 else if (nshock==2){
102 if(shock=='exp'){
103 ##### definisco la funzione: integrale di x(t)
104 intx <- function(t,a1,b1,c1,a2,b2,c2) {
105 t + c1*(1/b1)*(exp(b1*(t-a1)) - 1 )*(t>=a1) + c2*(1/b2)*(exp(b2*(t-a2)) - 1 )*(t>=a2)
106 }
107 cat('##### Esponential shock ##### \n')
108
109 ##
110 xt <- function(t,a1,b1,c1,a2,b2,c2){ 1 + (c1* exp(b1*(t-a1)))*(t>=a1) + (c2* exp(b2*(t-a2)))*(t
    >=a2) }
111 ##
112 }
113 else if (shock=='rett') {
114 ##### definisco la funzione: integrale di x(t)
115 intx <- function(t,a1,b1,c1,a2,b2,c2) {
116 t + c1 * (t-a1) *(a1<=t)*(t<=b1) + c1*(b1-a1)*(b1<t) + c2 * (t-a2) *(a2<=t)*(t<=b2) + c2*(b2-a2)*
    (b2<t)
117 }
118 cat('##### Rectangular shock ##### \n')
119
120 ##

```

```

121 xt <- function(t,a1,b1,c1,a2,b2,c2){ 1 + c1*(t>=a1)*(t<=b1) + c2*(t>=a2)*(t<=b2) }
122 ##
123
124 }
125 else if (shock=='mixed'){
126 ##### definisco la funzione: integrale di x(t)
127 intx <- function(t,a1,b1,c1,a2,b2,c2) {
128 t + (c1/b1)*(exp(b1*(t-a1)) - 1)*(a1<=t) + c2*(t-a2)*(a2<=t)*(t<=b2) + c2*(b2-a2)*(b2<t)
129 }
130 cat('##### Mixed shock ##### \n')
131
132 ##
133 xt <- function(t,a1,b1,c1,a2,b2,c2){ 1 + (c1* exp(b1*(t-a1)))*(t>=a1) + c2*(t>=a2)*(t<=b2) }
134 ##
135
136 }
137 else if (shock=='armonic'){
138 ##### definisco la funzione: integrale di x(t)
139
140 xt <- function(t,a1,b1,c1,a2,b2,c2){
141 1 + c1*cos( 2*pi*( (t-a1)/(b1-a1) ) )*(t>=a1)*(b<=b1) + c2*cos( 2*pi*( (t-a2)/(b2-a2) ) )*(t>=a2)
142 *(b<=b2)
143 }
144 intx <- function(t,a1,b1,c1,a2,b2,c2) {
145 t + c1*( (b1-a1)/(2*pi) ) *cos( 2*pi*( (t-a1)/(b1-a1) ) )*(t>=a1)*(b<=b1) +
146 c2*( (b2-a2)/(2*pi) ) *cos( 2*pi*( (t-a2)/(b2-a2) ) )*(t>=a2)*(b<=b2)
147 }
148 cat('##### Armonic shock ##### \n')
149
150 }
151 ##### stimo e ritorno i risultati
152 # ricorda cambiare s con c
153 ff <- function(t,m,p,q,a1,b1,c1,a2,b2,c2) { m*(1 - exp(- (p+q)* intx(t,a1,b1,c1,a2,b2,c2))) / (1
154 + (q/p) * exp(- (p+q) * intx(t,a1,b1,c1,a2,b2,c2))) }
155 ff1 <- function(t,par) {c - ff(t,par[1],par[2],par[3],par[4],par[5],par[6],par[7],par[8],par[9])}
156 ff2 <- function(t,par) {ff(t,par[1],par[2],par[3],par[4],par[5],par[6],par[7],par[8],par[9])}
157 #zprimo <- function(t,m,p,q,a1,b1,c1,a2,b2,c2) { m*(( p * ((p+q)^2) ) * xt(t,a1,b1,c1,a2,b2,c2) *
158 exp(- (p+q)* intx(t,a1,b1,c1,a2,b2,c2)))/( (p*exp(- (p+q) * intx(t,a1,b1,c1,a2,b2,c2)) + q
159 ^2 ) }
160 zprimo <- function(t,m,p,q,a1,b1,c1,a2,b2,c2) {
161 m*(p+q*(ff(t,m,p,q,a1,b1,c1,a2,b2,c2)/m))*(1-(ff(t,m,p,q,a1,b1,c1,a2,b2,c2)/m))*xt(t,a1,b1,c1,a2,
162 b2,c2)
163 }
164 ##### ottengo stime e std.Error e IC
165
166 stime <- nls.lm(par=prelimestimates, fn=ff1, t=t)$par
167 aa <- data.frame(summary(nls.lm(par=prelimestimates, fn=ff1,t=t))$coefficients[,c(1,2)],0,0,0)
168 names(aa) <- c('Estimate','Std.Error','Lower','Upper','P-value')
169 row.names(aa) <- c('m :','p :','q :','a1 :','b1 :','c1 :','a2 :','b2 :','c2 :')
170 for(i in 1:NROW(aa)){
171 aa[i,c(3,4)] <- aa[i,1]+c(-1,1)*qnorm(1-alpha/2)*aa[i,2]
172 }
173 sssss <- signif(summary(nls.lm(par=prelimestimates, fn=ff1,t=t))$coefficients,digits=3)
174 aa[,5] <- sssss[,4]
175
176 ##### plotto grafico dati e curva teorica
177 par(mfrow=c(1,2))
178 plot(t,c,xlim=c(0,length(t)+100),ylim = c(0,max(s)+5000),main='Cumulative Sales of GBM')
179 curve(ff2(x,stime),type='l',add=T,col=2)
180 plot(t,sales,main='Instantaneous Sales of GBM')
181 curve(zprimo(x,stime[1],stime[2],stime[3],stime[4],stime[5],stime[6],stime[7],stime[8],stime[9]),
182 col=2,add=T)

```

```

179 par(mfrow=c(1,1))
180 ##### mi calcolo R^2
181
182 s.hat <- ff2(t,stime)
183 tss <- sum( (c - mean(s))^2 )
184 rss <- sum( (c - s.hat)^2 )
185 r.squared <- 1-rss/tss
186 r.squared.adj <- 1 - ( (1- r.squared)*(length(s)-1) )/( length(s) - 1 - NROW(aa))
187
188 ##### restituisco i valori
189
190
191 list(Estimate=aa , Rsquared=r.squared , RsquaredAdj=r.squared.adj , RSS=rss )
192 }
193 else if (nshock==3){
194 if (shock=='exp' | shock=='rett' | shock=='armonic'){
195 if (shock=='exp'){
196 ##### definisco la funzione: integrale di x(t)
197 intx <- function(t,a1,b1,c1,a2,b2,c2,a3,b3,c3) {
198 t + c1*(1/b1)*(exp(b1*(t-a1)) - 1)*(t>=a1) + c2*(1/b2)*(exp(b2*(t-a2)) - 1)*(t>=a2) + c3*(1/b3)
199 *(exp(b3*(t-a3)) - 1)*(t>=a3)
200 }
201 cat('##### Esponential shock ##### \n')
202 ##
203 xt <- function(t,a1,b1,c1,a2,b2,c2){ 1 + (c1* exp(b1*(t-a1)))*(t>=a1) + (c2* exp(b2*(t-a2)))*(t
204 >=a2) + (c3* exp(b3*(t-a3)))*(t>=a3) }
205 ##
206 }
207 else if (shock=='rett') {
208 ##### definisco la funzione: integrale di x(t)
209 intx <- function(t,a1,b1,c1,a2,b2,c2,a3,b3,c3) {
210 ( t + c1 * (t-a1) *(a1<=t)*(t<=b1) + c1*(b1-a1)*(b1<t) + c2 * (t-a2) *(a2<=t)*(t<=b2) + c2*(b2-a2)
211 *(b2<t) + c3 * (t-a3) *(a3<=t)*(t<=b3) + c3*(b3-a3)*(b3<t) )
212 }
213 cat('##### Rectangular shock ##### \n')
214 ##
215 xt <- function(t,a1,b1,c1,a2,b2,c2){ 1 + c1*(t>=a1)*(t<=b1) + c2*(t>=a2)*(t<=b2) + c3*(t>=a3)*(t
216 <=b3) }
217 ##
218 }
219 else if (shock=='armonic'){
220 ##### definisco la funzione: integrale di x(t)
221
222 xt <- function(t,a1,b1,c1,a2,b2,c2,a3,b3,c3){
223 1 + c1*cos( 2*pi*( (t-a1)/(b1-a1) ) )*(t>=a1)*(b<=b1) + c2*cos( 2*pi*( (t-a2)/(b2-a2) ) )*(t>=a2)
224 *(b<=b2) + c3*cos( 2*pi*( (t-a3)/(b3-a3) ) )*(t>=a3)*(b<=b3)
225 }
226 intx <- function(t,a1,b1,c1,a2,b2,c2,a3,b3,c3) {
227 t + c1*( (b1-a1)/(2*pi) ) *cos( 2*pi*( (t-a1)/(b1-a1) ) )*(t>=a1)*(b<=b1) +
228 c2*( (b2-a2)/(2*pi) ) *cos( 2*pi*( (t-a2)/(b2-a2) ) )*(t>=a2)*(b<=b2) +
229 c3*( (b3-a3)/(2*pi) ) *cos( 2*pi*( (t-a3)/(b3-a3) ) )*(t>=a3)*(b<=b3)
230 }
231 cat('##### Armonic shock ##### \n')
232
233 }
234
235 ##### stimo e ritorno i risultati
236

```

```

237 ff <- function(t,m,p,q,a1,b1,c1,a2,b2,c2,a3,b3,c3) { m*(1 - exp(- (p+q)* intx(t,a1,b1,c1,a2,b2,c2
,a3,b3,c3))) / (1 + (q/p) * exp(- (p+q) * intx(t,a1,b1,c1,a2,b2,c2,a3,b3,c3))) }
238 ff1 <- function(t,par) {c - ff(t,par[1],par[2],par[3],par[4],par[5],par[6],par[7],par[8],par[9],
par[10],par[11],par[12])}
239 ff2 <- function(t,par) {ff(t,par[1],par[2],par[3],par[4],par[5],par[6],par[7],par[8],par[9],par
[10],par[11],par[12])}
240 zprimo <- function(t,m,p,q,a1,b1,c1,a2,b2,c2,a3,b3,c3) {
241 m*(p+q*(ff(t,m,p,q,a1,b1,c1,a2,b2,c2,a3,b3,c3)/m))*(1-(ff(t,m,p,q,a1,b1,c1,a2,b2,c2,a3,b3,c3)/m))
*xt(t,a1,b1,c1,a2,b2,c2,a3,b3,c3)
242 }
243
244 ##### ottengo stime e std.Error e IC
245
246
247 stime <- nls.lm(par=prelimestimates, fn=ff1, t=t)$par
248 aa <- data.frame(summary(nls.lm(par=prelimestimates, fn=ff1,t=t))$coefficients[,c(1,2)],0,0,0)
249 names(aa) <- c('Estimate','Std.Error','Lower','Upper','P-value')
250 row.names(aa) <- c('m :','p :','q :','a1 :','b1 :','c1 :','a2 :','b2 :','c2 :','a3 :','b3 :','c3
:')
251 for(i in 1:NROW(aa)){
252 aa[i,c(3,4)] <- aa[i,1]+c(-1,1)*qnorm(1-alpha/2)*aa[i,2]
253 }
254 sssss <- signif(summary(nls.lm(par=prelimestimates, fn=ff1,t=t))$coefficients,digits=3)
255 aa[,5] <- sssss[,4]
256
257 ##### plotto grafico dati e curva teorica
258
259 par(mfrow=c(1,2))
260 plot(t,c,xlim=c(0,length(t)+100),ylim = c(0,max(s)+5000) ,main='Cumulative Sales of GBM')
261 curve(ff2(x,stime),type='l',add=T,col=2)
262 plot(t,sales,main='Instantaneous Sales of GBM')
263 curve(zprimo(x,stime[1],stime[2],stime[3],stime[4],stime[5],stime[6],stime[7],stime[8],stime[9],
stime[10],stime[11],stime[12]),col=2,add=T)
264 par(mfrow=c(1,1))
265
266 ##### mi calcolo R^2
267
268 s.hat <- ff2(t,stime)
269 tss <- sum( (c - mean(s))^2 )
270 rss <- sum( (c - s.hat)^2 )
271 r.squared <- (tss-rss)/tss
272 r.squared.adj <- 1 - ( (1- r.squared)*(length(s)-1) )/( length(s) - 1 - NROW(aa) )
273
274 ##### restituisco i valori
275
276
277 list(Estimate=aa , Rsquared=r.squared , RsquaredAdj=r.squared.adj , RSS=rss )
278 }
279 else if (shock=='mixed'){
280 cat("--- Sorry you don't have 3 shocks in this case ---")
281 }
282 }
283 else{
284 cat("I'm sorry but we don't have implemented yet a model with more than 3 shocks.")
285 }
286 }

```

B.0.4 Modello con mercato potenziale variabile:

```

1  GG.model <- function(sales,prelimestimates=NULL,mt='base',alpha=0.05,...){
2  require(minpack.lm)
3  library(minpack.lm)
4  require(numDeriv)
5  library(numDeriv)
6  t <- seq(1,length(sales),by=1)
7  s <- sales
8  c <- cumsum(sales)
9
10
11  if(is.function(mt)==T){
12  ff <- function(t,k,ps,qs){
13  k*mt(t) * (1-exp(-(ps+qs)*t)) / (1+(qs/ps)*exp(-(ps+qs)*t))
14  }
15  ff1 <- function(t,par) {c - ff(t,par[1],par[2],par[3])}
16  ff2 <- function(t,par) {ff(t,par[1],par[2],par[3])}
17  #cat(is.null(prelimestimates)==TRUE)
18  if(is.null(prelimestimates)==TRUE){
19  prelimestimates <- BASS.standard(sales = s,display = F)$Estimate[,1]
20  }
21  ##### ottengo stime e std.Error e IC
22
23  stime <- nls.lm(par=prelimestimates, fn=ff1, t=t,control = nls.lm.control(maxiter = 100))$par
24  aa <- data.frame(summary(nls.lm(par=prelimestimates, fn=ff1,t=t,control = nls.lm.control(maxiter
25  = 100)))$coefficients[,c(1,2)],0,0,0)
26  names(aa) <- c('Estimate','Std.Error','Lower','Upper','P-value')
27  row.names(aa) <- c('k :','ps :','qs :')
28  for(i in 1:NROW(aa)){
29  aa[i,c(3,4)] <- aa[i,1]+c(-1,1)*qnorm(1-alpha/2)*aa[i,2]
30  }
31  sssss <- signif(summary(nls.lm(par=prelimestimates, fn=ff1,t=t,control = nls.lm.control(maxiter =
32  100)))$coefficients,digits=3)
33  aa[,5] <- sssss[,4]
34  }
35  else{
36  ### mercato potenziale variabile
37
38  mt <- function(t,k,pc,qc){
39  k*sqrt( (1-exp(-(pc+qc)*t)) / ( 1+(qc/pc)*exp( -(pc+qc)*t ) ) )
40  }
41
42  ### modello GUSEO GUIDOLIN cumulato
43
44  ff <- function(t,k,pc,qc,ps,qs){
45  mt(t,k,pc,qc) * (1-exp(-(ps+qs)*t)) / (1+(qs/ps)*exp(-(ps+qs)*t))
46  }
47
48  ##### stimo e ritorno i risultati
49
50  ff1 <- function(t,par) {c - ff(t,par[1],par[2],par[3],par[4],par[5])}
51  ff2 <- function(t,par) {ff(t,par[1],par[2],par[3],par[4],par[5])}
52
53  if(is.null(prelimestimates)==TRUE){
54  prelimestimates <- c(BASS.standard(sales = s,display=F)$Estimate[,1],1.00000e-03,1.00000e-01)
55  }
56
57  ##### ottengo stime e std.Error e IC
58
59  stime <- nls.lm(par=prelimestimates, fn=ff1, t=t,control = nls.lm.control(maxiter = 100))$par
60  aa <- data.frame(summary(nls.lm(par=prelimestimates, fn=ff1,t=t,control = nls.lm.control(maxiter
61  = 100)))$coefficients[,c(1,2)],0,0,0)
62  names(aa) <- c('Estimate','Std.Error','Lower','Upper','P-value')

```

```
60 row.names(aa) <- c('k :', 'pc :', 'qc :', 'ps :', 'qs :')
61 for(i in 1:NROW(aa)){
62 aa[i,c(3,4)] <- aa[i,1]+c(-1,1)*qnorm(1-alpha/2)*aa[i,2]
63 }
64 sssss <- signif(summary(nls.lm(par=prelimestimates, fn=ff1,t=t,control = nls.lm.control(maxiter =
100)))$coefficients,digits=3)
65 aa[,5] <- sssss[,4]
66
67 }
68 ##### plotto grafico dati e curva teorica
69
70 par(mfrow=c(1,2))
71
72 plot(t,c,main='Cumulative Sales of GGM',xlim=c(0,max(t)+200),ylim=c(0,ff2(max(t)+200,stime)+100),
73 ylab = 'Cumulative')
74 curve(ff2(x,stime),add=T,xlim=c(0,max(t)+200),...)
75 plot(t,sales,xlim=c(0,max(t)+200),main='Instantaneous Sales of GGM',ylab='Instantaneous')#,ylim=c
(0,50000),)
76 curve(grad(function(t) ff2(t,stime),x,method = 'simple'),add=T,...)
77
78 par(mfrow=c(1,1))
79
80 ##### mi calcolo R^2
81
82 s.hat <- ff2(t,stime)
83 tss <- sum( (c - mean(s))^2 )
84 rss <- sum( (c - s.hat)^2 )
85 r.squared <- 1-rss/tss
86 r.squared.adj <- 1 - ( (1- r.squared)*(length(s)-1) )/( length(s) - 1 - NROW(aa))
87
88 ##### restituisco i valori
89
90
91 list(Estimate=aa , Rsquared=r.squared , RsquaredAdj=r.squared.adj )
92 }
```


Bibliografia

- [1] **Frosini, B.V. Magagnoli, U. e Boari, G.**, *Studi in Onore di Angelo Zanella*, Vita e Pensiero, 2002, p 345-361
- [2] **Guseo, R.**, *Interventi strategici e aspetti competitivi nel ciclo di vita di innovazioni*, Unveristà di Padova Working Paper Series, 2004
- [3] **Guseo, R. e Gudolin, M.**, *Market potential dynamics in innovation diffusion: modelling the synergy between two driving forces*, Working Paper Series, 2009
- [4] **Rogers, E.M.**, *DIFFUSION OF INNOVATIONS*, The free press, 1983
- [5] **Pace, L. e Salvan, S.**, *Introduzione alla statistica - II. Inferenza, verosimiglianza, modelli*, Cedam, Padova, 2001
- [6] **Ligges, U.**, *R help desk: Package management*, R News, 2003, 3(3)
- [7] **Wickham, H.**, *R Packages: Organize, Test, Document, and Share Your Code*, O'Reilly Media, 2015
- [8] **Bittanti, S., Laub, A. e Willems, J.**, *The Riccati Equation*, Springer-Verlag Berlin, 1991
- [9] **Azzalini, A.**, *Inferenza statistica: una presentazione basata sul concetto di verosimiglianza*, Springer Italia, 2001.
- [10] Dispense didattiche di "Metodi statistici per l'azienda", A.A. 2017/2018. A cura della Prof.ssa Mariangela Guidolin