POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Lead scoring: modello di classificazione dei clienti di una insurtech

Tesi di Laurea Magistrale in
Mathematical Engineering - Ingegneria Matematica

Autore: **Edoardo Maria Palli**

Matricola: 976092
Relatore: Prof. Daniele Marazzina
Correlatore: Prof. Bruno Sfogliarini
Anno Accademico: 2022-23

Abstract

This work stems from the necessity of an insurtech start-up to profile the increasing number of users visiting their company website for the purpose of purchasing insurance policies. Customers who do not complete the entire process of policy subscription are subsequently contacted to finalize the process, receiving telephonic assistance from the in-house sales department or an external call center in collaboration with the company. The surge in leads, up to the point of saturating the company's resources for handling potential new clients, has led to the need to classify these leads into three main categories: the first involves no assistance from a consultant to the users, while the other two categories entail providing user support through either an in-house sales representative or an external call center.

To adequately classify each customer, an analysis of their key commercial characteristics, which are most significant, is necessary. These characteristics are used to identify the commercially most appealing prospects. The idea is to develop a lead scoring system aimed at establishing priority among new prospects through a coefficient representing the level of interest the company assigns to each of them.

Based on the data provided by users on the website, integrated with system data associated with each user, a logistic multinomial classification model has been designed. In this model, each user constitutes an observation in the dataset, and the response variable consists of three classes indicating their respective allocation modes: *Not Managed*, *Sales Department*, and *Call Center*. The model assigns a probability to each user belonging to each of the three categories, and the probability of belonging to the *Sales Department* category represents the lead's scoring.

Subsequently, the company's project, which extends beyond the scope of this thesis, involves transitioning from a logistic multinomial model to a machine learning model. This transition represents a significant improvement in the dynamics of learning from new data, reflecting the company's data-driven process automation policy. This non-parametric version of the model allows for greater flexibility and adaptability to the data, further enhancing the model's predictive performance.

However, the insurtech sector presents some unique characteristics that require a distinctive approach to data management. Specifically, the nature of the insurance products sold by the Lokky start-up and the needs of each customer imply that each user is interested in only one insurance product, unlike other sectors where a customer may be interested in multiple products. Secondly, the influx of new data into the company's database is often limited because once a user subscribes to a policy, they are unlikely to be interested in other insurance products before the expiration of their current policy, typically on an annual basis. Both dynamics necessitate a data enrichment approach to better explore the space of variables used as regressors in the machine learning model more comprehensively. Through this approach, it will be possible to acquire additional insights into user behavior and further enhance the predictive performance of both models.

Therefore, the thesis will be structured as follows: in the first chapter, the multinomial logistic model will be presented from both a theoretical and applied perspective, including the implementation of various structural variations and the selection of the model, along with the presentation of key results.

In the second chapter, the theme of data enrichment and the techniques used in generating synthetic data for database enrichment will be introduced. Theoretical discussions on the functioning of Bayesian networks, used in estimating conditional distributions of variables, will precede the presentation of the synthetic data generation algorithm. This will involve a detailed walkthrough of the main phases of the process, from estimating joint distributions to sampling synthetic data, and ultimately, the analysis and validation of data quality.

In the third chapter, the results of the multinomial logistic model trained with real data will be compared to those obtained with enriched data, using appropriate metrics to evaluate predictive capacity. In the initial phase, the real database was replicated in a general manner, and subsequently, the decision was made to enrich information related to a specific cluster of clients. The results produced in both contexts will be compared, discussing the optimal choice of parameters to be set in the data generation algorithm to ensure the best predictive performance.

Keywords: Lead scoring, multinomial logistic regression, data enrichment, synthetic data.

Sommario

Questo lavoro nasce dall'esigenza di una start-up nel settore insurtech di profilare i sempre più numerosi utenti che entrano nel sito aziendale per stipulare una polizza assicurativa. I clienti che non ultimano l'iter fino alla sottoscrizione della polizza vengono ricontattati per concludere il processo, assistiti telefonicamente dal reparto sales o dal call center esterno in collaborazione con l'azienda. L'aumento dei lead fino alla saturazione delle risorse aziendali per la gestione dei possibili nuovi clienti ha portarto alla necessità di classificare gli stessi in tre macrocategorie: la prima prevede di non affiancare alcun consulente agli utenti, nelle altre due invece si procede con un affiancamento del cliente rispettivamente ad un addetto alle vendite interno all'azienda o a un call center esterno. Per classificare in modo adeguato ciascun cliente è necessaria un'analisi delle principali caratteristiche dei clienti stessi che risultino commercialmente più significative, in base alle quali vengono individuati i soggetti commercialmente più appetibili. L'idea quindi è di sviluppare un lead scoring volto a stabilire una priorità tra i nuovi prospect tramite un coefficiente che simboleggi il grado di interesse che l'azienda attribuisce a ciascuno di essi.

Sulla base dei dati forniti dagli utenti sul sito online integrati con i dati di sistema associabili a ciascuno di essi, è stato progettato un modello di classificazione multinomiale logistico in cui ogni utente costituisce un'osservazione del dataset, e la variabile risposta è formata da tre classi per indicare la rispettiva modalità di allocazione: *Non gestito*, *Sales Interno* e *Call Center*. Il modello prevede l'assegnazione di una probabilità ad ogni utente di appartenere a ciascuna delle 3 categorie, e la probabilità di appartenere alla categoria *Sales Interno* rappresenta lo scoring del relativo lead.

Successivamente il progetto aziendale, che si estende al di fuori dei confini di questa tesi, prevede il passaggio da un modello multinomiale logistico a un modello di machine learning, transizione che rappresenta un importante miglioramento della dinamicità del processo di apprendimento dei nuovi dati, e in generale rispecchia una politica aziendale di automatizzazione dei processi data-driven. Questa versione non parametrica del modello, infatti, permette una maggiore flessibilità e capacità di adattamento ai dati. Inoltre, offre una maggiore flessibilità, peraltro già soddisfacente, nell'utilizzo delle variabili esplicative, migliorando così le performance predittive del modello stesso.

Tuttavia il settore insuretech presenta alcune peculiarità che rendono necessario un approccio singolare nella gestione dei dati. In particolare, la natura dei prodotti assicurativi venduti dalla startup Lokky e le esigenze di ciascun cliente implicano che ogni utente sia interessato a un solo prodotto assicurativo, a differenza di altri settori nei quali un cliente può essere interessato a diversi prodotti. In secondo luogo, il flusso di nuovi dati nel database aziendale è spesso limitato dal fatto che lo stesso utente, una volta sottoscritta la polizza, difficilmente sarà interessato ad altri prodotti assicurativi prima della scadenza della polizza stessa, solitamente di durata annuale. Entrambe le dinamiche rendono necessario un approccio di data enrichment al fine di esplorare in modo più completo lo spazio delle variabili utilizzate come regressori nel modello di machine learning. Grazie a questo approccio, sarà possibile acquisire ulteriori informazioni sul comportamento degli utenti e migliorare ulteriormente le performance predittive di entrambi i modelli.

Pertanto la tesi sarà strutturata nel modo seguente: nel primo capitolo viene presentato il modello logistico multinomiale sia dal punto di vista teorico che applicato al contesto di lavoro, l'implementazione di diverse varianti struttrali, la scelta del modello con l'esposizione dei principali risultati.

Nel secondo capitolo viene introdotto il tema del data enrichment e le tecniche utilizzate nella generazione dei dati sintetici volta all'arricchimento del database. In primo luogo viene trattato teoricamente il funzionamento delle reti bayesiane, utilizzate nella stima delle distribuzioni condizionate delle variabili. Segue la presentazione dell'algoritmo di generazione dei dati sintetici, percorrendo in modo dettagliato le fasi principali del processo, dalla stima della distribuzione congiunta al campionamento dei dati sintetici, fino all'analisi e alla validazione della qualità dei dati generati.

Nel terzo capitolo vengono confrontati i risultati del modello multinomiale logistico allenato con i dati reali e con i dati arricchiti, attraverso l'utilizzo di apposite metriche di valutazione della capacità predittiva. In una prima fase il database reale è stato replicato in maniera generale, successivamente si è scelto di arricchire le informazioni relative ad un cluster specifico di clienti. Sono stati confrontati i risultati prodotti in entrambi i contesti, discutendo la scelta ottimale dei parametri da fissare nell'algoritmo di generazione dei dati al fine di garantire le performance predittive migliori.

Parole chiave: Lead scoring, regressione multinomiale logistica, data enrichment, dati sintetici.

Indice

Introduzione

Il problema della classificazione delle osservazioni in categorie ha sempre suscitato grande interesse nell'ambito della statistica e della matematica applicata sia dal punto di vista accademico sia per l'indiscutibile utilità dal punto di vista pratico. Il fatto di assegnare una categoria agli individui di una popolazione, formando in questo modo gruppi distinti all'interno di un campione statistico, permette di evidenziare le caratteristiche tipiche che accomunano i membri di un determinato gruppo e che li differenziano invece dagli altri.

Tale problema può essere declinato in diversi modi a seconda del contesto in cui viene inquadrato, motivo per il quale presenta per sua natura una complessità tale da richiedere una adeguata attenzione nella formulazione del problema stesso. In primo luogo la scelta del numero dei diversi gruppi di individui può essere fissato a priori, oppure può essere sconosciuto e si vuole investigare l'esistenza eventuale di sottopopolazioni distinte all'interno del campione statistico rispetto alle variabili di cui sono corredate. Quest'ultimo caso nella letteratura scientifica è identificato con il nome di clustering e fa parte della famiglia di tecniche di unsupervised learning, e non sarà oggetto di trattazione in questa tesi. Il primo caso invece richiede la definizione delle diverse categorie che è possibile identificare all'interno del campione statistico e alle quali ciascuna osservazione verrà assegnata, sulla base dlle variabili che sono state identificate come significative nella classificazione. Queste tecniche a differenza del caso precedente fanno parte dei metodi di supervised learning e saranno approfondite in questa tesi.

Questo lavoro si inserisce in un contesto specifico, quello di un'azienda che opera nel settore insuretech, Lokky, intermediario assicurativo che si rivolge a clienti business, con la promozione di prodotti specificatamente progettati per le microimprese e per i lavoratori autonomi. Dalla natura stessa di Lokky, piccola impresa e di giovane età, nasce l'esigenza di allocare in maniera efficiente le proprie risorse nella fase di finalizzazione della vendita delle polizze assicurative. In particolare l'azienda in questione è sviluppata quasi interamente online, attraverso un sito web per mezzo del quale avviene l'interazione con i lead, integrato con un sistema automatico di preventivazione, grazie al quale avviene la formulazione di un preventivo online immediato su richiesta del cliente: selezionando il prodotto assicurativo desiderato e inserendo i dati personali che sono richiesti per la sot-

toscrizione della polizza, è possibile ricevere un preventivo personalizzato per assicurare la propria impresa e sottoscrivere la polizza direttamente online. Per aumentare il volume delle vendite la sezione commerciale dell'azienda si occupa di ricontattare telefonicamente i prospect che non hanno completato l'iter per intero nel tentativo di finalizzare la vendita. Da questi presupposti nasce l'esigenza di assegnare una priorità ai lead che sono considerati commercialmente più interessanti e classificarli in tre categorie: coloro che non devono essere ricontattati, che devono essere contattati dal reparto vendite interno all'azienda oppure da un call center esterno. I dati relativi ai profili di cui l'azienda dispone sono forniti dai clienti stessi e integrati con la raccolta delle informazioni di sistema (come il dispositivo utilizzato, la campagna pubblicitaria per mezzo del quale è stato acquisito il cliente), più orientate alla descrizione strutturale degli utenti a causa dell'involontarietà con il quale sono prodotte e arricchiscono enormemente il patrimonio informativo di cui l'azienda dispone perché conservano caratteristiche che hanno maggiormente a che vedere con il profilo comportamentale degli stessi. La raccolta di questi dati avviene secondo i termini di legge attraverso l'autorizzazione concessa dagli utenti stessi.

Per entrare maggiormente nel dettaglio i modelli di supervised learning mappano lo spazio delle covariate con le label della variabile categorica rispetto alla quale avviene la classificazione. Nel corso degli anni è stata sviluppata una vasta teoria sulla costruzione di tale funzione nel quale è possibile identificare due filoni concettualmente opposti nell'approccio al problema. I primi, anche in ordine cronologico, sono i modelli parametrici, nei quali si assume a priori una formulazione che leghi la variabile categorica di output con le covariate. Tale equazione assume una scrittura generale in quanto dipende da un insieme di parametri, chiamati parametri del modello, che vengono stimati massimizzando l'adattamento delle categorie ai dati osservati. Pertanto la mappa di classificazione è identificata univocamente attraverso l'assegnazione dei parametri del modello. In questo contesto è dunque possibile riformulare in maniera equivalente il problema di classificazione in un problema di ottimizzazione dei parametri. Tali modelli possono mostrare diversi gradi di flessibilità nell'adattamento ai dati, in base alla formulazione matematica e alla quantità di parametri che sono stati ipotizzati, e garantiscono buone performance dal punto di vista predittivo.

Di contro, si è sviluppata negli ultimi anni una teoria parallela ed altrettanto affascinante sui modelli di classificazione non parametrici, ovvero nei quali non viene fatta alcuna ipotesi riguardante il legame tra la variabile di output e le covariate, bensì la mappa di classificazione viene costruita attraverso tecniche di machine learning a partire dai dati osservati. L'obiettivo in questo contesto è quello di raggiungere un maggiore adattamento del classificatore ai dati e performance predittive migliori, a discapito di una perdita di

interpretabilità nel modello stesso, in cui è più difficile comprendere i legami tra la variabile di output e le covariate, e quindi come quest'ultime influenzino la prima. Alcuni esempi di algoritmi di machine learning sono gli alberi di classificazione (CART), i boosted tree e le random forest.

La natura diversa di questi due approcci si riflette anche a livello applicativo sui dati a disposizione: i modelli parametrici possono essere adattati a campioni non necessariamente numerosi, garantendo risultati soddisfacenti. Inoltre un'aumento della popolazione nel campione non corrisponde necessariamente ad una stima più accurata dei parametri, soprattutto se il training set risulta statisticamente rappresentativo. L'aggiunta di osservazioni che potrebbero delineare nuovi profili con caratteristiche specifiche, formando in pratica un nuovo cluster, potrebbero non essere modellati con precisione se la loro numerosità non incide significativamente su quella della popolazione.

D'altro canto la capacità dei modelli di machine learning di adattarsi in modo esaustivo ai dati rende più sensibile, e dunque più accurata, la classificazione in presenza di alcuni particolari raggruppamenti, intesi come sottopopolazioni dalle caratteristiche ricorrrenti, senza che siano necessariamente popolosi. Nel caso di Lokky eventuali cluster significativi sono rappresentati da profili di utenti o clienti, che ad esempio nella compilazione dei questionari per la sottoscrizione del preventivo interrompono l'interazione con il sito in determinati punti, o accedono all'area personale in fasce orarie ricorrenti. Tuttavia ad oggi Lokky dispone di un database ridotto e scarsamente popolato per due motivi principali, che complicano la possibilità di progettare un algoritmo di machine learning adeguato: il primo è rappresentato dalla giovane vita aziendale di Lokky, startup fondata nel recente 2018, motivo per cui la numerosità dei clienti è ridotta. Inoltre i database aziendali hanno subito nel corso degli anni alcune variazioni strutturali, che rendono a volte incompatibile l'utilizzo di dati relativi a periodi temporali distinti, per l'assenza di allineamento di alcuni campi; la seconda motivazione è invece intrinseca nel settore in cui Lokky si inserisce, ovvero quello assicurativo, che presenta delle peculiarità nel flusso e nella raccolta delle informazioni dal sito web: in particolare, la natura dei prodotti assicurativi venduti da Lokky e le esigenze di ciascun cliente implicano che ogni lead sia interessato a un prodotto assicurativo specifico relativo alla propria professione, a differenza di altri contesti nei quali un cliente può essere interessato a diversi servizi, rappresentando una prima limitazione nel ventaglio degli interessi dei target. In secondo luogo, il flusso di nuovi dati nel database aziendale è spesso limitato dal fatto che lo stesso utente, una volta sottoscritta la polizza, difficilmente sarà interessato ad un altro prodotto prima della scadenza della polizza stessa, solitamente di durata annuale, limitando la velocità con la quale Lokky aggiorna le proprie conoscenze in merito ai customer.

Alla luce delle problematiche emerse è necessario un approccio orientato all'arricchimento del patrimonio informativo aziendale attraverso la generazione di dati sintetici, ovvero osservazioni artificiali create ad hoc che replichino fedelmente la struttura dei dati reali. A partire dalla distribuzione congiunta di questi ultimi, stimata con tecniche di statistica Bayesiana, avviene un campionamento di osservazioni dalla numerosità arbitraria volto ad ampliare il database e eventualmente a riempire in maniera mirata le lacune conoscitive rispetto ad alcuni target.

A tale scopo nelle prossime sezioni verrà sviluppato questo processo, a partire dal primo riguardante la scelta di un classificatore parametrico che sia compatibile con la struttura dei dati di Lokky e che garantisca delle performance soddisfacenti dal punto di vista predittivo, discutendone i risultati. Nel secondo capitolo invece verrà esposto in modo dettagliato l'algoritmo utilizzato per la generazione dei dati sintetici, fornendo delle conoscenze teoriche di base sia sulla teoria Bayesiana che sulla teoria dell'informazione alle quali si è attinto per ricostruire i legami di dipendenza tra le variabili reali, e saranno valutate alcune metriche volte alla valutazione della qualità dei dati generati. Infine nel terzo capitolo sarà valutato l'impatto dei dati sintetici sulle performance del modello parametrico, discutendone eventuali differenza tra l'uso dei dati reali e l'integrazione con i dati sintetici.

1 | Lead scoring: modello parametrico

In questo capitolo sarà introdotto il tema centrale del progetto di classificazione degli utenti per l'azienda Lokky e sviluppato un primo modello in forma embrionale, partendo da un inquadramento più preciso di cosa significhi classificare un cliente per poi analizzare la scelta del modello parametrico utilizzato e i principali concetti matematici su cui si basa. Inoltre verrano analizzati i risultati acquisiti sia in termini di fitting del modello che a livello predittivo su un database di test.

1.1. Scelta e formulazione del modello parametrico

Il patrimonio informativo di Lokky sugli utenti che si interfacciano con il sito web è formato essenzialmente da dati personali del sottoscrittore della polizza, da informazioni che riguardano l'attività assicurata e da caratteristiche che è possibile estrarre dall'interazione che un cliente ha prodotto sul sito e dal suo comportamento, e suggeriscono la distinzione tra dati che sono stati volontariamente forniti e informazioni che invece sono tracciate dal sito web e che ne delineano un aspetto comportamentale. In questo modo a ciascun utente è associato un insieme di caratteristiche codificate in variabili sia numeriche che categoriche, e l'insieme degli utenti popola il database dell'azienda. Il database è quindi formato da numerose osservazioni, ciascuna delle quali rappresenta un utente univocamente identificato dall'email che inserisce nel sito.

1.1.1. Inquadramento generale del problema in analisi

L'esigenza di distribuire in maniera ottimale la capacità di vendita offline delle polizze descritta nell'introduzione si traduce nella variabile *Allocazione* degli utenti, composta da tre classi: coloro che non devono essere ricontattati, coloro che devono essere contattati da un call center esterno all'azienda e coloro che invece devono essere affiancati da un consulente interno. Tale variabile sarà l'output del modello oggetto di questa tesi. L'allocazione

degli utenti già presenti a database al gruppo di appartenenza avviene seguendo la logica di assegnazione passata, ovvero replicando come sono stati effettivamente classificati nel momento in cui è avvenuta la vendita del prodotto assicurativo o la preventivazione, senza una indagine più approfondita su eventuali classificazioni commercialmente più vantaggiose: questa scelta è motivata dal fatto che l'output è a tutti gli effetti una variabile categorica nominale. Infatti un eventuale ordinamento rispetto alla priorità commerciale dei gruppi risulta non ben definito: ad esempio un cliente classificato *Non gestito* può essere commercialmente sia molto appetibile per Lokky, in quanto non necessita di un supporto nella sottoscrizione della polizza, sia di minimo interesse, e quindi da non meritare l'impiego di risorse aziendali. Dalla natura nominale delle categorie di allocazione dei clienti discende una difficoltà nello stabilire un criterio per valutare eventuali errori di classificazione, e di quantificare eventuali costi di misclassificazione che potrebbero essere inseriti nell'algoritmo. Inoltre la logica di replicazione delle label passate senza indagine di errore costituisce la versione più semplice del modello, perciò è stato deciso di procedere per gradi di difficoltà nell'implementazione del lead scoring. Tuttavia è importante sottolineare come questo aspetto rappresenta la lacuna principale del modello, nonchè una sfida da affrontare per migliorare la classificazione dei clienti.

Le altri variabili considerate in questa analisi sono la *email*, la *recency* che rappresenta quanto tempo è trascorso dall'interazione dell'utente sul sito, il *canale* di acquisizione, ovvero il mezzo attraverso il quale un utente è entrato nel sito di Lokky, i *tempi* di completamento dei questionari assicurativi, il fatto che l'utente si sia *registrato* al sito o meno, lo *stato* di avanzamento della sottoscrizione della polizza, il *prodotto* selezionato, il *settore* lavorativo di appartenenza nel quale si identifica, il *dispositivo* elettronico utilizzato, il numero di *preventivi* e di *polizze* sottoscritte, con il dettaglio sul loro *premio medio*. La tabella 1.1 fornisce una sintesi descrittiva. Variabili personali come l'età, il setto, la regione di provenienza, e altri variabili generali come il giorno in cui è avvenuta l'interazione tra il cliente e Lokky sono risultate ininfluenti ai fini della classificazione e dunque non sono state riportate in tabella.

1.2. Modello multinomiale logistico

Il modello multinomiale logistico è la generalizzazione del modello logistico e viene utilizzato quando la variabile di output è una variabile categorica con più di due classi, nominali e mutuamente esclusive, e risulta particolarmente utile se si vuole stimare le probabilità di appartenenza delle osservazioni a ciascuna classe in base ai regressori esplicativi.

Nome variabile	Tipo	Valori
Allocazione	Categorica	{Non gestito, Call Center, Sales Interno}
Dominio mail	Categorica	{Aziendale, Personale, Altro}
Recency	Numerica	$[0, +\infty)$
Canale	Categorica	Canale di acquisizione del sito Lokky
Tempi completamento questionari (4)	Numerica	$[0, +\infty)$
Registrato	Categorica	{sì, no}
Stato utente	Categorica	Stato di avanzamento acquisizione della polizza
Prodotto	Categorica	Prodotto assicurativo selezionato
Settore	Categorica	Settore lavorativo di apparteneza
Dispositivo	Categorica	{Mobile, Desktop}
Preventivi	Numerica	$\{0, 1, 2, 3, ...\}$
Polizze	Numerica	$\{0, 1, 2, 3, ...\}$
Premio medio	Numerica	$[0, +\infty)$

Tabella 1.1: Variabili del dataset

1.2.1. Formulazione del modello

Sia $A = \{X_1, ..., X_d\}$ un insieme di variabili aleatorie sia numeriche che categoriche, che saranno chiamati regressori, sia Y una variabile categorica con K categorie mutuamente esclusive. Per semplicità supponiamo che Y abbia valori in $\{0, 1, ..., K-1\}$. Il modello multinomiale logistico estende il concetto di regressione logistica costruendo $K-1$ modelli logistici indipendenti che si riferiscono alla medesima categoria $Y = 0$, chiamata classe di riferimento. In ciascuno di essi viene modellato il legame probabilistico di appartenenza alla classe $k \in \{1, ..., K-1\}$ ipotizzato un modello lineare nei coefficienti β tra i regressori e il logaritmo del rapporto tra la probabilità di appartenenza al gruppo k-esimo $(Y = k)$ e a quello di riferimento:

$$\begin{aligned}\text{logit}\, P(Y=k|X_1,...,X_d) &= \log\frac{P(Y=k|X_1,...,X_d)}{P(Y=0|X_1,...,X_d)}, \qquad k=1,...,K-1.\\ \text{logit}\, P(Y=k|X_1,...,X_d) &= \beta_{k0}+\beta_{k1}X_1+\beta_{k2}X_2+...+\beta_{kd}X_d, \quad k=1,...,K-1.\end{aligned} \tag{1.1}$$

Dalle equazioni (1.1) si deduce che:

$$P(Y=0|X_1,...,X_d) = \frac{1}{1+\sum_{k=1}^{K-1}\exp\left(\beta_{k0}+\beta_{k1}X_1+...+\beta_{kd}X_d\right)}, \tag{1.2}$$

$$P(Y=k|X_1,...,X_d) = \frac{\exp(\beta_{k0}+\beta_{k1}X_1+...+\beta_{kd}X_d)}{1+\sum_{k=1}^{K-1}\exp\left(\beta_{k0}+\beta_{k1}X_1+...+\beta_{kd}X_d\right)}, \quad k=1,...,K-1. \tag{1.3}$$

1.2.2. Stima dei coefficienti

In questa sezione sarà esplorato il calcolo dei coefficienti del modello multinomiale logistico, facendo riferimento ai testi [1] e [6]. Supponiamo che il nostro dataset D sia un campione di n osservazioni indipendenti dei regressori in A e di Y, ovvero:

$$Y_1,...,Y_n \overset{\text{iid}}{\sim} Multinomial((p_0,...,p_{K-1}),1) \tag{1.4}$$

$$X_{1j},...,X_{nj} \overset{\text{iid}}{\sim} X_j, \quad j=1,...,d. \tag{1.5}$$

Si ipotizzi inoltre che ciascuna Y_i condizionatamente alla d-tupla $X_{i1},...,X_{id}$ segua la legge descritta dall'equazione (1.1), $i=1,...,n$. Per costruire la funzione di verosomiglianza creiamo per la i-esima osservazione ($i=1,...,n$) K variabili binarie ausiliarie $Y_{i0},Y_{i1},...,Y_{iK-1}$ che assumono valori 0 o 1 per indicare l'appartenenza alla categoria corrispondente: se ad esempio $Y_i=2$ allora $Y_{ik}=0$ per $k\neq 2, Y_{i2}=1$. Si noti che in questo modo per qualunque valore assunto da Y_i segue che $\sum_{k=0}^{K-1}Y_{ik}=1$. Definiamo le seguenti funzioni:

$$g_k(\mathbf{x}) = \mathbf{x}\cdot\boldsymbol{\beta_k}, \quad \boldsymbol{\beta_k}=(\beta_{k0},\beta_{k1},...,\beta_{kd}), \quad k=1,...,K-1, \tag{1.6}$$

$$g_0(\mathbf{x}) = 0, \quad \boldsymbol{\beta_0}=\mathbf{0}, \tag{1.7}$$

$$\pi_k(\boldsymbol{\beta}|\mathbf{x}) = P(Y=k|\mathbf{x}), \quad \boldsymbol{\beta}=(\boldsymbol{\beta_0},\boldsymbol{\beta_1},...,\boldsymbol{\beta_{K-1}}), \quad k=0,...,K-1. \tag{1.8}$$

Il contributo alla log-likelihood dell'osservazione i, considerando che $\sum_{k=0}^{K-1} Y_{ik} = 1, \ \forall i = 1, ..., n$, e $\sum_{k=0}^{K-1} \pi_k(\boldsymbol{\beta}|\mathbf{x}) = 1$, è il seguente:

$$\begin{aligned} l(\boldsymbol{\beta}; \mathbf{x_i}) &= \log\Big[\prod_{k=0}^{K-1} \pi_k(\boldsymbol{\beta}|\mathbf{x_i})^{Y_{ik}}\Big] \\ &= \sum_{k=1}^{K-1} Y_{ik} \log \pi_k(\mathbf{x_i}) + \Big(1 - \sum_{k=1}^{K-1} Y_{ik}\Big) \log\Big[1 - \sum_{k=1}^{K-1} \pi_k(\mathbf{x_i})\Big] \\ &= \sum_{k=1}^{K-1} Y_{ik} \log \frac{\pi_k(\mathbf{x_i})}{1 - \sum_{k=1}^{K-1} \pi_k(\mathbf{x_i})} + \log\Big[1 - \sum_{k=1}^{K-1} \pi_k(\mathbf{x_i})\Big]. \end{aligned}$$

La likelihood è definita nel seguente modo:

$$L(\boldsymbol{\beta}; \mathbf{x}) = \prod_{i=1}^{n} \prod_{k=0}^{K-1} \pi_k(\boldsymbol{\beta}|\mathbf{x_i})^{Y_{ik}}, \tag{1.9}$$

e di conseguenza la log-likelihood:

$$l(\boldsymbol{\beta}; \mathbf{x}) = \log L(\boldsymbol{\beta}; \mathbf{x}) = \sum_{i=1}^{n} \sum_{k=0}^{K-1} Y_{ik} \pi_k(\boldsymbol{\beta}|\mathbf{x_i}). \tag{1.10}$$

Usando le equazioni (1.6) e (1.8) la log-likelihood diventa:

$$l(\boldsymbol{\beta}; \mathbf{x}) = \sum_{i=1}^{n} \Big[\sum_{k=1}^{K-1} Y_{ik} g_k(\mathbf{x_i})\Big] - \log\Big(1 + \sum_{k=1}^{K-1} e^{g_k(\mathbf{x_i})}\Big). \tag{1.11}$$

Il massimo rispetto a $\boldsymbol{\beta}$ è raggiunto applicando le derivate parziali di $l(\boldsymbol{\beta}; \mathbf{x})$ rispetto a ciascuno dei $(d+1)(k-1)$ parametri e ponendo ciascuna equazione a 0. La forma generale diventa:

$$\frac{\partial l(\boldsymbol{\beta}; \mathbf{x})}{\partial \beta_{kj}} = \sum_{i=1}^{n} x_{ij}[Y_{ik} - \pi_k(\mathbf{x_i})] = 0, \quad \forall k = 1, ..., K-1, \quad \forall j = 0, ..., d, \tag{1.12}$$

con $x_{i0} = 1$ per ogni osservazione. Lo stimatore di massima verosomiglianza $\hat{\boldsymbol{\beta}}$ si ottiene risolvendo il sistema descritto dall'equazione (1.12) rispetto a $\boldsymbol{\beta}$.

Inoltre è necessario calcolare la matrice delle derivate parziali di secondo grado per ottenere la matrice di informazione e la matrice varianza dello stimatore di massima veroso-

miglianza. La forma generale degli elementi della matrice delle derivate parziali seconde è:

$$\frac{\partial^2 l(\boldsymbol{\beta};\mathbf{x})}{\partial\beta_{kj}\partial\beta_{kj'}} = -\sum_{i=1}^{n} x_{ij'}x_{ij}\pi_k(\mathbf{x_i})(1-\pi_k(\mathbf{x_i})), \tag{1.13}$$

$$\frac{\partial^2 l(\boldsymbol{\beta};\mathbf{x})}{\partial\beta_{kj}\partial\beta_{k'j'}} = \sum_{i=1}^{n} x_{ij'}x_{ij}\pi_k(\mathbf{x_i})\pi_{k'}(\mathbf{x_i}). \tag{1.14}$$

per k e $k' = 1, ..., K-1$ e j e $j' = 0, ..., d$. La matrice di informazione osservata, $\mathbf{I}(\hat{\boldsymbol{\beta}})$, è la matrice $(K-1)(d+1)\times(K-1)(d+1)$ i cui elementi sono i valori negativi delle equazioni (1.13) e (1.14) valutati in $\hat{\boldsymbol{\beta}}$. Lo stimatore della matrice varianza è l'inverso della matrice di informazione, ovvero:

$$\widehat{\mathbf{Var}(\hat{\boldsymbol{\beta}})} = \mathbf{I}(\hat{\boldsymbol{\beta}})^{-1}. \tag{1.15}$$

1.2.3. Interpretazione dei coefficienti

I coefficienti beta nel modello multinomiale logistico sono parametri cruciali nella comprensione dell'effetto dei predittori sulle diverse categorie di outcome. Essi consentono di valutare il contributo relativo di ciascuna variabile predittiva nella determinazione della probabilità di appartenenza a una specifica categoria di risposta. In questa sezione, esamineremo l'interpretazione dei coefficienti beta e come possiamo utilizzarli per comprendere l'importanza delle variabili nel nostro modello.

È necessario distinguere due casi fondamentali al fine di attribuire un'interpretazione accurata ai coefficienti del modello: il caso in cui la variabile associata al coefficiente in questione sia numerica o categorica. Nel primo sia il regressore X_1 numerico, la quantità $\exp\beta_{k1} - 1$ rappresenta l'incremento relativo del rapporto tra la probabilità di appartenenza alla classe k rispetto alla classe di riferimento $Y = 0$ a fronte di un incremento unitario di X_1, fissando il valore delle restanti covariate. In particolare:

$$\frac{\frac{\hat{P}(Y=k|x_1+1)}{\hat{P}(Y=0|x_1+1)} - \frac{\hat{P}(Y=k|x_1)}{\hat{P}(Y=0|x_1)}}{\frac{\hat{P}(Y=k|x_1)}{\hat{P}(Y=0|x_1)}} = e^{\beta_{k1}} - 1. \tag{1.16}$$

Ciò implica che il segno di β_{k1} indica un aumento della probabilità di appartenenza alla classe k rispetto alla classe di riferimento a causa dell'aumento di X_1 nel caso in cui β_{k1}

sia positivo, viceversa se β_{k1} è negativo tale rapporto probabilistico diminuisce al crescere di X_1.

Se invece il regressore X_1 è categorico, supponiamo di assegnare a ciascuna categoria un numero naturale $1, 2, ..., k$ dove $k = |\operatorname{supp} X_1|$. In questo caso la quantità $e^{\beta_{k1}(j_1 - j_2)}$ rappresenta il rapporto tra l'odds ratio di appartenenza alla categoria k nel caso in cui X_1 appartenga a j_1 e j_2, tenendo sempre fissi i valori degli altri regressori:

$$\frac{\frac{\hat{P}(Y=k|x_1=j_1)}{\hat{P}(Y=0|x_1=j_1)}}{\frac{\hat{P}(Y=k|x_1=j_2)}{\hat{P}(Y=0|x_1=j_2)}} = e^{\beta_{k1}(j_1 - j_2)}. \tag{1.17}$$

Se i due gruppi j_1 e j_2 sono "vicini", ovvero $j_1 - j_2 = 1$, il segno di β_{k1} è sufficiente a dedurre un aumento nel caso sia positivo, o una diminuzione del rapporto di probabilità di appartenere alla classe k. Si noti che questo è il caso per esempio delle variabili categoriche dicotomiche.

In entrambi i casi invece l'intensità del coefficiente beta indica l'entità dell'effetto della variabile predittiva sulla probabilità di appartenenza a una specifica categoria di risposta. Un coefficiente con un valore assoluto maggiore indica un effetto più forte, mentre un coefficiente con un valore assoluto minore indica un effetto più debole. Tale interpretazione è molto utile per capire l'impatto e quindi l'importanza di una determinata variabile nel modello.

Per concludere i coefficienti β_{k0}, $k = 1, ..., K - 1$ non sono associati ad alcun regressore e sono chiamate intercette del modello. La quantità $e^{\beta_{k0}}$ rappresenta il rapporto tra la probabilità di appartenenza alla classe k e la classe di riferimento nel caso in cui tutte le altre variabili siano spente.

1.3. Fitting del modello

Per quanto riguarda le tecniche implementative del modello, le variabili indipendenti utilizzate e eventuali trasformazioni dei regressori sono state esaminate diverse opzioni. Il modello è stato progettato seguendo idee implementative varie nella formulazione delle covariate e nel fitting, nel tentativo di costruire un algoritmo che garantisse le migliori performance dal punto di vista previsionale e che modellasse ottimamente i legami tra le variabili. In particolare sono stati considerati diversi modelli multinomiali:

1. **Modello lineare nelle covariate:** tale modello rappresenta la forma più semplice nella formulazione, in cui viene ipotizzato un legame lineare sia nei coefficienti che

nei regressori tra i log-odds delle probabilità di appartenenza alle classi di output e le variabili indipendenti.

2. **Modello lineare nelle covariate con interazioni**: rispetto al precedente caso sono aggiunti i termini di interazione tra variabili numeriche e categoriche fino al secondo ordine al fine di modellare con maggiore dettaglio l'appartenenza degli utenti alle diverse classi.

3. **Modello multinomiale pesato**: l'idea è di attribuire un peso differente a ciascuna osservazione del dataset nella stima delle probabilità finali, al fine di modellare l'impatto che osservazioni più o meno importanti hanno sul modello.

1.3.1. Modello non pesato

Modello lineare nelle covariate

Il primo modello implementato prevede un legame lineare nei regressori, oltre che nei coefficienti, che è possibile formulare nel seguente modo, supponendo che le prime 7 variabili siano categoriche e le restanti 8 siano numeriche:

$$\text{logit}\, P(Y = k|X_1, ..., X_d) = \beta_{k0} + \beta_{k1}^{J_1} + \beta_{k2}^{J_2} + ... + \beta_{i7}^{J_7} + \beta_{k8}X_8 + ... + \beta_{k15}X_{15}, \quad (1.18)$$

$$k = 1, 2,\ j = 1, ..., 7,\ J_j = 1, ..., |J_j|.$$

È possibile notare che le variabili categoriche incidono solo in maniera additiva nel modello attraverso le costanti $\beta_{kj}^{J_j}$ a seconda dei gruppi a cui ciascun osservazione appartiene, modificando l'intercetta.

Modello con interazioni

Dal momento che l'insieme delle variabili è formato da regressori sia numerici sia categorici, sono stati aggiunti dei termini di interazione tra di esse: inizialmente il modello proposto era formato da tutti i termini lineari nei regressori e tutte le interazioni tra variabili numeriche e categoriche con al massimo due classi, quindi *Registrato* e *Dispositivo*. Il motivo di tale scelta rispetto a considerare più banalmente tutti i termini lineari e tutte le interazioni, risiede nel tentativo di evitare un eccessivo adattamento ai dati del nostro modello che potrebbe causare il fenomeno di overfitting. Aggiungere termini quadratici tra variabili numeriche, termini di interazione tra variabili categoriche o tra regressori

numerici con variabili categoriche troppo spefiche potrebbe rappresentare un rischio in quest'ottica.

Dopo aver implementato il modello in una prima versione embrionale sono stati analizzati i coefficienti $\boldsymbol{\beta}$ in termini di significatività statistica. In particolare un coefficiente è stato considerato significativo se risultante come tale in almeno uno dei due modelli logistici: come descritto precedentemente infatti l'algoritmo prevede la formulazione di $K-1$ modelli logistici indipendenti, dove K è il numero distinto di classi della variabile risposta. In questo caso $K=3$ e la categoria di riferimento corrisponde a *Call Center*. Il livello di significatività scelto in ciascun sottomodello per ogni coefficiente è pari al 5%. Una prima analisi sulla significatività dei coefficienti ha condotto all'esclusione dal modello dei regressori *Privacy* e *Premio medio*: la prima appartiene al gruppo di quattro variabili numeriche che quantificano il tempo di completamento in minuti del corrispondente questionario online, la seconda invece misura il premio medio delle polizze sottoscritte dal cliente a livello storico. Per il test condotto sui coefficienti viene ipotizzato che ciascuno di essi sia pari a zero, e sotto l'ipotesi nulla la statistica Z di Wald, calcolata come il rapporto tra il coefficiente stimato e il suo errore standard, è distribuita secondo una normale standard [6]. In questo modo segue il calcolo dei p-value:

$$Z_{kj} = \frac{\hat{\beta_{kj}}}{\hat{\mathrm{SE}}(\hat{\beta_{kj}})}, \quad k = 1, ..., K, \quad j = 1, ..., d, \tag{1.19}$$

$$\text{p-value}(\beta_{kj}) = 2\left(1 - \Phi\left(|Z_{kj}|\right)\right), \quad k = 1, ..., K, \quad j = 1, ..., d. \tag{1.20}$$

Le variabili che sono significative nel modello multinomiale logistico progettato sono descritte nella seguente tabella 1.2.

Inoltre la variabile *Registrato* non incide sull'intercetta del modello ma il fattore di interazione agisce solo sui restanti 6 regressori numerici. Supponendo dunque che le prime 6 variabili siano categoriche escludendo l'output e *Registrato*, la formulazione del modello è la seguente:

$$\text{logit}\, P(Y = k|X_1, ..., X_d) = \beta_{k0} + \beta_{k1}^{J_1} + \beta_{k2}^{J_2} + ... + \beta_{k6}^{J_6} + \beta_{k7}^{mr} X_7 + ... + \beta_{k12}^{mr} X_{12}, \tag{1.21}$$

$$k = 1, 2, \quad j = 1, ..., 6, \quad J_j = 1, ..., |J_j|, \quad m = \textit{Cellulare, Desktop}, \quad r = \textit{Si, No}.$$

β_{k0} è l'intercetta generale del modello e per ciascuno dei 6 regressori categorici viene aggiunta una costante $\beta_{kj}^{J_j}$ corrispondente alla j-esima variabile categorica appartente alla

Nome variabile	Tipo	Valori
Allocazione	Categorica	{Non gestito, Call Center, Sales Interno}
Dominio mail	Categorica	{Aziendale, Personale, Altro}
Canale	Categorica	Canale di acquisizione del sito Lokky
Registrato	Categorica	{sì, no}
Stato utente	Categorica	Stato di avanzamento acquisizione della polizza
Prodotto	Categorica	Prodotto assicurativo selezionato
Settore	Categorica	Settore lavorativo di apaprteneza
Dispositivo	Categorica	{Mobile, Desktop}
Recency	Numerica	$[0, +\infty)$
Tempi completamento questionari (3)	Numerica	$[0, +\infty)$
Preventivi	Numerica	$\{0, 1, 2, 3, ...\}$
Polizze	Numerica	$\{0, 1, 2, 3, ...\}$

Tabella 1.2: Variabili del modello con interazioni

classe J_j. Per quanto riguarda le variabili numeriche indicizzate da 7 a 12 il coefficente β_{kj}^{mr} rappresenta il coefficiente angolare della j-esima variabile relative alle osservazioni appartenti ai gruppi indicizzati da m e r. Infine l'indice k rappresenta l'appartenenza dell'osservazione alla classe k-esima della variabile *Allocazione*. La categoria di riferimento è *Call Center*.

1.3.2. Modello pesato

Per migliorare l'adattamento ai dati del modello, è stato pensato di assegnare un rango di importanza a ciascuna osservazione, in modo tale da distinguere quelle maggiormente significative da quelle meno rilevanti, e controllarne l'impatto sul modello. In particolare la variabile *recency*, che rappresenta il tempo trascorso da quando l'utente è entrato sul sito dell'azienda al momento in cui è stato implementato il modello, ha suggerito questa variante implementativa in quanto si suppone che un'osservazione sia tanto più importante quanto più recente, mentre l'informazione contenuta in ciascun campione statistico svanisca con il passare del tempo. Questa considerazione non è vera in generale perchè non tutti i problemi di classificazione sono dipendenti dall'istante temporale, tuttavia nel contesto specifico in cui è stata svolta questa analisi l'affermazione precedente è verosi-

mile principalmentre per due motivi: in primis a causa della giovane età dell'azienda la struttura dei dati e le relazioni tra i dataset aziendali sono soggetti a cambiamenti anche sostanziali nell'arco di un anno, rendendo a volte incompatibili le informazioni raccolte tra i diversi clienti in archi temporali troppo ampi, inoltre i prodotti assicurativi proposti sono mutevoli e in rapido aumento, avendo chiaramente un impatto sul bacino di clienti che l'azienda è in grado di attrarre.

Al fine di affinare dal punto di vista modellistico questa lacuna è stato pensato di utilizzare la *recency* di ciascuna osservazione non come regressore ma come peso. I valori che tale variabile assume sono compresi tra 0 e 500, quindi sono stati presi in esame gli ultimi 500 giorni, e ad ogni osservazione è assegnato un peso esponenziale decrescente nei giorni trascorsi. La funzione assegnata a tale scopo è la seguente, dove t ha come unità di misura i giorni:

$$\text{weight}_i = e^{-\alpha t_i}, \quad i = 1, ..., n. \tag{1.22}$$

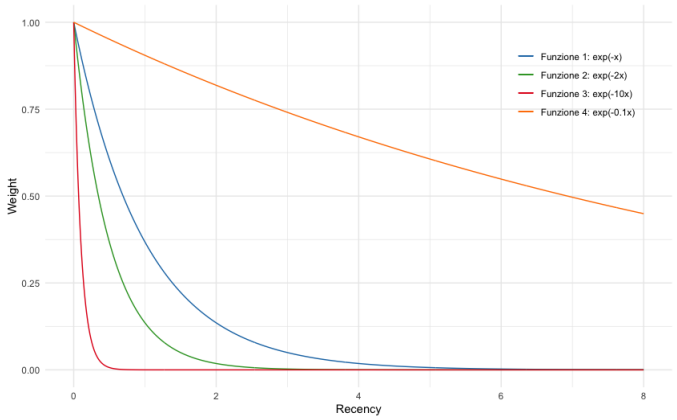

Figura 1.1: Pesi esponenziali

Ottimizzazione di α

La scelta del parametro α deve essere ponderata, nel tentativo da un lato di massimizzare l'accuratezza predittiva del modello, dall'altro di assegnare un peso a ciascuna osservazione che sia ragionevole, senza diluire eccessivamente il carico con l'aumentare della *recency*. A tale scopo sono stati implementati una serie di modelli multinomiali logistici pesati, lineari nei regressori come nel primo caso in analisi, facendo variare α tra 0 e 0.1: il primo caso corrisponde a una funzione costante pari a 1, che implica l'assegnazione dello stesso peso a ciascuna osservazione, il che è equivalente ad un modello lineare non pesato, il secondo caso invece assegna un'importanza decrescente fino al valore di e^{-500} che corrisponde circa a $e-22$. Il dataset usato nel fitting dei modelli è sempre lo stesso, diviso in training (70%)

e test set (30%). Di ciascun modello sono state valutate le principali metriche predittive come l'accuratezza, la sensitivitità e la specificità di ciascuna classe con i relativi risultati riportati in seguito, calcolate su entrambi i dataset.

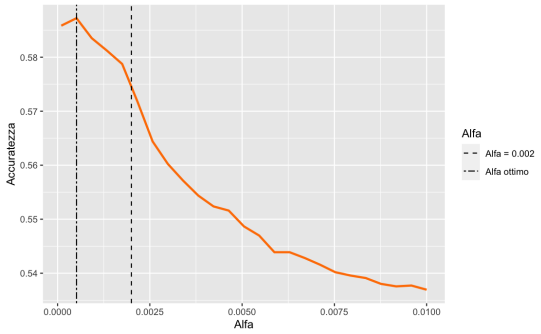

Figura 1.2: Accuratezza rispetto ad α, Test set

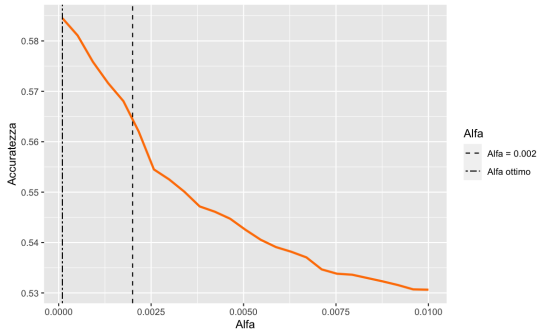

Figura 1.3: Accuratezza rispetto ad α, Training set

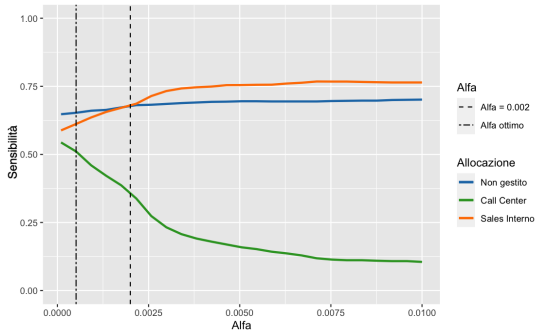

Figura 1.4: Sensibilità rispetto ad α, Test set

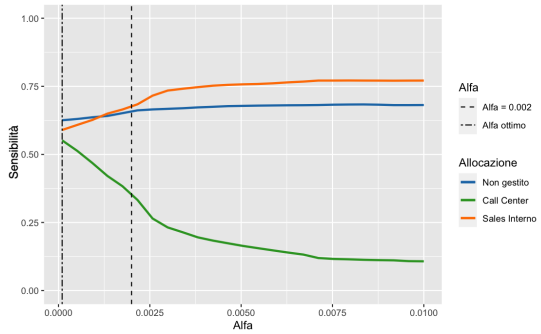

Figura 1.5: Sensibilità rispetto ad α, Training set

Il primo valore di α contrassegnato con una linea tratteggiata verticale corrisponde al massimo della accuratezza, raggiunto nel test set per $\alpha = 0.0005125$, in corrispondenza del quale anche i grafici relativi a ciascuna classe (REF) garantiscono delle ottime performance. Tale valore nell'assegnazione dei pesi corrisponde per $t = 500$ a circa 0.77, il che significa che l'interazione di un utente dopo 500 giorni incide nel modello per il 77% rispetto alla rilevanza assunta da un'osservazione attuale. Considerando la rapida evoluzione nella struttura dei dati aziendali, è stato preso in analisi anche un secondo valore pari a 0.002, corrispondente a un peso di circa il 37% per le osservazioni più datate rispetto a quelle più recenti, rispecchiando maggiormente il contesto di lavoro. Dal punto di vista predittivo la scelta di $\alpha = 0.002$ risulta adeguata come si può dedurre dai grafici sul test set, anche se non corrisponde con il valore ottimale.

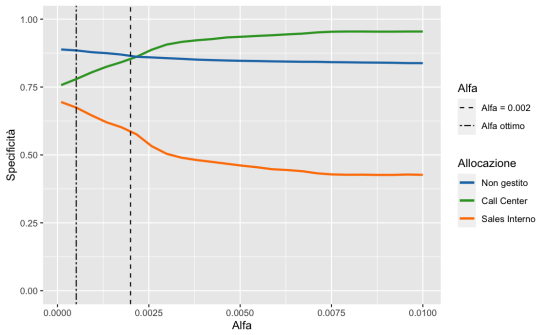

Figura 1.6: Specificità rispetto ad α, Test set

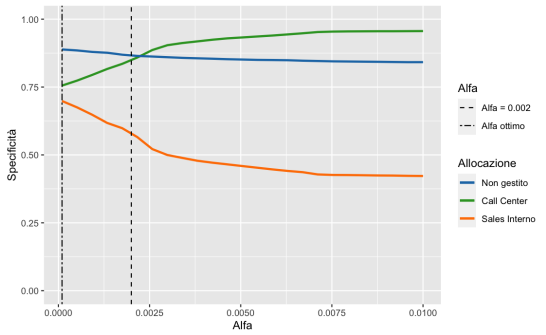

Figura 1.7: Specificità rispetto ad α, Training set

Figura 1.8: Risultati delle misurazioni

Inoltre è possibile notare un comportamento simile tra training set e test set per tutte le metriche considerate, il che scongiura la possibilità del fenomeno di overfitting per quanto riguarda l'adattamento ai dati.

Nell'esposizione dei risultati, analizzati nella sezione successiva, saranno valutati i modelli pesati relativi ad entrambi i valori di α di cui si è appena discusso.

I modelli sono stati fittati utilizzando il software RStudio attraverso la funzione *mlogit* dell'omonimo pacchetto.

1.4. Risultati

In questa sezione verranno presentati e analizzati i risultati dei modelli discussi in precedenza, con una particolare attenzione alla capacità di adattamento ai dati e alle performance predittive. Il dataset utilizzato per l'analisi in questione è comporto dalle 16 variabili descritte in tabella 1.1 e da 21610 osservazioni indipendenti, ciascuna rappresentata da un utente. Ogni modello è stato allenato su una frazione del dataset di 15127 osservazioni (circa il 70%) scelte in modo casuale, e le rimanenti sono state utilizzate come test set.

1.4.1. Bontà di adattamento ai dati

Nel presente sottoparagrafo vengono presentati i risultati dell'analisi di goodness of fit, che mira a valutare le capacità di adattamento ai dati dei quattro modelli multinomiali oggetto di studio, elemento cruciale nella scelta del modello. Per questa valutazione, sono state utilizzate tre metriche significative: la devianza, lo pseudo R^2 di Cohen e l'informazione

di Akaike. La devianza e lo pseudo R^2 di Cohen forniscono entrambi una misura oggettiva della qualità di adattamento del modello ai dati osservati: il primo in termini di discostamento dal modello ideale, interpolante tutte le osservazioni, il secondo invece colloca il modello nell'intervallo $[0, 1]$ i cui estremi rappresentano rispettivamente il modello nullo e il modello ideale. L'informazione di Akaike, invece, fornisce una misura della bontà del modello considerandone anche la complessità. Infatti si consideri che la devianza e pseudo R^2 sono misure assolute della qualità di adattamento ai dati rispetto ai regressori utilizzati, che non vengono considerati in queste metriche. Dunque se si considerano due modelli M_1 e M_2 annidati con $k_1 < k_2$ regressori, risulterà $\text{Devianza}_{M_1} > \text{Devianza}_{M_2}, R^2_{M_1} < R^2_{M_2}$. L'informazione di Akaike invece integra un elemento che penalizza un numero elevato di covariate, perciò vengono preferiti i modelli che trovano un compromesso tra goodness of fit e covariate, riducendo la probabilità di overfitting.

I principali risultati sono sintetizzati in tabella 1.3 e mostrano che il modello pesato con $\alpha = 0.002$ sia il migliore per devianza e AIC, ma presenta un valore di pseudo R^2 leggermente inferiore rispetto al modello lineare e al modello con le interazioni tra i regressori, che risulta il più performante rispetto a questa metrica.

Goodness of fit	Modello non pesato		Modello pesato	
Metrica	Senza interazioni	Con interazioni	α ottimale	$\alpha = 0.002$
Devianza	24358	24060	23269	16736
Pseudo R^2	0.24	0.25	0.20	0.22
AIC	24482	24204	22855	16876

Tabella 1.3: Goodness of fit

Questi risultati sono stati integrati con quelli relativi alle capacità predittiva dei modelli, che sono stati analizzati nel paragrafo successivo.

1.4.2. Predizione

Nella sezione presente, vengono presentati e confrontati i risultati ottenuti dai quattro modelli multinomiali differenti utilizzati nella predizione di nuove osservazioni, ovvero nuovi potenziali clienti interessati a sottoscrivere una polizza online. L'obiettivo di questo confronto è di analizzare le prestazioni dei diversi modelli e valutare quale di essi risulti più efficace nella previsione dei dati oggetto di studio. A tal fine, sono state considerate

diverse metriche di valutazione: l'accuratezza, la sensitività, la specificità, la precisione e il valore predittivo negativo, che forniscono un'indicazione sulle performance dei modelli. Attraverso questa analisi dettagliata, sarà possibile trarre conclusioni significative sulle prestazioni dei modelli multinomiali presi in considerazione, scegliendo il più performante.

In seguito sono riportati i risultati di ciascun modello rispetto alle metriche elencate in precedenza calcolate sul test set, formato da 6483 osservazioni.

I modelli non pesati mostrano una migliore capacità predittiva in generale rispetto ai modelli con i pesi, raggiungendo un'accuratezza superiore. Ciò è dovuto ad una classificazione più accurata delle classi *Sales Interno* e *Call Center* come dimostrato dalla sensitività delle rispettive classi, metriche in cui i modelli pesati presentano alcune lacune (troppo bassi i valori di sensitività di entrambi i modelli per la classe *Call Center*). Per quanto riguarda la classe *Non gestito* invece i modelli pesati sono in grado di classificare in modo corretto una frazione maggiore di utenti, come dimostrato dalla sensitività, registrando però una precisione per i *Non gestiti* di più di 10 punti percentuali inferiori rispetto ai modelli non pesati: ciò significa che i modelli pesati tendono a classificare più utenti come *Non gestito* rispetto a quelli pesati, e per questo motivo ne registrano una sensitività migliore.

Prediction		Modello non pesato		Modello pesato	
Metrica		Senza interazioni	Con interazioni	α ottimale	$\alpha = 0.002$
Accuratezza		0.6469	0.6534	0.5872	0.5741
Sensitività	Sales Interno	0.6986	0.7054	0.6117	0.6792
	Call Center	0.5884	0.5951	0.5107	0.3570
	Non gestito	0.6210	0.6263	0.6528	0.6778
Specificità	Sales Interno	0.6730	0.6794	0.6744	0.5879
	Call Center	0.8114	0.8178	0.7794	0.8521
	Non gestito	0.9337	0.9322	0.8849	0.8644
Precisione	Sales Interno	0.6552	0.6619	0.6257	0.5945
	Call Center	0.6007	0.6116	0.5274	0.5379
	Non gestito	0.7059	0.7029	0.5923	0.5614
Valore predittivo negativo	Sales Interno	0.7151	0.7217	0.6613	0.6732
	Call Center	0.8035	0.8073	0.7676	0.7332
	Non gestito	0.9058	0.9069	0.9087	0.9128

Tabella 1.4: Prediction

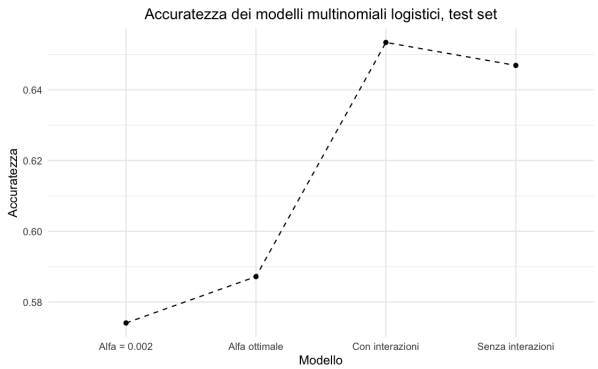

Figura 1.9: Accuratezza dei modelli

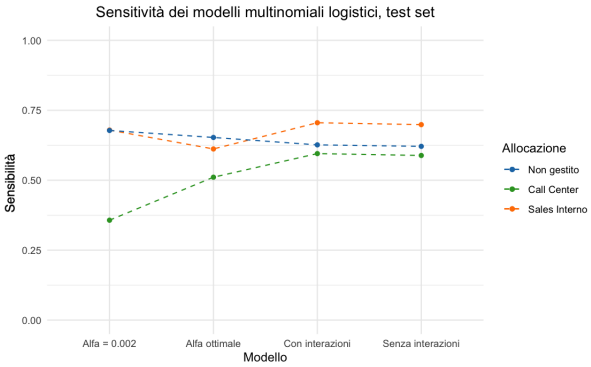

Figura 1.10: Sensibilità dei modelli

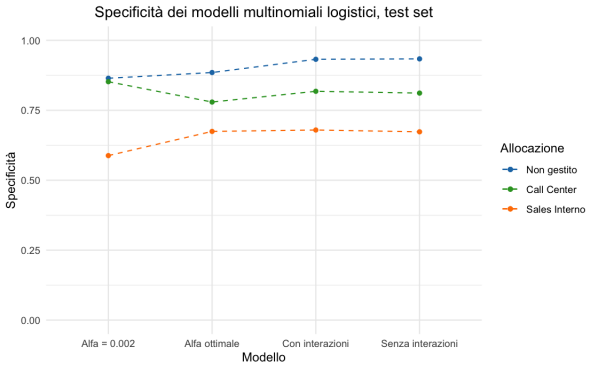

Figura 1.11: Specificità dei modelli

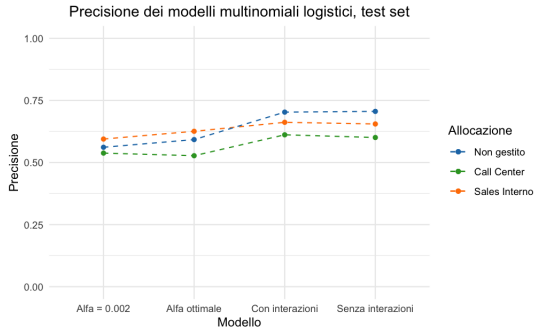

Figura 1.12: Precisione dei modelli

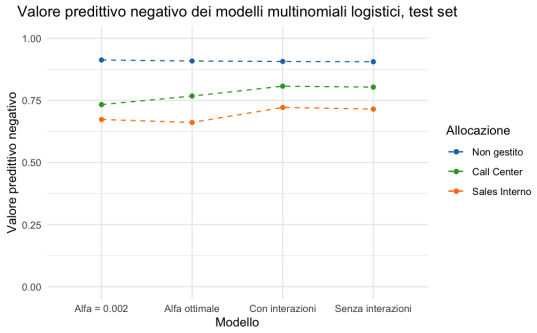

Figura 1.13: Valore predittivo negativo dei modelli

1.4.3. Modello scelto

Alla luce delle considerazioni sulle capacità di adattamento ai dati e sui risultati predittivi, il modello non pesato con le interazioni sembra essere quello che maggiormente cattura le relazioni tra regressori e output, massimizzando l'accuratezza totale e raggiungendo ottimi valori predittivi su tutte e 3 le categorie, rispetto a tutte le metriche. Inoltre raggiunge il massimo di pseudo R^2 dimostrando un buon fitting.

Coefficienti stimati del modello con le interazioni

I coefficienti stimati $\hat{\boldsymbol{\beta}}$ sono riportati nelle seguenti tabelle, suddivisi per motivi espositivi tra variabili categoriche e numeriche. Le prime sono riportate in tabella 1.5, e sono costanti additive nella formulazione (1.21), le seconde invece sono riassunte in tabella 1.6 con la distinzione rispetto alle categorie *Registrato* e *Dispositivo*.

È possibile notare che la categoria con l'effetto maggiore, e quindi più importante per il modello, corrisponde al *Canale* di acquisizione *Diretto* del cliente, con un rapporto tra la

probabilità di appartenere alla classe *Non Gestito* e *Call Center* di $e^{2.4} \approx 11$ rispetto al canale di acquisizione *Campagne*. Anche la classe *Professionisti* della variabile *Prodotto* risulta essere fortemente determinante, con un rapporto con il gruppo *Altro* pari rispettivamente a $e^{-2.2} \approx 0.11$ e $e^{-2.4} \approx 0.09$ per gli output *Non Gestito* e *Sales Interno* rispetto alla classe *Call Center*. Per quanto riguarda le variabili numeriche invece è interessante notare che i coefficienti relativi alla variabile *recency*, che quantifica il numero di giorni trascorsi dall'interazione del cliente con il sito, siano sempre negativi: ciò significa che con il trascorrere dei giorni è preferibile allocare il cliente al *Call Center* esterno. In particolare per ogni giorno trascorso, considerando costanti gli altri regressori, l'incremento relativo del rapporto tra la probabilità che l'utente sia *Non Gestito* e *Call Center*, nel caso in cui l'utente sia *Registrato* e abbia navigato da *Desktop*, è pari a $e^{-0.0067} - 1 \approx -0.007$ che equivale quindi a un decremento dello 0.7%.

Infine l'intercetta è da interpretare come segue: nel caso in cui il cliente appartenga alle classi di riferimento delle variabili categoriche (*Mail* = *Altro*, *Canale* = *Campagne*, *Stato* = *Cliente*, *Prodotto* = *Altro*, *Settore* = *Altri Professionisti* e *Dispositivo* = *Desktop*) e le variabili numeriche siano nulle, il rapport tra le probabilità di appartenere rispettivamente a *Non Gestito* e *Call Center* è pari a $e^{6.9} \approx 10^3$, mentre lo stesso rapporto tra la classe *Sales Interno* e *Call Center* è pari a $e^{1.8} \approx 6$.

Tuttavia il modello scelto presenta delle lacune nella classificazione degli utenti da allocare al *Call Center* e *Non gestiti*. Soprattutto questi ultimi, categorizzati in maniera corretta nel 63% dei casi, rappresentano un'opportunità commerciale per l'azienda: aumentare la capacità del classificatore di modellare in maniera corretta i clienti che non devono essere affiancati da un operatore telefonico, che sono a tutti gli effetti dei consulenti esperti dei prodotti di Lokky, consente di ridurre i costi di gestione dei clienti e di allocare gli addetti alla vendita ai prospect che maggiormente necessitano di una spinta verso la sottoscrizione della polizza, aumentando anche il tasso di conversione. Questo è l'obiettivo primario al fine di migliorare il modello multinomiale implementato. A tale scopo è necessario ampliare il database di training del modello riguardante questa categoria, attraverso la generazione di dati sintetici dalla distribuzione dei dati reali, volto ad estendere le conoscenze del modello sui clienti. In seguito a questa operazione è previsto un aumento nella capacità allocativa dei clienti *Non gestiti*, probabilmente a discapito delle altre classi. Tuttavia questa operazione si inserisce in modo coerente in un progetto più ampio, ovvero la transizione del lead scoring da un modello parametrico a un modello di machine learning, nel quale il database aziendale dovrà necessariamente essere arricchito. Questo cambiamento strutturale nell'algoritmo si rifletterà inanzitutto in un ulteriore aumento dell'accuratezza generale del modello, e in secondo luogo metterà a disposizione

Coefficienti variabili categoriche		Gestione Lead	
Variabile	Classe	Non Gestito	Sales Interno
Intercetta	-	6.9	1.8
Mail	Aziendale	-1.3	1
	Personale	-1.9	1
	Altro	0	0
Canale	Diretto	2.4	0.8
	Facile	0	0.1
	Google	0.5	0.4
	Altri Partner	1	0.9
	Campagne	0	0
Stato	Lightbox	-1	-0.4
	Ulteriori dati	-1	0
	Proposta	-1.3	-0.2
	Dati contraente	-0.7	-0.6
	Preventivo	-1.2	-0.7
	Cliente	0	0
Prodotto	Professionisti	-2.2	-2.4
	Artigiani	-1.3	-0.2
	Negozi	-1.3	-0.2
	Altro	0	0
Settore	Artigiani	-1.7	-1.5
	Ristorazione	-1.8	-1.1
	Professionisti	-0.6	1.1
	Installatori	-1.2	-0.8
	Altro	0.3	-0.4
	Altri Professionisti	0	0
Dispositivo	Cellulare	-0.4	-0.3
	Desktop	0	0

Tabella 1.5: Variabili categoriche

Coefficienti variabili numeriche		Gestione Lead	
Variabile	Classe	Non Gestito	Sales Interno
Recency	Registrato - Desktop	-0.0067	-0.0019
	Registrato - Mobile	-0,0058	-0.0008
	Non registrato - Desktop	-0.0089	-0.0051
	Non registrato - Mobile	-0.008	-0,004
Ulteriori dati	Desktop	-0.0916	-0.0103
	Mobile	0.0577	-0.0124
Proposta	Registrato	0.1198	0.1035
	Non registrato	-0.0643	-0.0025
Dati contraente	Registrato	0.0234	0.0368
	Non registrato	0.4433	0.3
Preventivi	Registrato - Desktop	-1.3541	-0.3686
	Registrato - Mobile	-1.6614	-0.3789
	Non registrato - Desktop	-0.7521	0.0945
	Non registrato - Mobile	-1.0594	0,0842
Polizze	Registrato	0.796	0.5678
	Non registrato	-0.2183	0.0389

Tabella 1.6: Variabili numeriche

dell'azienda uno strumento innovativo e performante incentrato sul cliente, integrato con il sistema di preventivazione online e utilizzabile in tempo reale, coerentemente con la politica aziendale di digitalizzazione e automatizzazione dei processi.

2 | Generazione dati sintetici

Dopo aver sviluppato un modello di regressione logistica multinomiale per lo scoring degli utenti del sito, l'obiettivo di questa sezione è una transizione verso i più moderni metodi di classificazione non parametrici che fanno uso di tecniche di supervised learning per il labelling delle osservazioni. Tale sviluppo è fondamentale se si vuole migliorare le performance predittive, ma garantisce anche una maggiore modernità all'immagine stessa del lead scoring.

L'obiettivo è di usare i dati sintetici per migliorare la classificazione dei lead, confrontando la performance predittiva del modello multinomiale descritto al capitolo 1 nel caso si usino solo i dati reali con l'utilizzo dei dati sintetici per integrare il patrimonio informativo. In una futura evoluzione del progetto, nel quale il classificatore sarà disegnato facendo uso di tecniche di machine learning, l'arricchimento del database aziendale attraverso la generazione di dati sintetici avrà verosimilmente un impatto ancora più significativo nelle capacità predittive del modello.

L'algoritmo per la generazione di dati sintetici si chiama *PrivBayes* ed è contenuto nella libreria python *DataSynthesizer* che consiste di due step fondamentali: il primo è la stima della distribuzione congiunta dei dati reali attraverso le distribuzioni condizionate, facendo uso di una rete bayesiana per modellare il legame di dipendenza tra le variabili del dataset. La seconda è la generazione dei dati sintetici facendo un sampling dalla distribuzione stimata.

I dati sintetici prodotti dovranno in primis replicare in maniera fedele la distribuzione congiunta delle variabili, aumentando la numerosità delle osservazioni, ma dovranno esplorare in modo più continuo lo spazio delle caratteristiche senza però violare le distribuzioni marginali dei dati reali. Tale compromesso permette da un lato di evitare di replicare in modo eccessivamente rigoroso il dataset per poter ottenere un significativo aumento dell'informazione a nostra disposizione, dall'altro di riprodurre in maniera sufficientemente accurata le distribuzioni delle variabili per ottenere dati verosimili. Nella generazione dei dati sarà possibile regolare arbitrariamente al variare del parametro ϵ la distanza tra le distribuzioni marginali reali e quelle da cui verrà effettuato il campionamento.

In seguito saranno fornite nozioni teoriche sulle reti Bayesiane, utilizzate nell'algoritmo PrivBayes per stimare la struttura di dipendenza tra le vadiabili del dataset D. Successivamente verrà dettagliato l'algoritmo di generazione dei dati sintetici.

2.1. Reti Bayesiane

2.1.1. Reti Bayesiane come grafi direzionati

Le reti Bayesiane sono una tipologia di modello grafico utilizzato per rappresentare le relazioni di dipendenza probabilistica tra le variabili aleatorie. In particolare, sono composte da nodi e da archi, che rappresentano rispettivamente le variabili casuali e i loro legami di dipendenza. Tali dipendenze condizionali nel grafo sono stimate facendo uso di teorie matematiche e metodi computazionali avanzati, combinando principi di teoria dei grafi, teoria di probabilità e statistica Bayesiana. Iniziamo a fornire alcune definizioni utili nella trattazione dell'argomento. Le definizioni utilizzate nella sezione 2.1, se non diversamente riportato, fanno riferimento al testo Learning Bayesian Networks (Neapolitan, Richard)[7].

Definizione 2.1. *Un grafo G è una coppia (V, E), in cui V è un insieme finito e non vuoto detto insieme dei vertici di G (ed i suoi elementi sono detti vertici o nodi) e E, chiamato insieme degli archi, è un insieme di coppie di vertici di V. Se una coppia (u, v) è ordinata con u che precede v l'arco si dice orientato, altrimenti (u, v) indicherà lo stesso arco di (v, u) e sarà definito non orientato. Un grafo in cui tutti gli archi siano orientati sarà orientato, uno in cui tutti gli archi siano non orientati sarà detto non orientato. Un grafo in cui sono presenti archi orientati e non orientati è detto misto ([5]).*

I modelli a grafo dunque possono essere suddivisi in due tipologie: orientati e non orientati, a seconda della relazione di dipendenza che intercorre tra le variabili. I primi sono utili quando tra una coppia di variabili aleatorie X_i e X_j esiste una dipendenza asimmetrica, nella quale ad esempio X_i dipende direttamente da X_j. I secondi invece permettono di rappresentare strutture di dipendenza simmetrica tra variabili casuali. Definiamo inoltre il concetto di cammino e le relazioni di parentela tra i nodi, utili nella costruzione della rete Bayesiana.

Definizione 2.2. *L'insieme degli archi che collegano k nodi da X_1 a X_k, $k \geq 2$, è chiamato cammino. Se esiste un cammino p da u a u′ diremo che u′ è raggiungibile da u. Un ciclo è un cammino che abbia almeno un arco in cui $u_0 = u_k$. Esso si dirà semplice se tutti i vertici ad eccezione del primo e dell'ultimo sono distinti. Un cammino diretto è*

un cammino in cui tutti gli archi sono diretti e sono percorsi lungo la loro direzione. In maniera analoga si definisce un ciclo diretto. Un grafo senza cicli è aciclico. Un DAG è un grafo direzionato aciclico, ovvero un grafo aciclico in cui l'insieme degli archi E è formato esclusivamente da archi direzionati ([5]).

Definizione 2.3. *Un grafo G è connesso se, per ogni coppia di vertici, esiste un cammino tra di essi. Un grafo si dice fortemente connesso se ogni vertice è raggiungibile da tutti gli altri nodi ([5]).*

Definizione 2.4. *Dato un DAG $G = (V, E)$ e due vertici u e $v \in E$ diremo che ([5]):*

- *u è genitore o padre di v se esiste in E un arco da u a v. allo stesso modo v è figlio di u*
- *u è detto antenato di v e, viceversa, v è discendente di u, se esiste un cammino che da u va in v. In caso contrario, ci si riferisce a v come ad un nodo non discendente. Si noti che, poiché nella definizione di cammino abbiamo imposto $k \geq 2$, u non è discendente di se stesso. Inoltre, se u è un nodo, gli antenati di u sono anche non discendenti di u.*

In particolare una rete Bayesiana è un grafo orientato aciclico la cui struttura è definita da due insiemi: l'insieme dei nodi e l'insieme degli archi orientati. I nodi rappresentano le variabili casuali $X_1,\ldots,X_d$, gli archi orientati rappresentano una dipendenza statistica diretta tra esse. Per ogni coppia di variabili X e Y possono esistere 3 tipologie di dipendenza tra esse [12], elencate in seguito con il sostegno dell'esempio 2.1, DAG che modella la probabilità di gol relativa a ciascun tiro effettuato nel calcio:

- *Dipendenza diretta:* esiste un arco direzionato da Y a X. Questo significa che la distribuzione marginale di Y dipende dai valori che X assume. Segue che X è un genitore di Y, e Y figlio di X. Ad esempio in Figura 2.1 la distribuzione di *xG* (Expected Goals) dipende direttamente dalla *Posizione di tiro* e dalla *Velocità del tiro*.
- *Indipendenza condizionale debole:* esiste un percorso che parte da X e arriva in Y ma non esiste una dipendenza diretta tra le due variabili. In altre parole X appartiene all'insieme degli antenati di Y. In questo caso X e Y sono indipendenti condizionatamente all'insieme degli ascendenti di Y. Ad esempio in Figura 2.1 *xG* e *Ruolo* sono indipendenti condizionatamente alla *Posizione di tiro* del calciatore e alla *Velocità del tiro*. Intuitivamente significa che la probabilità che un tiro vada a segno dipende dal ruolo del calciatore che ha tirato, ma una volta noto da dove

è stato effettuato il tiro e con che velocità, sapere il ruolo del calciatore che ha effettuato il tiro non aumenta le informazioni riguardanti la probabilità di segnare. Allo stesso modo invece *xG* e *Calciatore* sono indipendenti condizionatamente a *Posizione di tiro* e *Velocità del tiro* congiuntamente.

- *Indipendenza condizionale forte:* non esiste alcun percorso tra X e Y, quindi X e Y sono indipendenti condizionatamente sia all'insieme degli antenati sia di X sia di Y.

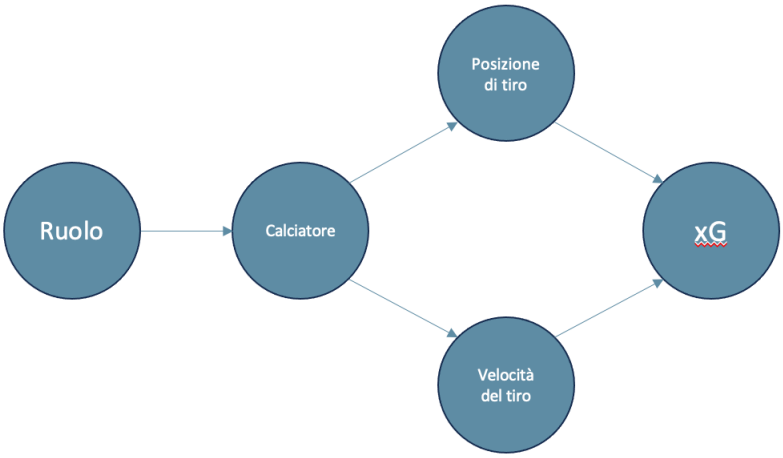

Figura 2.1: Esempio di rete Bayesiana

Il concetto di indipendenza condizionale forte è usato per ridurre, talvolta in maniera significativa, il numero di parametri necessari per caratterizzare la distribuzione di probabilità congiunta delle variabili, e consente di calcolare in maniera efficiente la distribuzione a posteriori una volta note le marginal likelihood [2]. Il concetto di aciclicità invece consiste nell'ipotizzare che ogni nodo del grafo non può essere suo stesso ascendente o discendente. Tale restrizione è di vitale importanza per la fattorizzazione della distribuzione di probabilità congiunta come mostreremo in seguito, permettendo alla struttura modellistica di esistere [2]. Dopo questa premessa diamo una definizione più formale di una rete Bayesiana, facendo uso della nozione di Condizione di Markov:

Definizione 2.5. *Sia $P[A]$ la distribuzione congiunta delle variabili aleatorie appartenenti ad A, sia G un DAG avente A come insieme dei nodi. Diremo che la coppia $(G, P[A])$ soddisfa la Condizione di Markov se per ogni X appartenente ad A, X è condizionatamente indipendente dall'insieme dei suoi non-discendenti dato l'insieme dei suoi genitori. In*

maniera equivalente, diremo che G e P soddisfano la Condizione di Markov l'uno rispetto all'altro. In simboli:

$$I_P(X, ND_X|PA_X).$$

Dove ND_X è l'insieme dei non discendenti di X e PA_X è l'insieme dei genitori di X. Ricordiamo che $PA_X \subset ND_X$.

Definizione 2.6. *Sia $A = \{X_1, \ldots, X_d\}$ un insieme di variabili aleatorie, in cui ogni X_i è unica in A e avente insieme di ascendenti Π_i, e sia P una distribuzione congiunta di probabilità su A. Definiamo una rete Bayesiana N su A come una coppia $N = (G, P)$, in cui G è un grafo direzionato aciclico avente A come insieme dei nodi e che soddisfa la Condizione di Markov. In particolare, P sarà una distribuzione di probabilità congiunta attraverso le distribuzioni condizionate, che formano l'insieme dei parametri della rete.*

In particolare G è un insieme di coppie variabile aleatoria – antenati (X_i, Π_i) tale che [12]:

1. $\Pi \subset A \setminus \{X_i\}, \quad \forall i = 1, ..., d.$
2. $Xj \notin \Pi_i, \quad \forall i, j : 1 \leq i < j \leq d.$ Ovvero non esiste alcun arco direzionato da X_j a X_i. In questo modo è garantita l'aciclicità di N.

L'insieme P nella definizione 2.6 è formato dai parametri $\theta_{Xi|\Pi_i} = P[x|\Pi_i]$, ovvero le distribuzioni condizionate di ciascuna variabile aleatoria X_i condizionatamente a Π_i, per ogni valore x che appartiene al supporto di X_i. Definiamo ora con k il grado della rete Bayesiana N come la massima cardinalità dell' insieme Π_i dei parenti di X_i su N per ogni i, nonché il massimo numero di variabili a cui X_i viene condizionata. La distribuzione congiunta di A, $P[A] = P[X_1, \ldots, X_n]$, ipotizzando che X_i e $X_j \notin \Pi_i$ siano indipendenti condizionatamente a Π_i, è:

$$P[A] = P[X_1]P[X_2|X_1]\ldots P[X_d|X_{d-1}, \ldots, X_1] = \prod P[X_i|X_{i-1}, \ldots, X_1]. \tag{2.1}$$

Tale distribuzione viene approssimata in N da

$$P_N[A] = \prod P[X_i|\Pi_i]. \tag{2.2}$$

Intuitivamente N descrive le relazioni di dipendenza di A se Π_i contiene tutte le variabili da cui X_i dipende direttamente. La scelta di k quindi ha una forte incidenza sulla qualità dell'approssimazione di $P[A]$ attraverso $P_N[A]$: una scelta del valore di k elevato consiste nel fatto che Π_i comprenda un consistente numero di ascendenti di X_i, ma il calcolo di

$P_N[X_i|\Pi_i]$ comporta un costo computazionale dispendioso a causa dell'elevato numero di valori distinti che Π_i può assumere. Contenuti valori di k invece permettono a $P_N[X_i|\Pi_i]$ di giacere in uno spazio di poche dimensioni, facilitando il calcolo delle distribuzioni condizionate e di conseguenza la stima della distribuzione congiunta. Un adeguato valore di k dunque è frutto di un equo bilanciamento tra il mantenimento della ricchezza informativa nelle distribuzioni condizionate e una efficienza computazionale nella stima delle stesse.

2.1.2. Relazione tra DAG e indipendenze condizionate

La Condizione di Markov fornisce un criterio per identificare l'indipendenza tra le variabili, ma non è uno strumento informativo per fare inferenza sulle effettive dipendenze in G: il fatto che in G esista un arco da X_1 a X_2 non implica che tale legame sia reale. In generale vorremmo quindi che ciascun arco rappresenti una effettiva dipendenza tra le variabili in A. In altre parole è necessario ampliare la nostra conoscenza sulle indipendenze condizionate, evidenziando anche quelle che non vengono dedotte direttamente dalla Condizione di Markov, in modo tale che G rappresenti tutte e sole le indipendenze condizionate esistenti. In questo senso iniziamo a definire il concetto di indipendenza indotta, che ci servirà per definire la d-separazione tra nodi di un grafo e introdurre in seguito la nozione di faithfulness:

Definizione 2.7. *Sia dato $G = (A, E)$ DAG. Diremo che G induce l'indipendenza condizionale $I_P(A, B|C)$ per $B, C, D \subseteq A$ se $I_P(B, D|C)$ $\forall P \in P_G$ dove con P_G indichiamo l'insieme delle distribuzioni per cui (G, P) soddisfi la Condizione di Markov.*

Si consideri ora l'esempio in figura 2.1. Per semplicità sono assegnati i seguenti nomi ai nodi della rete:

R : Ruolo.

C : Calciatore.

P : Posizione di tiro.

V : Velocità del tiro.

xG : xG.

Le uniche indipendenze condizionali che sono dedotte dalla Condizione di Markov sono $I_P(xG, C|\{P, V\})$, $I_P(xG, R|\{V, P\})$, $I_P(P, R|C)$, $I_P(P, V|C)$ e $I_P(V, R|C)$. Tuttavia non significa che non sia possibile dedurre, a partire dalla Condizione di Markov, altre indipendenze condizionali, come ad esempio $I(xG, R|C)$. Infatti ipotizzando che R asssuma il valore di *Attaccante*, ciò restringe la variabile C esclusivamente agli attaccanti. Ciò

implicherà ragionevolmente che la posizione da cui viene effettuato il tiro sia molto avanzata e la velocità di tiro superiore alla media, in quanto gli attaccanti hanno una ottima tecnica di tiro, e il valore di Expected Goals sia elevato. Perciò è inverosimile che R e xG siano indipendenti. Per lo stesso motivo è da escludere che C e xG lo siano. Tuttavia una volta noto C (il calciatore che effettua il tiro), la distribuzione di xG non è influenzata dal valore assunto da R. Infatti R influenza il valore di xG in quanto agisce su C, ma una volta noto C il valore assunto da R diventa irrilevante nella stima di xG. Infatti è possibile dimostrare che dalla Condizione di Markov discende $I_P(xG, R|C)$:

$$P(xg|c,r) = \sum_{p,v} P(xg|p,v,c,r)P(p,v|c,r) = \sum_{p,v} P(xg|p,v,c)P(p,v|c) = P(xg|c).$$

Nella seconda uguaglianza è stata applicata la Condizione di Markov. L'esempio mostra come sia possibile dedurre altri legami di indipendenza condizionata dalla Condizione di Markov, che ci permettono di introdurre la nozione di d-separazione. Tale idea è fondamentale perché fornisce una relazione tra il concetto di indipendenza condizionata e la "separazione" nel DAG delle variabili, codificando graficamente nel DAG una relazione puramente probabilistica. In particolare mostreremo che dalla Condizione di Markov si deducono i seguenti risultati:

1. Tutte le d-separazioni sono indipendenze condizionate;
2. Ogni indipendenza condizionata indotta è rappresentata da una d-separazione.

Ciò implica intuitivamente che se X_1 e X_2 sono indipendenti condizionatamente a X_3, debbano essere graficamente "separate" da X_3 in G. Infatti dimostreremo che in un DAG due variabili sono adiacenti se e solo se non esiste un sottoinsieme dei nodi che le d-separi. Prima di definire la nozione di d-separazione definiamo i concetti di cammini attivi e bloccati, concetti illustrati in Neapolitan, Richard [7]:

Definizione 2.8. *Una Catena tra X_1 e X_k è l'insieme di archi (non direnzionari) che formano il cammino tra X_1 e X_k. Lo indicheremo sia come $[X_1, X_2, \ldots, X_k]$ sia come $[X_k, X_{K-1}, \ldots, X_1]$. Dato l'arco $X_1 \to X_2$, X_1 è chiamata testa, X_2 coda. Inoltre:*

- *La catena $X \to Y \to Z$ è un incontro testa-coda, gli archi si incontrano testa-coda in Y e Y è detto nodo testa-coda;*
- *la catena $X \leftarrow Y \to Z$ è un incontro coda-coda, gli archi si incontrano coda-coda in Y e Y è detto nodo coda-coda. La catena $X \to Y \leftarrow Z$ è un incontro testa-testa, gli archi si incontrano testa-testa in Y e Y è detto nodo testa-testa.*

Definizione 2.9. *Sia $G = (A, E)$ un DAG, $B \subseteq A$, X e Y nodi distinti in $A \setminus B$ e l una catena tra X e Y. l è una catena bloccata da B se una delle seguenti condizioni è verificata:*

- *$Z \in B$ sulla catena l tale che gli archi incidenti a Z su l siano archi testa-coda;*
- *$Z \in B$ sulla catena l tale che gli archi incidenti a Z su l siano archi coda-coda;*
- *$Z \in l$ tale che Z e tutti i suoi discendenti non sono in B, e gli archi incidenti su Z in l si incontrano testa-testa in Z.*

Se nessuna di queste condizioni si verifica, allora la catena è attiva dato A.

Definizione 2.10. *Sia $G = (A, E)$ un DAG, $B \subseteq A$ e X e Y due nodi distinti in $A \setminus B$. Diciamo che X e Y sono d- separati da B in G se ogni catena tra X e Y è bloccata da A [7].*

Definizione 2.11. *Sia $G = (V, E)$ un DAG e siano A, B e C sottoinsiemi mutuamente disgiunti di V. Diciamo che A e B sono d-separati da C in G se per ogni X in A, Y in B, X e Y sono d-separati da C [7]. Scriviamo:*

$$I_G(A, B|C).$$

Lemma 2.1. *Sia P una distribuzione di probabilità su A insieme di variabili aleatorie, sia $G = (A, E)$ un DAG. Allora (G, P) soddisfa la Condizione di Markov se e solo se per ogni insieme A, B, e $C \subseteq V$ reciprocamente disgiunti, se A e B sono d-separati da C allora A e B sono indipendenti condizionatamente a C rispetto a P [7]. Ovvero (G, P) soddisfano la Condizione di Markov se e solo se*

$$I_G(A, B|C) \implies I_P(A, B|C). \tag{2.3}$$

Tale lemma garantisce che in un DAG G in cui vale la Condizione di Markov, se A e B sono d-separati da C allora sono condizionatamente indipendenti. Per tale ragione diciamo che G è una mappa di indipendenza di P. Ora invece mostriamo che non esistono in G indipendenze condizionate indotte dalla Condizione di Markov che non siano identificabili da d-separazioni. Per arrivare a ciò dobbiamo prima dare la seguente definizione [7]:

Definizione 2.12. *Sia V un insieme di variabili casuali e A_1, B_1, C_1, A_2, B_2 e C_2 siano sottoinsiemi di V. Diciamo che l'indipendenza condizionale $I_P(A_1, B_1|C_1)$ è equivalente all'indipendenza condizionale $I_P(A_2, B_2|C_2)$ se per ogni distribuzione di probabilità P di V, $I_P(A_1, B_1|C_1)$ vale se e solo se vale $I_P(A_2, B_2|C_2)$.*

Lemma 2.2. *Ogni indipendenza condizionata indotta dalla condizione di Markov è equivalente a una indipendenza condizionata tra insiemi disgiunti di variabili aleatorie [7].*

Lemma 2.3. *Sia $G = (V, E)$ un DAG, sia Π l'insieme delle distribuzioni di probabilità P su V insieme di variabili aleatorie tale che (G, P) soddisfi la Condizione di Markov. Allora per ogni insieme $A, B,$ e $C \subseteq V$ reciprocamente disgiunti [7]:*

$$I_P(A, B|C)\ \forall P \in \Pi \implies I_G(A, B|C). \tag{2.4}$$

Come risultato immediato dei precedenti lemmi, un DAG G rappresenta tutte e sole le indipendenze condizionate che sono identificate da d-separazioni. È importante soffermarsi sul significato del lemma precedente: si noti che per particolari distribuzioni P, che soddisfano la Condizione di Markov in G, alcune indipendenze condizionate possono non essere identificate da d-separazioni. Tuttavia le uniche indipendenze che sono soddisfatte da tutte le distribuzioni che soddisfino la Condizione di Markov sono quelle identificate da d-separazioni.

Soffermiamoci ora sulla struttura grafica di un DAG. Molti grafi aciclici presentano la stessa architettura nel senso che hanno le stesse d-separazioni. Possiamo quindi definire una equivalenza tra DAG sulla base del concetto di d-separazione e dividere i DAGs in classi di equivalenza [7].

Definizione 2.13. *Siano $G_1 = (V, E_1)$ e $G_2 = (V, E_2)$ due DAGs. G_1 e G_2 sono Markov equivalenti se $\forall A, B,$ e $C \subseteq V$ reciprocamente disgiunti, A e B sono d-separati da C in G_1 se e solo se A e B sono d-separati da C in G_2.*

Teorema 2.1. *Due DAGs sono Markov equivalenti se e solo se inducono le stesse indipendenze condizionate dalla Condizione di Markov.*

Lemma 2.4. *Sia $G = (A, E)$ un DAG, X e Y appartenenti a A. X e Y sono adiacenti in G se e solo se non sono d-separati in G da qualche sottoinsieme di A.*

L'ultimo lemma ci permette di dedurre che sotto la Condizione di Markov X e Y sono adiacenti in G, e che quindi esiste un arco da X a Y o viceversa se non sono d-separati da nessun altro insieme di nodi. Oppure in altre parole se non esiste un insieme $C \subseteq V$ tale che $I_P(X, Y|C)\ \forall P \in \Pi$ insieme delle distribuzioni su A. Tuttavia ciò non implica che X e Y non siano condizionatamente indipendenti a C per qualche particolare distribuzione di probabilità che soddisfi comunque la Condizione di Markov. È chiaro quindi che non sia sufficiente che X e Y siano adiacenti perché esista un vero legame di dipendenza tra

esse. Abbiamo bisogno del concetto di faithfulness per stabilire se tra variabili adiacenti vi sia una dipendenza diretta [7].

Definizione 2.14. *Supponiamo di avere una distribuzione congiunta P su A e sia $G = (A, E)$ un DAG. (G, P) soddisfa la Condizione di Faithfulness se G induce, attraverso la Condizione di Markov, tutte e sole le indipendenze condizionate in P. Ovvero entrambe le seguenti condizioni devono essere verificate:*

- *(G, P) soddisfano la Condizione di Markov.*
- *Tutte le indipendenze condizionate in P sono rappresentate da G, secondo la Condizione di Markov.*

Se (G, P) soddisfano la Condizione di Faithfulness diciamo che G è una mappa perfetta di P. Forniamo ora due condizioni per verificare tale condizione in un DAG [7]:

Teorema 2.2. *Supponiamo che P sia una distribuzione congiunta su A e sia $G = (A, E)$ un DAG. (G, P) soddisfa la Condizione di Faithfulness se e solo se tutte e sole le indipendenze condizionate in P sono identificate da d-separazioni in G.*

Teorema 2.3. *Se (G, P) soddisfa la Condizione di Faithfulness, allora P soddisfa tale condizione insieme a qualunque DAG Markov equivalente a G e solamente con essi.*

Alla luce di questi due teoremi è importante chiedersi quanto sia verosimile, assegnando arbitrariamente le distribuzioni condizionate alle variabili aleatorie di un DAG, che sia soddisfatta la Condizione di Faithfulness. In questo senso ci viene in soccorso il seguente studio: se attribuiamo in modo arbitrario delle distribuzioni condizionali alle variabili di un grafo aciclico diretto, è improbabile ottenere una distribuzione congiunta che non soddisfi tale condizione con il DAG. Uno studio, noto come il Teorema di Spirtes et al. del 1993 e del 2000, ha dimostrato che nel caso dei modelli lineari, in cui ogni variabile è una funzione lineare dei suoi genitori e di un termine di errore, l'insieme delle assegnazioni delle distribuzioni tali che non sia soddisfatta la condizione di faithfulness ha misura di Lebesgue nulla rispetto allo spazio delle assegnazioni di probabilità condizionate [9]. Nel caso in cui P ammetta una mappa perfetta in G, l'obiettivo è trovare quell'unica classe di equivalenza a cui G appartiene. Tale problema è chiamato learning della struttura Bayesiana e sarà trattato nella sezione successiva. Terminiamo la sezione corrente definendo i Markov Blankets e Boundaries, che ci aiutano a capire meglio quanto un nodo in una rete Bayesiana possa risentire dell'influenza di un nodo distante [7].

Definizione 2.15. *sia A un insieme di variabili aleatorie, P distribuzione congiunta su A, $X \in A$. M_X è detto Markov blanket di X se*

$$I_P(X, A \setminus (M_X \cup X).$$

Un Markov blanket di X dunque è un insieme di variabili MX condizionatamente a cui X è indipendente da ogni altra variabile.

Definizione 2.16. *Sia A un insieme di variabili aleatorie, P distribuzione congiunta su A, $X \in A$. Un Markov boundary di X è qualunque Markov blanket di X tale che nessuno dei suoi sottoinsiemi propri sia un Markov blanket.*

Da ciò segue questo interessante teorema che ci permette di stabilire fino a che "profondità" in una rete Bayesiana, un nodo può essere influenzato dai predecessori [7].

Teorema 2.4. *Supponiamo che (G, P) soddisfi la Condizione di Faithfulness. Allora per ogni variabile aleatoria X, l'insieme dei genitori di X, i figli di X e i genitori dei figli di X sono l'unico Markov boundary di X in (G, P).*

2.1.3. Learning della struttura Bayesiana

Il processo di apprendimento della struttura del grafo, noto come "learning della struttura del grafo", riveste un ruolo cruciale nell'identificazione delle connessioni causali tra le variabili. Invece di dipendere esclusivamente da conoscenze pregresse o ipotesi sull'architettura della rete, la costruzione del grafo si basa sui dati osservati per determinare la migliore configurazione dei nodi e degli archi. In questo capitolo, sarà esplorato e approfondito il processo di apprendimento della struttura del grafo nelle reti Bayesiane. L'obiettivo principale è quello di sviluppare un metodo efficiente ed efficace per estrarre una disposizione del grafo coerente con i dati a partire da un insieme di variabili aleatorie.

Il learning della struttura della rete Bayesiana può essere visto più generalmente come un problema di model selection probabilistico, in cui l'obiettivo è trovare la struttura di rete che rappresenti i dati osservati in maniera ottimale. L'approccio probabilistico al learning della struttura del grafo si basa sull'utilizzo delle probabilità condizionate e delle regole di Bayes. Partendo da un insieme di dati osservati, il processo di apprendimento consiste nella determinazione della struttura del grafo che massimizzi la verosimiglianza dei dati stessi.

Innanzitutto in questo contesto un modello probabilistico M per un insieme di variabili aleatorie A è un insieme di distribuzioni di probabilità congiunte su A. Normalmente, ogni distribuzione di probabilità congiunta in un modello si ottiene assegnando i valori

ai membri di un insieme di parametri Θ che fanno parte del modello. Se la distribuzione di probabilità P è un membro di M, diciamo che P è inclusa in M. Se le distribuzioni di probabilità in un modello sono ottenute assegnando valori ai membri di un insieme di parametri Θ, significa che esiste un'assegnazione di valori ai parametri che produce tale distribuzione di probabilità. Un modello di rete Bayesiano N consiste in una coppia $N = (G, \Theta)$ dove $G = (A, E)$ è un DAG, A è un insieme di variabili casuali e E è un insieme di coppie orientate di elementi di A, Θ è un insieme di parametri i cui membri determinano le distribuzioni di probabilità condizionate per i DAG, tali che per ogni assegnazione ammissibile di valori ai membri di Θ, la distribuzione di probabilità congiunta di A è data dal prodotto di tali distribuzioni condizionate e soddisfa la Condizione di Markov con il DAG. Si noti che una volta assegnato Θ è possibile dedurre G dalle relazioni di dipendenza condizionata descritte dai parametri.

Per selezionare il modello ottimale utilizziamo un criterio di scoring. In particolare costruiamo una funzione che assegni un valore $score(M, D)$ a ciascun candidato DAG M rispetto ai dati osservati D. Tale funzione è progettata in modo da assegnare valori tanto maggiori quanto più la distribuzione congiunta delle variabili $P[A]$ si discosta da $P_N[A]$, distribuzione stimata dalla rete N. Scegliamo come funzione di scoring la KL-divergence, che mappa due distribuzioni su uno stesso spazio misurabile in $\mathbb{R}^+$, quantificandone la differenza nel senso delle distribuzioni. In particolare è definita come segue [10]:

Definizione 2.17. *Siano P e Q due misure di probabilità su uno spazio misurabile $(\Omega, \mathcal{F})$. La divergenza di Kullback tra P e Q è definita come:*

$$D_{KL}(P|Q) = \begin{cases} \int_\Omega \log(\frac{dP}{dQ})\, dP & P \ll Q; \\ +\infty & \textit{Altrimenti.} \end{cases} \tag{2.5}$$

Tale funzione è sempre ben definita come si dimostra in Appendice A. Inoltre non è difficile mostrare che $D_{KL}(P, Q) \geq 0 \ \forall P,\ Q \in \Pi$ insieme delle misure di probabilità sullo spazio misurabile $(\Omega, \mathcal{F})$. La Divergenza di Kullback Leibler non è una distanza perchè non è simmetrica, ma appartiene alla famiglia delle f-divergenze.

L'interpretazione probabilistica che si attribuisce alla Divergenza di Kullback Leibler tra due distribuzioni P e Q consiste nell'informazione persa quando Q è usata per approssimare P. Segue immediatamente che possiamo usare tale funzione per selezionare G^* tra i candidati G che minimizzi $D_{KL}(P|P_G)$ rispetto ai dati contenuti in D.

2.2. Algoritmo di generazione dei dati sintetici

In questa sezione presentiamo l'algoritmo utilizzato per la generazione di dati sintetici a partire dal dataset a nostra disposizione, nel quale sono presenti variabili numeriche e categoriche relative agli utenti che interagiscono con il nostro sito. Le variabili sono sia caratteristiche proprie dell'utente sia relative al comportamento degli stessi durante il flusso per sottoscrivere un preventivo, e verranno specificate in modo più approfondito nel capitolo 3. La struttura dell'algoritmo è formata da tre sezioni principali che vengono eseguite nel seguente ordine:

1. **Costruzione di una rete Bayesiana con tecnica di greedy search.** In questa sezione avviene la selezione del modello ottimale condizionatamente ai dati, affinchè a ciascuna variabile X_i è associato un insieme Π_i di al più k genitori massimizzando l'Informazione mutua tra X_i e Π_i.

2. **Stima delle distribuzioni condizionate con aggiunta di rumore.** Dapprima per ciascuna variabile X_i avviene la stima della distribuzione congiunta $P[X_i, \Pi_i]$. Viene successivamente iniettato un rumore con distribuzione di Laplace centrata in 0 in ciascuna delle $P[X_i, \Pi_i]$ e successivamente normalizzate. È possibile controllare la varianza del rumore aggiunto tramite un parametro ε, per regolare la distanza tra $P[X_i, \Pi_i]$ e $P^*_N[X_i, \Pi_i]$ ovvero le distribuzioni sporcate dal rumore. Se $\varepsilon = 0$ nessun rumore viene aggiunto. In un secondo momento vengono dedotte le distribuzioni condizionate $P^*_N[X_i|\Pi_i]$ da $P^*_N[X_i, \Pi_i]$.

3. **Campionamento dei dati sintetici a partire dalle distribuzioni stimate.** È possibile generare un dataset sintetico D^* di dimensione arbitraria, che in seguito verrà scelto pari alla dimensione del dataset reale n, eseguendo un sampling da $P^*_N[A]$.

In seguito ci proponiamo di illustrare nel dettaglio ciascuna di esse.

2.2.1. Costruzione della reta Bayesiana

La rete bayesiana è progettata a partire dall'idea di Chow e Liu di aggiungere ad ogni iterazione la coppia (X_i, Π_i) che massimizzi la dipendenza diretta estendendola al caso in cui $k > 1$, migliorando la precisione nella stima di $P_N[A]$ rispetto alle reti con al massimo 1 genitore. L'algoritmo prende in input un dataset D formato da n righe e d colonne, ciascuna delle quali è un campione di n valori nelle variabili aleatorie contenute in $A = X_1, \ldots, X_d$, e restituisce una lista G formata da d coppie $(X_1, \Pi_1), (X_2, \Pi_2), \ldots, (X_d, \Pi_d)$ che rappresentano un DAG che associa ad ogni nodo un insieme di genitori che lo precedono diretta-

mente nel grafo. Per quanto riguarda la stima delle dipendenze dirette in G, ricordando che viene usata la Divergenza di Kullback Leibler tra $P[A]$ e $P_N[A]$ per la selezione del modello, si riscrive $D_{KL}(P|P_N)$ in funzione dell'entropia $H(\cdot)$ e dell'informazione reciproca $I(\cdot,\cdot)$ come mostrato in [12]:

$$D_{KL}(P|P_N) = -\sum_{i=1}^{d} I(X_i, \Pi_i) + \sum_{i=1}^{d} H(X_i) - H(A). \tag{2.6}$$

Si noti che nell'equazione (2.6) il termine $\sum_{i=1}^{d} H(X_i) - H(A)$ non dipende dalla scelta del grafo ma solo dai dati, per cui il problema risulta equivalente a massimizzare il termine $\sum_{i=1}^{d} I(X_i,\ \Pi_i)$, ovvero l'Informazione mutua totale delle variabili e dei propri genitori. Nel caso in cui $k = 1$, ricordando che k è la massima cardinalità dell'insieme dei genitori di ciascun nodo, Chow e Liu dimostrano che il massimo è raggiunto aggiungendo ad ogni step l'arco direzionato che massimizza l'Informazione mutua all'iterazione corrente [4]. Tuttavia, come dimostrato in [3], questo problema di ottimizzazione è NP-difficile quando $k > 1$. Pertanto, cerchiamo un metodo efficiente che estenda tale intuizione.

Proponiamo quindi un algoritmo greedy in cui ad ogni iterazione venga massimizzata $I(X_i, \Pi_i)$ e in cui Π_i sia scelto tra i sottoinsiemi di V di dimensione $\min\{k, |V|\}$, che chiamiamo $\binom{V}{k}$, in cui V è l'insieme dei non discendenti di X_i e viene riaggiornato per ogni variabile. Dunque se $|V| < k$ allora $\Pi_i = V$. L'algoritmo è descritto in 2.1: esso prende in input un database D e due parametri ε e θ e nel dettaglio allo step 1 viene inizializzato il DAG G e l'insieme V. Al passo successivo viene scelta casualmente una variabile da A, che chiamiamo X_1, e imponiamo $\Pi_1 = \emptyset$. Il resto dell'algoritmo consiste in $d-1$ iterazioni (righe 3-7), in ognuna delle quali aggiungiamo a G una coppia (X_i, Π_i) da un insieme di candidati Q (4) che contiene tutte le coppie di AP (X, Π) che soddisfino tre requisiti:

1. $|\Pi| = \min\{k, |V|\}\, k$. Ciò è garantito dalla scelta di Π solo da $\binom{V}{k}$.
2. G non contiene archi da X_i a X_j per qualsiasi $j < i$, il che garantisce che G sia un DAG, per il fatto che V, all'inizio di ogni iterazione, contiene solo gli attributi i cui insiemi di genitori sono stati decisi nelle iterazioni precedenti (9). In altre parole, l'insieme dei genitori di X_i può essere un sottoinsieme di $X_1, X_2, ..., X_{i-1}$.
3. Dato X_i è necessario che $|V_{\Pi_i}| \leq \frac{n\varepsilon}{4d|V_{X_i}|\theta}$, ovvero che il numero di valori distinti assunti da Π_i non sia maggiore di $\frac{n\varepsilon}{4d|V_{X_i}|\theta}$ dove V_X è l'insieme dei valori distinti assunti da X. Tale condizione è legata alla robustezza delle distribuzioni condizionate che verranno stimate nella seconda parte dell'algoritmo. Verrà formita una spiega-

zione più dettagliata sulla questione nella sezione 2.2.2. Definiamo Ω_i l'insieme dei genitori di X_i che soddisfino questo vincolo.

Algorithm 2.1 GreedyBayes(D, ε, θ): return G

1: $G = \emptyset, V = \emptyset$.
2: Scegliere randomicamente X_1 da A. Aggiungere $(X_1, \emptyset)$ a G, aggiungere X_1 a V.
3: **for** $i = 2, ..., d$ **do**
4: Inizializzazione di $Q = \emptyset$.
5: **for all** $X \in A \setminus V$ **do**
6: Per ogni $\Pi \in \binom{V}{k} \cap \Omega_i$ aggiungere (X, Π) a Q.
7: **end for**
8: Selezionare (X_i, Π_i) da Q che massimizzi $I(X_i|\Pi_i)$.
9: Aggiungere (X_i, Π_i) a G. Aggiungere X_i a V.
10: **end for**
11: Return G.

Si noti che i parametri ε e θ sono utilizzati allo step 6 nella verifica del terzo requisito elencato in precedenza. Una volta che è stata calcolata la rete N, si procede con la stima delle distribuzioni condizionate.

2.2.2. Stima delle distribuzioni condizionate con aggiunta di rumore

Dato il grafo direzionato G costruito nella sezione precedente, l'obiettivo è stimare l'insieme dei parametri della rete Bayesiana $\Theta = \left\{\theta_{X_1}, \theta_{X_2|X_1} ..., \theta_{X_d|\Pi_d}\right\} = \{P[X_1], ..., P[X_d|\Pi_d]\}$, ovvero l'insieme delle distribuzioni condizionate. La distribuzione congiunta modellata da N è descritta dalla seguente equazione:

$$P_N[A] = \prod_{i=1}^{d} \theta_{X_i|\Pi_i} = \prod_{i=1}^{d} P[X_i|\Pi_i]. \tag{2.7}$$

Per la stima dei parametri $\theta_{X_i|\Pi_i}$ è stato utilizzato l'algoritmo *NoisyConditionals* all'interno della libreria python *DataSynthesizer* descritto in Algorithm 2.2, che riceve in input un dataset D formato da d colonne, un DAG G che rappresenta una rete Bayesiana di grado k e un parametro ε che regola l'intensità del rumore aggiunto alle densità congiunte. Ad ogni iterazione viene calcolata la distribuzione congiunta di ciascuna coppia variabile-genitori,

iniettato un rumore distribuito secondo Laplace e dedotte le distribuzioni condizionate. In particolare allo step 1 avviene l'inizializzazione dell'insieme delle distribuzioni Θ con l'insieme vuoto e alla riga 2 viene calcolato il grado k della rete. A questo punto vengono stimate le distribuzioni $P[X_i, \Pi_i]$ distinguendo il caso in cui $i > k$ e $i \leq k$. Nel primo caso (4) vengono dapprima dedotte le distribuzioni $P[X_i, \Pi_i]$ direttamente dai dati e, nel caso in cui $\varepsilon > 0$, vengono sporcate con l'aggiunta di un rumore secondo una distribuzione di $Laplace(0, \frac{4d}{n\varepsilon})$ (6). In seguito alla riga 10 si impone che $P^*[X_i, \Pi_i]$ siano delle densità fissando a 0 eventuali valori negativi e normalizzando, vengono dedotte le distribuzioni marginali (riga 11) e aggiunte a Θ in riga 12. Nel secondo caso invece vengono dedotte $P^*[X_i|\Pi_i]$ direttamente da $P^*[X_{k+1}, \Pi_{k+1}]$ senza ulteriori interazioni con i dati e in riga 16 aggiunte a Θ. Ciò è possibile perchè per costruzione $X_i \in \Pi_{k+1}, \Pi_i \subset \Pi_{k+i}\ \forall i = 1, ..., k$.

Algorithm 2.2 NoisyConditionals(D, G, ε): return Θ

1: Inizializzare $\Theta = \emptyset$.
2: Calcolare k da G.
3: **for** $i = k + 1, ..., d$ **do**
4: Stimare la distribuzione $P[X_i, \Pi_i]$.
5: **if** $\varepsilon > 0$ **then**
6: Generare $P^*[X_i, \Pi_i]$ aggiungendo a $P[X_i, \Pi_i]$ un rumore $\sim Laplace(0, \frac{4d}{n\varepsilon})$.
7: **else if** $\varepsilon = 0$ **then**
8: $P^*[X_i, \Pi_i] = P[X_i, \Pi_i]$.
9: **end if**
10: Assegnare a 0 i valori negativi di $P^*[X_i, \Pi_i]$ e normalizzare.
11: Derivare $P^*[X_i|\Pi_i]$ da $P^*[X_i, \Pi_i]$.
12: $\theta_{X_i|\Pi_i} = P^*[X_i|\Pi_i]$. Aggiungere $\theta_{X_i|\Pi_i}$ a Θ.
13: **end for**
14: **for** $i = 1, ..., k$ **do**
15: Derivare $P^*[X_i|\Pi_i]$ da $P^*[X_{k+1}, \Pi_{k+1}]$.
16: $\theta_{X_i|\Pi_i} = P^*[X_i|\Pi_i]$. Aggiungere $\theta_{X_i|\Pi_i}$ a Θ.
17: **end for**
18: Return Θ.

A questo punto il nostro modello per stimare le relazioni di dipendenza delle variabili in D è formato da una rete Bayesiana $N = (G, \Theta)$ che riproduca in G tutte le dipendenze dirette e in Θ le relative distribuzioni condizionate. La distribuzione congiunta di A stimata da N è regolata dall'equazione (2.7) ed è tale che la divergenza di Kullback tra $P[A]$ e $P_N[A]$ sia ridotta sfruttando l'equazione (2.6) e massimizzando ad ogni iterazione

l'informazione mutua di ogni coppia variabile-genitori, scegliendo il primo nodo in modo casuale, come descritto nell'algoritmo 2. L'aggiunta eventuale di un rumore laplaciano alle distribuzioni condizionate permette di poter regolare la distanza tra le distribuzioni che stimiamo dai dati reali e quelle da cui vogliamo samplare i dati sintetici: questo è utile perchè talvolta non si desidera solamente replicare il campione osservato, ma esplorare in maniera più continua lo spazio delle variabili in A. Tecnicamente tale scelta è regolata dal parametro $\varepsilon \geq 0$: se $\varepsilon = 0$ non viene aggiunto alcun rumore perciò la produzione dei dati sintetici avviene dalla stessa distribuzione congiunta che viene dedotta da D, mentre se $\varepsilon > 0$ alle distribuzioni nodo-genitori $P[X_i, \Pi_i]$ è aggiunto un rumore distribuito secondo Laplace con media nulla e varianza inversamente proporzionale a ε^2, perciò per bassi valori di ε corrisponde un elevato rumore nelle distribuzioni di campionamento $P^*[X_i, \Pi_i]$. Forzando le distribuzioni condizionate a discostarsi da quelle osservate si favorisce quindi il campionamento di osservazioni meno frequenti o di outlier, aumentando quindi la ricchezza informativa dei nostri dati. Questo aspetto può avere anche un forte impatto a livello di business per un'azienda: è possibile ad esempio creare nuovi cluster nei profili dei clienti, simulando una più ampia gamma di profilazione dei prospect, per testare eventuali modelli predittivi di vendita dei prodotti su clienti che difficilmente entrano nel database aziendale.

Scelta di k e θ-usefulness

Abbiamo discusso della costruzione di una rete Bayesiana di grado k, parametro considerato un'informazione di input dell'algoritmo. Tuttavia k è sconosciuto e deve essere scelto con attenzione. La scelta di k non è banale: come precedentemente accennato in modo intuitivo una rete Bayesiana con un k maggiore è più informativa sulla distribuzione congiunta $P[A]$; ad esempio, una rete Bayesiana di grado $d-1$ approssima perfettamente $P[A]$ senza alcuna perdita di informazione. Tuttavia, l'uso di un k troppo elevato presenta uno svantaggio: costringe *NoisyConditionals* ad anonimizzare un insieme di distribuzioni di dimensioni elevate, le quali sono molto sensibili al rumore a causa della multidimensionalità dei loro domini [12]. Queste distribuzioni sporcate sono meno utili ai fini del campionamento, soprattutto quando ε è basso. Di conseguenza, il database sintetico non è più rappresentativo dei dati reali. Con valori molto piccoli di ε, la scelta migliore potrebbe essere quella di scegliere $k = 0$, cioè modellare tutte le variabili in modo indipendente. Pertanto, la scelta di k dovrebbe bilanciare l'informatività di una rete Bayesiana e la robustezza delle distribuzioni congiunte variabile-genitore. Questo bilanciamento è influenzato da tre parametri: la varianza del rumore iniettato che dipende da ε, il numero totale di osservazioni n nel database e un terzo parametro, chiamato in [12] θ *usefulness*,

che misura quanto una distribuzione è informativa rispetto al rumore che è stato iniettato in essa. La definizione di θ-usefulness è riportata in seguito [12] insieme alle definizioni di *scala media di informazione* e *scala media di rumore* dalle quali la prima discende:

Definizione 2.18. *Sia* (X, Π) *una coppia nodo-genitori in* N*. Siano* $supp(X)$ *e* $supp(\Pi)$ *rispettivamente il supporto di* X *e di* Π*. Siano inoltre* $S_X = X_1, ..., X_n \overset{\text{iid}}{\sim} X \in D$ *e* $S_\Pi = \Pi_1, ..., \Pi_n \overset{\text{iid}}{\sim} \Pi \in D$ *due campioni di* X *e* Π *presenti nel dataset reale. Definiamo infine* $V_X = \{x \in supp(X) : x \text{ è un valore distinto di } S_X\}$ *e* $V_\Pi = \{\pi \in supp(\Pi) : \pi \text{ è un valore distinto di } S_\Pi\}$ *gli insiemi di valori distinti assunti nei campioni* S_X *e* S_Π *rispettivamente da* X *e* Π*. Sotto l'ipotesi semplificativa di probabilità uniforme su* (X, Π) *la scala media di informazione di* $P_N^*[X, \Pi] = \frac{1}{|V_X||V_\Pi|}$*, ovvero l'inverso della cardinalità del prodotto cartesiano tra* S_X *e* S_Π*, mentre la scala media di rumore è pari a* $\frac{4d}{n\varepsilon}$*, parametro della distribuzione di Laplace.*

Definizione 2.19. *Una distribuzione rumorosa è* θ*-useful se il rapporto tra la scala media di informazione e la scala media di rumore non è inferiore a* θ*.*

Nella definizione precedente con il termine *scala media di informazione* si intente la quantità media di informazione contenuta in ciascuna cella di $P_N^*[X_i, \Pi_i]$, mentre la *scala media di rumore* è l'intensità media di rumore iniettato in ciascuna di esse e coincide con il secondo parametro della distribuzione di Laplace in *NoisyConditionals* (Algoritmo 2.2) alla riga 6.

Pertanto, $P_N^*[X, \Pi]$ è θ-useful se e solo se $|V_\Pi| \leq \frac{n\varepsilon}{4d|V_X|\theta}$. In altre parole, dato X, siamo interessati solo a quei sottoinsiemi di $\Pi \in \binom{V}{k}$ in *GreedyBayes* 2.1 riga 6 che appartengano anche ad Ω_i, ossia i cui valori distinti non siano maggiori di $\frac{n\varepsilon}{4d|V_X|\theta}$.

La nozione di θ-usefulness fornisce un metodo più intuitivo per selezionare automaticamente il valore di k senza dover analizzare in maniera approfondita il database D. In generale, riteniamo che una distribuzione rumorosa con θ-usefulness 0.5 non sia buona poiché il rumore è due volte più ampio rispetto alle informazioni fornite, mentre una distribuzione con utilità 5 è considerata più affidabile grazie al suo elevato rapporto informazione/rumore. Nella pratica, viene definita una soglia θ e si seleziona il valore intero positivo più grande di k che garantisca un'utilità non minore di tale parametro (è importante notare che questa scelta è indipendente dai dati in quanto dipende solo dai valori di ε, θ, n e d). Nel caso in cui non esista un valore di k che soddisfi questa condizione, k viene fissato al valore di 3 come consigliato in [12].

In conclusione secondo questa logica la scelta di k è guidata interamente (fissati ε e D) dal valore di θ. Come dimostrato empiricamente in [12] un range ottimale per esso consiste

in $[2, 6]$, poichè se θ è troppo basso otterremo delle marginali che risentono eccessivamente dell'aggiunta di rumore, se invece θ è sovrastimato la rete Bayesiana non sarà sufficientemente informativa. Dunque coerentemente con la precedente analisi decidiamo di condividere la decisione del suddetto paper fissando $\theta = 4$.

Infine nella sezione seguente concludiamo la descrizione dell'algoritmo di generazione dei dati sintetici, spiegando come avviene il campionamento dei dati. Tale procedura è anche la meno complessa e computazionalmente meno dispendiosa.

2.2.3. Campionamento dei dati sintetici

Per quanto sia sintetica l'equazione (2.7), risulta comunque dispendioso campionare direttamente da $P_N^*[A]$, calcolando il valore di probabilità per ogni elemento nel dominio di A. Fortunatamente, la rete bayesiana N offre un metodo efficiente per il campionamento senza dover materializzare $P_N^*[A]$. Come mostrato nell'equazione (2.7), possiamo campionare ogni X_i dalla distribuzione condizionata $P_N^*[X_i|\Pi_i]$ in modo indipendente, senza considerare variabili non presenti in $\Pi_i \cup X_i$. Inoltre, le proprietà di N (discusse nella Sezione 2.2.1) garantiscono che $X_j \notin \Pi_i$ per ogni $j > i$. Pertanto, se campioniamo X_i $(i \in 1, ..., d)$ in ordine crescente di i, al momento del sampling di X_j $(j \in 2, ..., d)$ avremo già campionato tutti gli attributi in Π_j, ovvero saremo in grado di generare X_j da $P_N^*[X_j|\Pi_j]$. In altre parole, il campionamento di X_j non richiede l'intera distribuzione $P_N^*[A]$.

Utilizzando l'approccio di sampling descritto sopra, possiamo generare un numero arbitrario di tuple da $P_N^*[A]$ per creare un database sintetico D^*. In questa tesi, supponiamo che la dimensione di D^* sia fissata a n, cioè uguale al numero di tuple presenti nei dati reali D. Ovviamente, potrebbe essere più adatto in alcune particolari analisi generare un numero diverso di dati sintetici, tuttavia campionare lo stesso numero di osservazioni ci permette di avere un database sintetico direttamente confrontabile con l'input. Se l'ipotesi di modellazione è valida (cioè, i dati sono ben modellati da una rete Bayesiana), possiamo immaginare che l'input originale D sia un campione di n tuple da un modello Bayesiano, pertanto la scelta di campionare questo numero di osservazioni è appropriata.

2.3. Dati sintetici generati

In questa sezione del capitolo saranno mostrati i dati sintetici che sono stati campionati al variare del parametro ε, confrontandoli con quelli reali e valutandone la verosimiglianza. Il campionamento dei dati sintetici è eseguito principalmente in due fasi separate, che

riflettono le lacune del dataset reale e le esigenze informative dell'analisi. In particolare in un primo passaggio viene replicato il dataset per intero, scegliendo la dimensione del campione desiderato, con l'obiettivo di ricostruire le strutture generali presenti nei dati reali e raggiungere la numerosità campionaria desiderata. In seguito si lavora in modo più mirato arricchendo particolari punti dello spazio campionario che rappresentano lacune informative o cluster specifici che presentano caratteristiche comuni tra gli utenti. In questo modo è possibile sia aumentare conoscenze più specifiche riguardanti i clienti, sia migliorare aspetti mirati dell'algoritmo di classificazione: ad esempio è stata evidenziata la capacità non ottimale del modello multinomiale nel classificare i clienti non gestiti, dovuto principalmente ad una numerosità non adeguata dei dati. Dunque è stato individuato un cluster di utenti non gestiti, con caratteristiche il più diverse possibili dagli utenti delle altre due classi, ed è stato eseguito un campione di replica utilizzato nel training set insieme ai dati reali, nel tentativo di migliorare la sensibilità del modello rispetto a questa classe.

La valutazione circa la qualità dei dati generati è effettuata sia qualitativamente, confrontando le distribuzioni congiunte e marginali dei dati reali e sintetici, sia quantitativamente attraverso due indicatori chiamati *propensity score MSE* e *propensity score MSE ratio*, i quali forniscono una stima del livello di verosimiglianza dei dati finti.

I propensity score rappresentano le probabilità di appartenenza ad un gruppo ed è un concetto comunemente utilizzato negli studi di inferenza statistica. Per utilizzarli come misura di utilità dei dati sintetici è necessario modellare l'appartenenza ai gruppi tra i dati originali e quelli mascherati per ottenere una stima della loro distinguibilità. Una bassa distinguibilità è correlata a un'elevata similarità distribuzionale tra i dati originali e quelli mascherati [8]. In caso di una stima accurata dei propensity score, questa misura generale dovrebbe catturare relazioni tra i dati che metodi come la funzione di ripartizione empirica potrebbero trascurare [8]. Il metodo di propensity score, descritto nell'articolo [11] e presentato nell'algoritmo 2.3, procede nel seguente modo. Viene specificato un insieme di variabili predittive per i dati originali e quelli sintetici. I due set di dati vengono combinati aggiungendo una variabile indicatrice che indica la fonte dei dati (0 per i dati originali e 1 per quelli finti). Viene quindi stimato un punteggio di propensione per ciascuna riga del dato combinato, come probabilità di classificazione per la variabile indicatrice, e calcolato lo scarto quadratico medio rispetto alla probabilità di classificazione casuale, chiamato propensity score MSE.

Algorithm 2.3 Propensity score MSE

1: Unire le n_R osservazioni del dataset reale D_R con le n_S osservazioni del dataset sintetico D_S, per formare $N = n_R + n_S$ righe del dataset combinato D_C.
2: Aggiungere al dataset D_C una variabile indicatrice $I = \{1 : x_i \in D_S; 0 : x_i \in D_R\}$.
3: Fittare un modello di classificazione per predirre I rispetto alle variabili specificate dal dataset D_C.
4: Preditte il propensity score $\hat{p}_i$ per ogni riga di D_C.
5: Calcolare lo scarto quadratico medio dei propensity score come $\frac{1}{N}\sum_{i=1}^{N}(\hat{p}_i - c)^2$, dove $c = \frac{n_S}{N}$.

Supponendo che D_C sia perfettamente bilanciato, per cui $c = 0.5$, allora pMSE è compreso tra 0 e 0.25. Il massimo è raggiunto nel caso in cui il classificatore interpoli perfettamente ciascun osservazione, assegnando sempre una probabilità di 0 o 1, caso in cui la distinzione tra dati reali e finti risulta netta. Invece i migliori risultati sono raggiunti in prossimità di $pMSE = 0$, in corrispondenza del quale i dati originali e quelli mascherati sono identici. Questo è altamente improbabile per i dati sintetici, poiché l'obiettivo non è avere osservazioni identiche, ma ottenere una somiglianza distribuzionale tra i dati osservati e il modello utilizzato per generare i dati sintetici. Questa condizione è necessaria affinché qualsiasi inferenza dai dati sintetici sia valida, e la chiameremo sintesi corretta e, quando ci riferiamo alla distribuzione nulla di una statistica, questo implicherà la distribuzione in caso di sintesi corretta.

Per la classificazione delle osservazioni è possibile usare sia metodi parametrici, sia algoritmi più avanzati come random forest o CART. In questa analisi è stato utilizzato un modello logistico con termini di primo grado nelle variabili, che permette, grazie allo studio condotto in [8], di sfruttare un'importante risultato sulla distribuzione nulla di pMSE, proporzionale a una $\chi^2(k-1)$, dove k è pari al numero di regressori, comprese le interazioni. Da ciò si deduce il valore atteso nullo di pMSE, ovvero:

$$\mathbb{E}[pMSE] = (k-1)(1-c)^2 c/N. \tag{2.8}$$

Nel caso in cui $c = 0.5$, si deduce che $\mathbb{E}[pMSE] = (k-1)/8N$. Con l'espressione del valore atteso nullo di pMSE è possibile costruire una seconda statistica, il propensity score MSE ratio, calcolato come il rapporto tra il pMSE e il suo valore atteso nullo. Valori elevati di questa statistica sono attesi se l'ipotesi di sintesi corretta non è validata.

Sebbene lo studio di Woo ([11]) sottolinei l'importanza di inserire regressori nel modello

logistico di ordine superiore al primo, questa pratica si è trovata in conflitto con una stima accurata dei coefficienti β: fittando il modello con tutte le covariate al primo ordine e tutte le interazioni tra variabili numeriche e categoriche la stima dei coefficienti è risultata troppo approssimativa senza raggiungere la convergenza del risolutore. Per questo motivo è stato necessario limitare i regressori al primo ordine escludendo anche le interazioni tra essi.

Il propensity score MSE e il pMSE ratio forniscono dunque uno strumento per quantificare al variare del parametro ε la differenza tra la distribuzione dei dati reali e quella da cui vengono generati i dati sintetici: è dunque lecito aspettarsi un minimo di pMSE in corrispondenza di $\varepsilon = 0$, e valori decrescenti nel parametro quando $\varepsilon > 0$.

2.3.1. Generazione dataset completo

Sono stati generati diversi dataset sintetici, ciascuno di 15000 osservazioni e 16 variabili descritte in tabella 1.1. Ciascun dataset è stato campionato da una distribuzione ottenuta da quella dei dati reali aggiungendo un rumore laplaciano con varianza inversamente proporzionale a ε, che è stato fatto variare su una griglia equispaziata da 0 a 1 con passo costante pari a 0.1, e tra 1 e 20 con passo costante pari a 1. Ciò permette alla distribuzione dei dati finti di discostarsi in maniera controllata dalla distribuzione reale, al fine di studiarne la verosimiglianza e l'impatto sul modello multinomiale.

I dati generati si presentano in modo del tutto analogo rispetto a quelli reali, rendendoli indistinguibili a prima vista, se non sottoposti ad un'analisi più approfondita. Le distribuzioni marginali sono in grado di replicare fedelmente quelle originali per $\varepsilon = 0$ o per alti valori di ε, sia per $K = 2$ che per $K = 3$. In seguito sono riportate le marginali della variabile *Recency*, stimate utilizzando una rete Bayesiana con due gradi di profondità, nel quale è possibile notare come al crescere di ε la distribuzione di campionamento vada ad approssimare in modo sempre più preciso la distribuzione reale. Nel caso in cui $\varepsilon = 0$ le due distribuzioni sono pressocchè sovrapposte, poichè non c'è stata aggiunta di rumore.

Si noti come effettivamente le distribuzioni marginali seguono un andamento atteso dal punto di vista teorico: indirettamente il parametro ε regola la distanza L^2 tra la distribuzione reale e quella di campionamento. È stata riportata solo la *Recency* a titolo esemplificativo ma il discorso è analogo anche per le altre variabili. È importante però che questo accada non solo per le marginali, ma anche per la distribuzione congiunta, studio in oggetto del prossimo paragrafo.

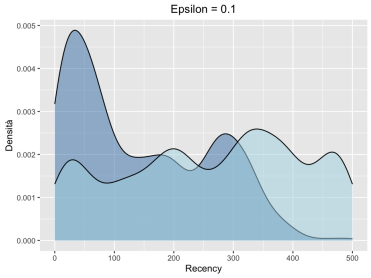

Figura 2.2: densità di *Recency*, $\varepsilon = 0.1$

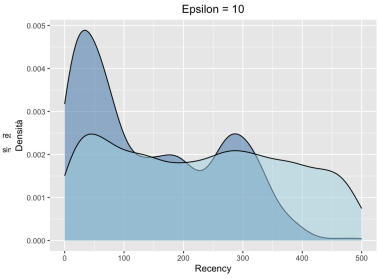

Figura 2.3: densità di *Recency*, $\varepsilon = 1$

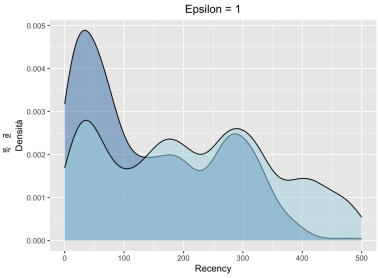

Figura 2.4: densità di *Recency*, $\varepsilon = 10$

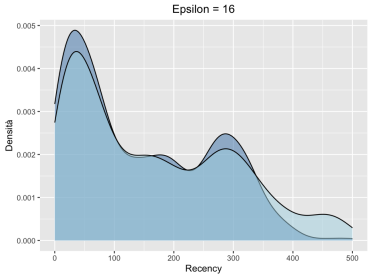

Figura 2.5: densità di *Recency*, $\varepsilon = 16$

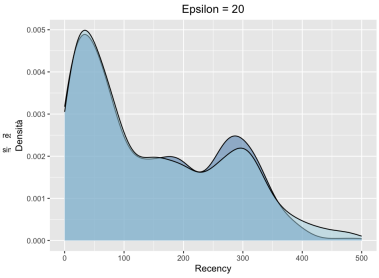

Figura 2.6: densità di *Recency*, $\varepsilon = 20$

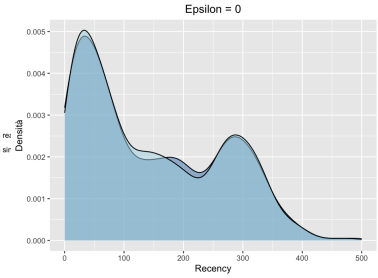

Figura 2.7: densità di *Recency*, $\varepsilon = 0$

Analisi sulla qualità dei dati sintetici

In questa sezione è stato valutato in modo sia qualitativo, attraverso dei depth-depth plot, sia quantitativo, con il calcolo di pMSE e pMSE ratio, il comportamento delle distribuzioni congiunte, in particolare in quale misura le due distribuzioni congiunte si discostassero al variare di ε.

Il "depth vs depth plot", o DDPlot, rappresenta una generalizzazione molto utile del plot quantile-quantile unidimensionale: per due distribuzioni di probabilità F e G, entrambe in $\mathbb{R}^d$, si definisce DDPlot:

$$DD(F, G) = \{(D(z, F), D(z, G)), z \in \mathbb{R}^d\},$$

in cui $D(\cdot, \cdot)$ rappresenta la Depth di un punto. Il suo corrispettivo campionario calcolato per due campioni $X^n = X_1, ..., X_n$ da F e $Y^m = Y_1, ..., Y_m$ da G è definito come:

$$DD(F_n, G_m) = \{(D(z, F_n), D(z, G_m)), z \in \{X^n \cup Y^m\}\}.$$

Tali definizioni si trovano nella documentazione del relativo pacchetto (https://cran.r-project.org/web/packages/DepthProc/DepthProc.pdf). Il significato che si attribuisce a questi grafici riguarda la differenza tra le distribuzioni F e G: se tali distribuzioni sono uguali, le osservazioni si dispongono lungo la diagonale, discostandosi da essa tanto più differiscono le due distribuzioni. In seguito sono riportati i DDPlot dei dati reali e sintetici per le variabili numeriche prodotti con l'utilizzo del pacchetto R *DepthProc*.

Dai grafici sottostanti è possibile notare come per $K = 2$ i punti risultano sempre più allineati per ε crescente, producendo un buon allineamento nel caso in cui $\varepsilon = 0$. Questo conferma quanto già riscontrato nelle distribuzioni marginali, ovvero che la distribuzione congiunta di campionamento delle variabili numeriche approssima quella dei dati reali in modo qualitativamente migliore per alti valori di ε, mentre la massima accuratezza è raggiunta per $\varepsilon = 0$. Per quanto riguarda il caso in cui $K = 3$ i dati sintetici prodotti non provengono dalla distribuzione desiderata per valori positivi di ε, sebbene la rete Bayesiana sia in grado di catturare più dettagliatamente le indipendenze condizionali tra le variabili, perchè l'anonimizzazione delle distribuzioni congiunte variabile-ascendenti produce una scala di rumore troppo elevata, dovuta ad una dimensionalità delle stesse troppo alta come dimostra anche il paper [12].

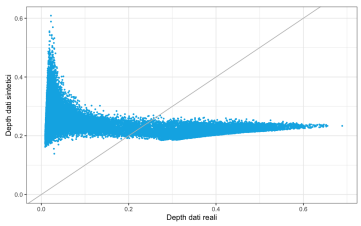

Figura 2.8: DDPlot dataset globale, $\varepsilon = 0.1$

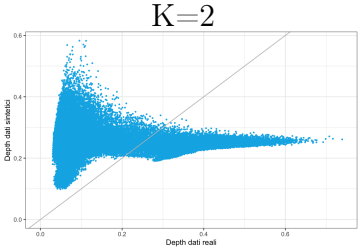

Figura 2.9: DDPlot dataset globale, $\varepsilon = 1$

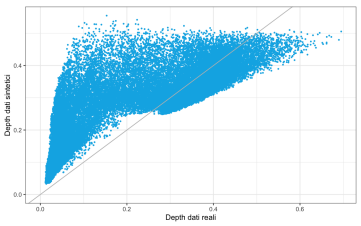

Figura 2.10: DDPlot dataset globale, $\varepsilon = 10$

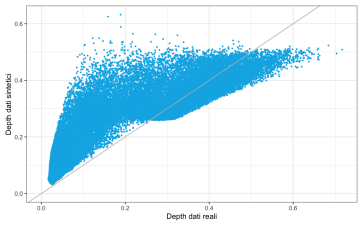

Figura 2.11: DDPlot dataset globale, $\varepsilon = 16$

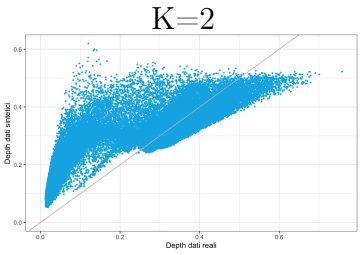

Figura 2.12: DDPlot dataset globale, $\varepsilon = 20$

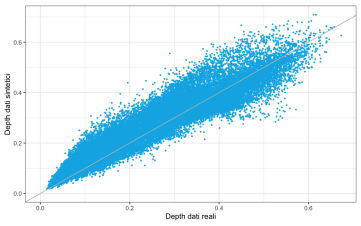

Figura 2.13: DDPlot dataset globale, $\varepsilon = 0$

I grafici di pMSE e pMSE ratio, calcolati nei casi in cui la profondità della rete Bayesiana è pari a 2 e a 3, mostrano in entrambi i casi che il minimo è raggiunto per $\varepsilon = 0$ come

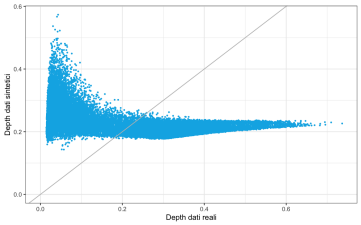

Figura 2.14: DDPlot dataset globale, $\varepsilon = 0.1$

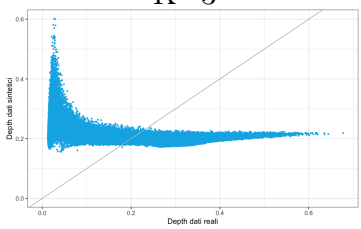

Figura 2.15: DDPlot dataset globale, $\varepsilon = 1$

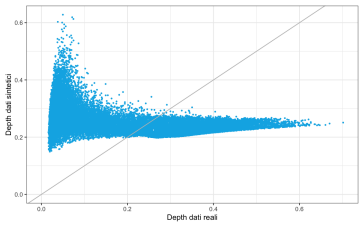

Figura 2.16: DDPlot dataset globale, $\varepsilon = 10$

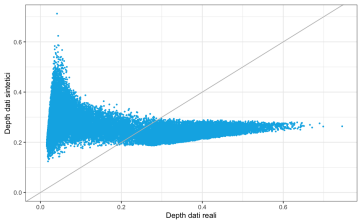

Figura 2.17: DDPlot dataset globale, $\varepsilon = 16$

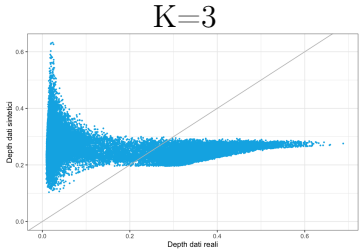

Figura 2.18: DDPlot dataset globale, $\varepsilon = 20$

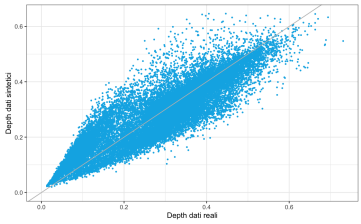

Figura 2.19: DDPlot dataset globale, $\varepsilon = 0$

lecito attendersi dalla trattazione teorica, in corrispondenza del quale pMSE è pari circa a 1, valore che conferma dunque l'ipotesi di sintesi corretta. Nel caso in cui $K = 2$ i due indicatori sono descrescenti in ε, mostrando come effettivamente l'utilità dei dati sintetici cresca al crescere del parametro, poichè la distanza tra la distribuzione dei dati reali e quella da cui sono generati i dati sintetici diminuisce, diminuendo in questo modo la distinguibilità tra i due gruppi per il classificatore logistico, come dimostra il grafico relativo all'accuratezza. Tale comportamento invece non sussiste nel caso in cui $K = 3$, in cui il modello logistico distingue quasi perfettamente le classi di appartenenza dei dati per ogni valore di ε, raggiungendo il massimo di pMSE e accuratezza.

Questi primi risultati prodotti dimostrano come la rete Bayesiana con $K = 2$ sia in grado di produrre dati sintetici più simili a quelli reali di quanto avvenga con $K = 3$ dal punto di vista distribuzionale, a causa del rumore limitato rispetto alla scala di informazione che sono in grado di produrre. I risultati appena discussi saranno importanti nella scelta degli iperparametri di campionamento, affiancando quelli relativi ai dati sintetici provenienti dagli utenti *Non gestiti* e al loro impatto sul modello multinomiale logistico.

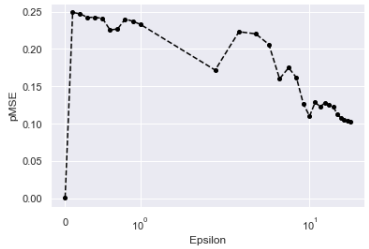

Figura 2.20: Propensity score, $K = 2$

K=2

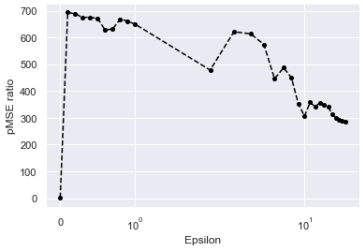

Figura 2.21: Propensity score ratio, $K = 2$

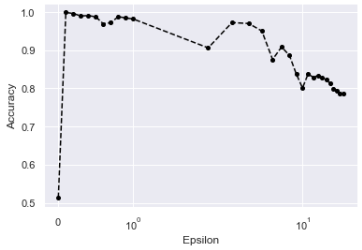

Figura 2.22: Accuratezza classificatore logistico, $K = 2$

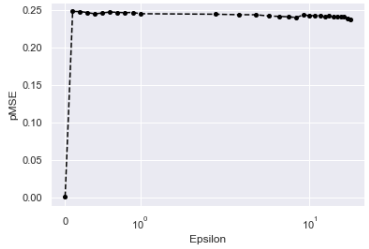

Figura 2.23: Propensity score, $K = 3$

K=3

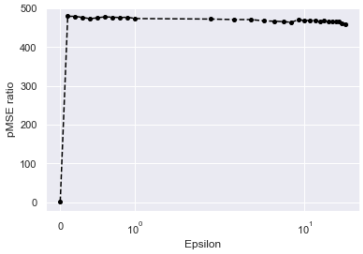

Figura 2.24: Propensity score ratio, $K = 3$

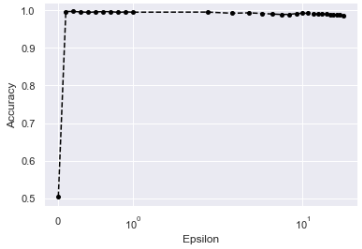

Figura 2.25: Accuratezza classificatore logistico, $K = 3$

2.3.2. Generazione dati da utenti *Non gestiti*

In una seconda fase della generazione dei dati si opera in modo chirurgico come anticipato in precedenza, intervenendo specificatamente in alcuni punti dello spazio campionario. In questo contesto l'obiettivo è effettuare un'analisi locale del dataset individuando aree in cui le informazioni sugli utenti non gestiti sono assenti e altre invece in cui le caratteristiche dei cluster sono più evidenti, al fine di individuare eventuali sottogruppi da replicare e aggiungere al training set del modello multinomiale, affinchè quest'ultimo sia più sensibile nell'individuazione e classificazione degli utenti di questa classe.

Clustering supervisionato con algoritmo UMAP

Per aumentare la capacità di previsione del modello multinomiale sugli utenti non gestiti, è stato individuato un cluster da questa classe che rappresentasse un profilo commercialmente riconoscibile, e fosse il più diverso possibile dalle altre due categorie, affinchè potesse essere aumentato attraverso un campionamento di dati sintetici. A tale scopo, per identifi-

care il target è stata eseguita una riduzione della dimensionalità dello spazio delle variabili numeriche con l'algoritmo UMAP (https://umap-learn.readthedocs.io/en/latest/supervised.html), volto a individuare le direzioni principali tali da massimizzare la separazione tra i gruppi in modo supervisionato attraverso le label di classe. AMPLIARE

Le osservazioni proiettate nello spazio ridotto sono raffigurate in 2.26.

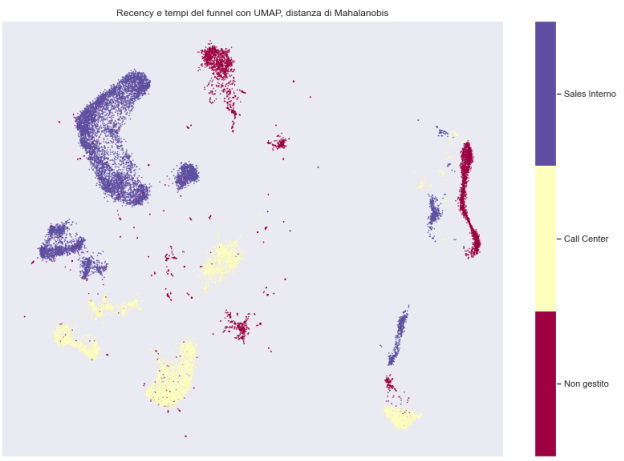

Figura 2.26: Dati reali, variabili numeriche nello spazio mappato da UMAP

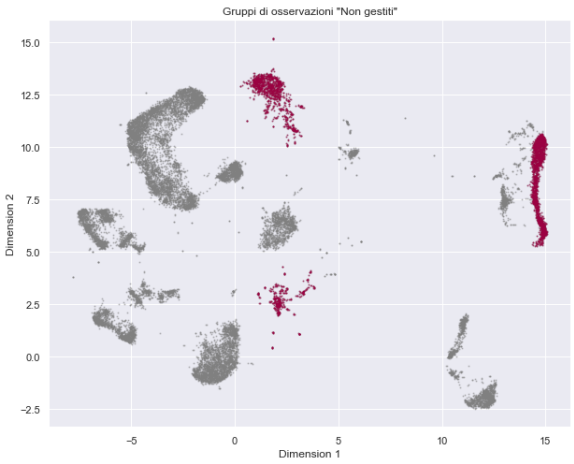

Figura 2.27: Utenti non gestiti nello spazio ridotto

Come mostrato nelle figure 2.26 e 2.27 è possibile individuare principalmente 3 sottogruppi distinti di osservazioni relative a utenti non gestiti. Ciascuno di questi 3 sottogruppi contiene delle importanti caratteristiche dal punto di vista commerciale che, una volta replicate con i dati sintetici, avranno un impatto sul nostro modello. Per questo sono state analizzate e confrontate le distribuzioni dei dati reali tra i 3 cluster, individuandone le principali differenze. In particolare le variabili categoriche *Stato Utente* e *Settore* sono quelle più esplicative, come mostrato dagli istogrammi 2.28 e 2.29. Le distribuzioni delle altre variabili sono riportate in 2.30, 2.31 e 2.32.

In particolare il cluster 2, che si trova sulla parte più a destra della figura 2.27, è formato da utenti che Lokky ha acquisito per fonti esterne, principalmente attraverso le campagne pubblicitarie o di Google (campagne *Organic Search*), oppure tramite il nostro parner principale *Facile*, con un bassissimo numero di utenti che hanno navigato sul sito attraverso il link di Lokky (*Direct*). Inoltre sono utenti che si trovano allo stato *Lightbox*, che rappresenta il primo step nella compilazione dei questionari per arrivare alla sottoscrizione della polizza. Questo significa che questi utenti sono nuovi prospect che hanno abbandonato il sito nelle prime fasi dell'iter di vendita, dunque richiedono uno sforzo maggiore nell'acquisizione del cliente perchè non conoscono il brand e perchè sono meno disposti a condividere con l'azienda i loro dati come testimonia la variabile *Stato Utente*, dunque è

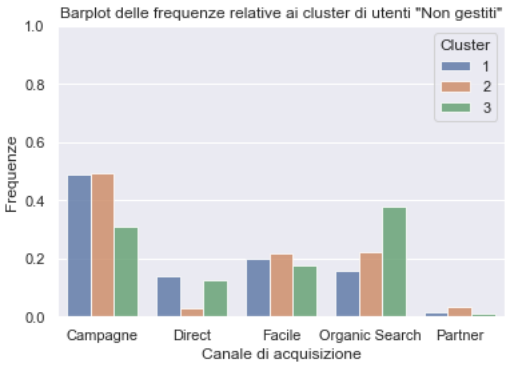

Figura 2.28: Frequenze canale di acquisizione

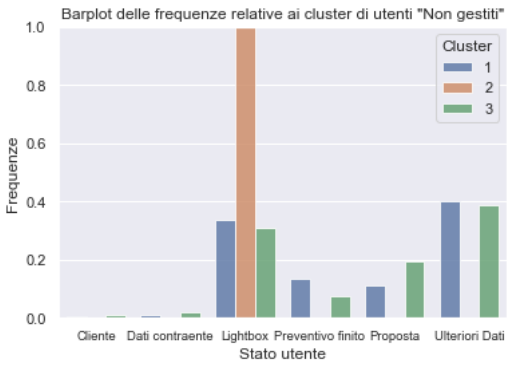

Figura 2.29: Frequenze stato utente

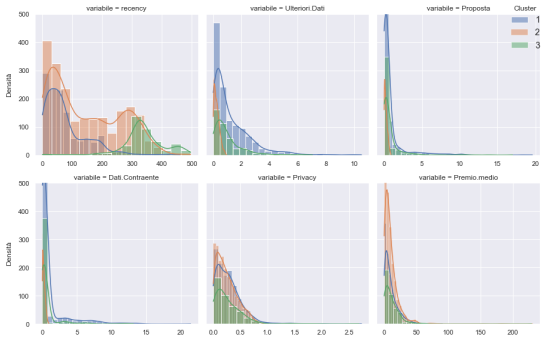

Figura 2.30: Distribuzioni utenti non gestiti, variabili numeriche

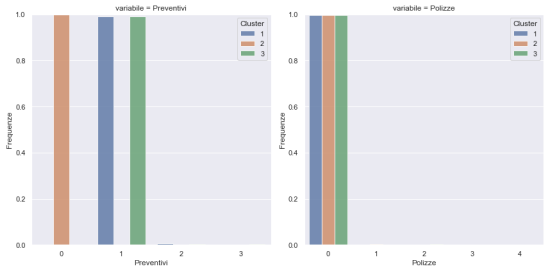

Figura 2.31: Distribuzioni utenti non gestiti, Preventivi e Polizze

necessario un dispendio superiore di risorse aziendali. Per questi motivi sono etichettati come *Non gestiti*.

Perciò sono state estratte le osservazioni dal cluster 2 ed è stato eseguito un campionamento di dati finti usando la procedura che è stata spiegata precedentemente in questo capitolo con diversi valori degli iperparametri: ε è stato fatto variare su una griglia di valori compresi tra 0 e 20, mentre la profondità della rete K è stata scelta di 2 o 3 nodi. I dati ottenuti sono illustrati in modo esemplificativo per $K = 2$ e $\varepsilon = 0, 0.5, 2, 10$. Le osservazioni rappresentate sono i dati reali e i dati sintetici rispettivamente in azzurro e rosso, ristrettamente alle variabili numeriche, che sono state proiettate sullo spazio ridotto. Il training dell'algoritmo per l'individuazioni dello spazio ridotto è avvenuto utilizzando i soli dati reali con l'algoritmo UMAP, mentre i dati sintetici sono stati proiettati. Si noti che i dati sintetici riempiono una zona dello spazio ridotto che precedentemente era vuota, nelle vicinanze di un piccolo cluster di osservazioni appartenenti al *Sales Interno*.

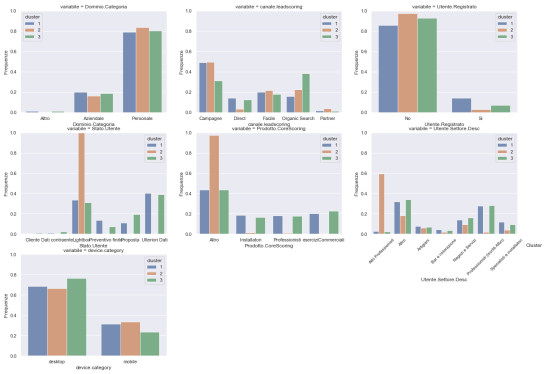

Figura 2.32: Distribuzioni utenti non gestiti, variabili categoriche

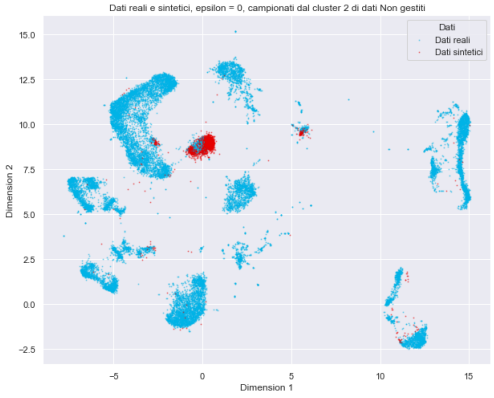

Figura 2.33: $\varepsilon = 0, K = 2$

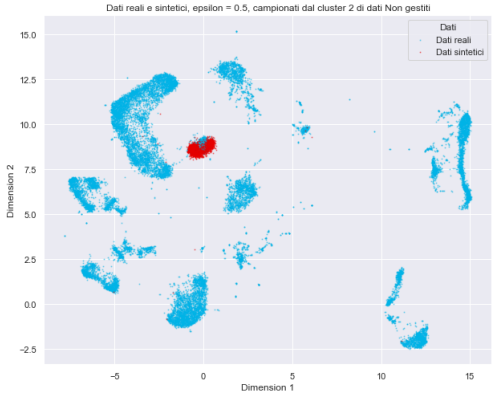

Figura 2.34: $\varepsilon = 0.5, K = 2$

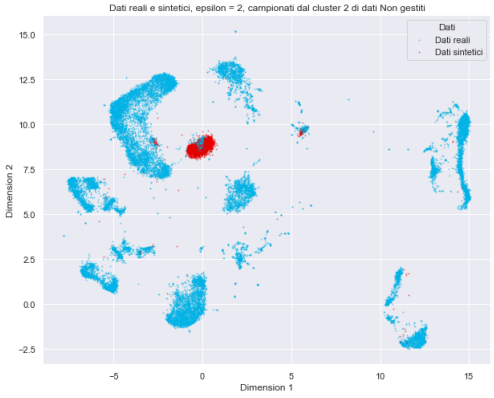

Figura 2.35: $\varepsilon = 2, K = 2$

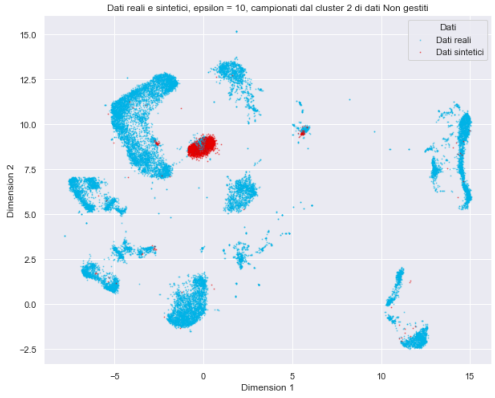

Figura 2.36: $\varepsilon = 10, K = 2$

Analisi sulla qualità dei dati sintetici

Analogamente al caso di dati sintetici provenienti dal dataset reale per intero, sono stati prodotti i grafici DDPlot, di pMSE e pMSE ratio, accompagnati dall'accuratezza del classificatore logistico, nel caso in cui $K = 2, 3$. Data la scarsa variabilità delle variabili

intere *Preventivi* e *Polizze*, sono state escluse dalle covariate della regressione logistica perchè troppo influenti nella classificazione.

I DDPlot mostrati in seguito evidenziano una migliore approssimazione della distribuzione reale nel caso in cui $K = 3$ rispetto al confronto tra dati reali e sintetici globali: le figure 2.46 e 2.47 mostrano un migliore allineamento dei dati lungo la diagonale, da cui si evince un minore rumore iniettato nelle distribuzioni variabile-ascendenti da parte dell'algoritmo 2.2. In analogia al contesto precedente sono preferiti i dati sintetici con $K = 2$ e alti valori di ε, oppure $\varepsilon = 0$.

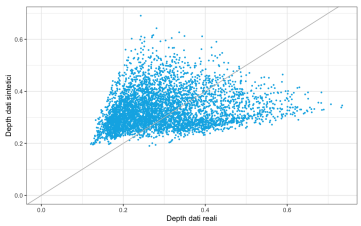

Figura 2.37: DDPlot *Non gestiti*, $\varepsilon = 0.1$

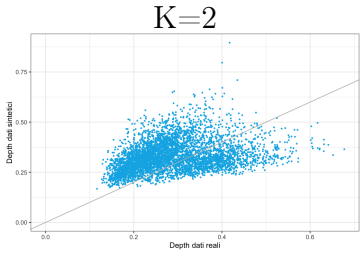

Figura 2.38: DDPlot *Non gestiti*, $\varepsilon = 1$

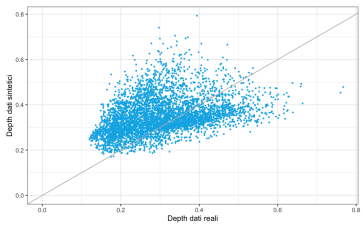

Figura 2.39: DDPlot *Non gestiti*, $\varepsilon = 10$

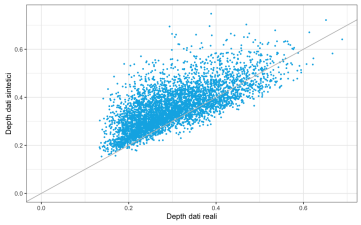

Figura 2.40: DDPlot *Non gestiti*, $\varepsilon = 16$

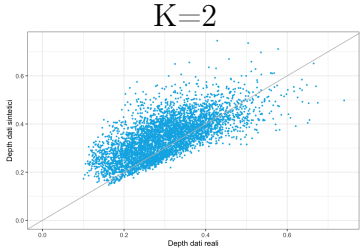

Figura 2.41: DDPlot *Non gestiti*, $\varepsilon = 20$

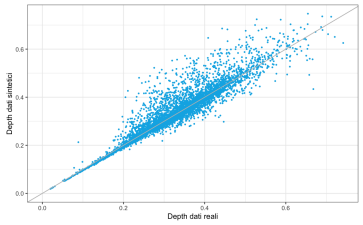

Figura 2.42: DDPlot *Non gestiti*, $\varepsilon = 0$

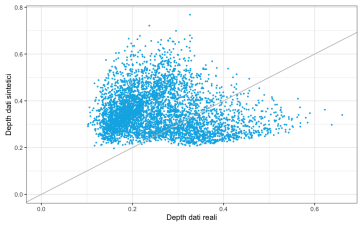

Figura 2.43: DDPlot *Non gestiti*, $\varepsilon = 0.1$

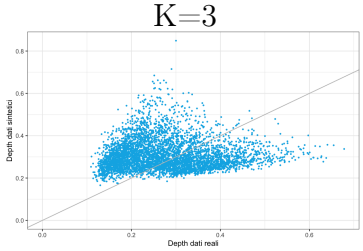

Figura 2.44: DDPlot *Non gestiti*, $\varepsilon = 1$

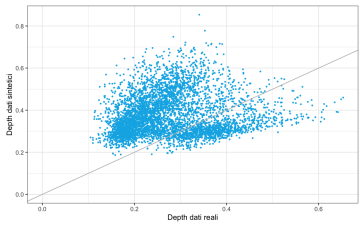

Figura 2.45: DDPlot *Non gestiti*, $\varepsilon = 10$

Inoltre da quanto si evince dal propensity score MSE e da pMSE ratio il comportamento è simile a quanto riscontrato nella generazione globale dei dati, con valori di utilità migliori

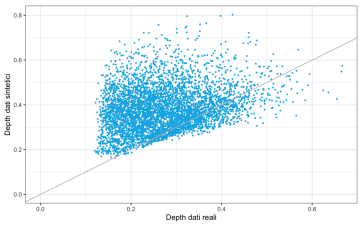

Figura 2.46: DDPlot *Non gestiti*, $\varepsilon = 16$

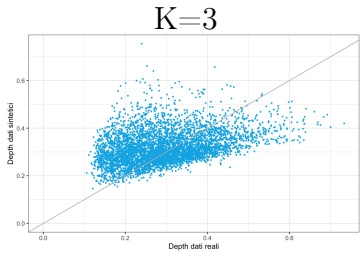

Figura 2.47: DDPlot *Non gestiti*, $\varepsilon = 20$

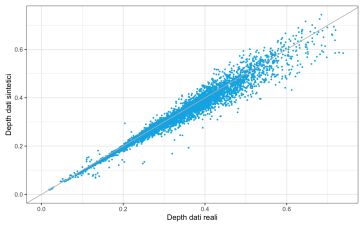

Figura 2.48: DDPlot *Non gestiti*, $\varepsilon = 0$

in corrispondenza di $K = 2$ e alti valori di ε, superiori a 10. Tuttavia l'unico caso in cui l'ipotesi di sintesi corretta non è statisticamente rifiutata è in corrispondenza di $\varepsilon = 0$, mentre per ε positivi i due dataset provengono da distribuzioni diverse.

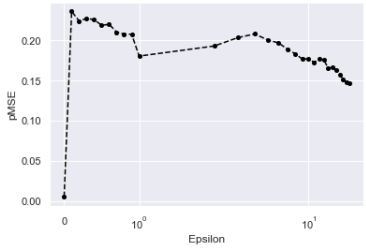

Figura 2.49: Propensity score, $K = 2$

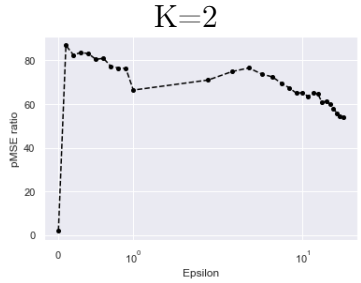

Figura 2.50: Ratio propensity score, $K = 2$

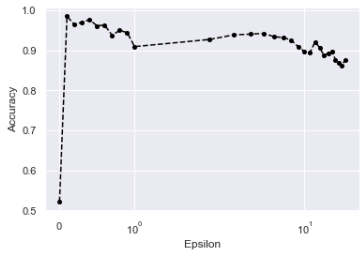

Figura 2.51: Accuratezza classificatore logistico, $K = 2$

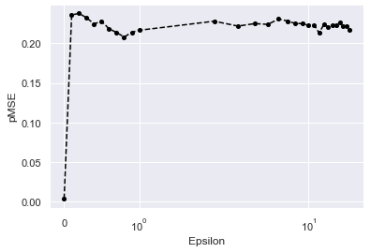

Figura 2.52: Propensity score, $K = 3$

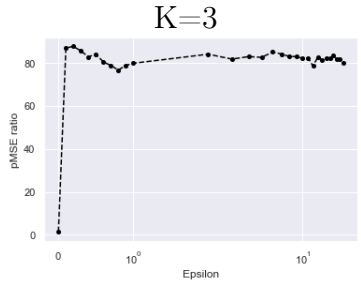

Figura 2.53: Ratio propensity score, $K = 3$

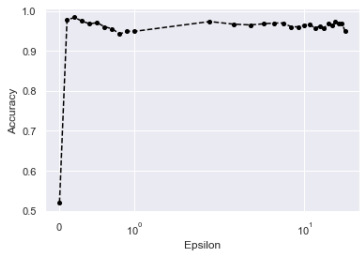

Figura 2.54: Accuratezza classificatore logistico, $K = 3$

In generale, entrambe le fasi di campionamento di dati sintetici hanno evidenziato un limite, confermato anche nello studio [12], da parte dell'algoritmo 2.2 di sporcare le distribuzioni congiunte (X_i, Π_i) (Π_i sono gli ascendenti di X_i calcolati dalla rete Bayesiana)

nel caso in cui $K = 3$ rispetto a $K = 2$, a causa della dimensione maggiore di ciascuna Π_i, che comporta una qualità peggiore dei dati sintetici. Ciò non necessariamente si riflette sulla performance del modello multinomiale, perciò entrambi i casi saranno considerati nell'analisi sull'impatto dei dati sintetici sulla qualità della classificazione, tuttavia costituiscono un risultato importante sulla scelta degli iperparametri di campionamento.

3 | Impatto dei dati sintetici sulla capacità predittiva del modello multinomiale

In questo capitolo saranno illustrati i risultati più importanti di questa tesi, ovvero l'impatto dei dati sintetici sulla prediction del classificatore multinomiale logistico che modella le interazioni tra le variabili. A tale scopo, considerando la duplice esigenza dell'azienda di aumentare la numerosità totale del proprio database e di intervenire in maniera più specifica sulla classe di utenti *Non gestiti*, sono state percorse due strade diverse ma parallele. La prima prevede semplicemente di generare dati finti dalla distribuzione generale del dataset, al fine di ampliare la struttura generale del dataframe e risolvere il primo problema. Per far fronte al secondo problema invece è stato individuato un cluster di utenti non gestiti rappresentativo da cui è stato generato un campione di dati sintetici.

In entrambe le metodologie è stata misurata la capacità predittiva del modello multinomiale su diversi dataset sintetici, campionati facendo variare il parametro ε introdotto nel capitolo 2. Si ricorda che tale parametro è in pratica una misura della distanza tra la distribuzione stimata dei dati reali e la distribuzione dalla quale si esegue il campionamento, ottenuta dalla prima a seguito di una aggiunta di rumore laplaciano [12]. L'analisi si è svolta lavorando inoltre sulla profondità della rete Bayesiana costruita nella stima delle distribuzioni congiunte, modellata dal parametro K. Si è preso in considerazione solo i casi in cui K fosse uguale a 2 o a 3: non sono stati considerati valori inferiori a 2 perchè la rete non sarebbe sufficientemente complessa da modellare le strutture di dipendenza tra le variabili, mentre per valori superiori ciò che accade, come discusso in 2, è che *PrivBayes* è costretto a mascherare distribuzioni marginali dall'elevata dimensione, motivo per cui sono molto sensibili al rumore raggiunto, ottenendo di fatto delle distribuzioni marginali eccessivamente rumorose [12].

3.1. Sampling dal dataset completo al variare di ε

Lo studio è stato svolto utilizzando il database aziendale sui clienti. In reguito ad una analisi preliminare su variabili significative e outlier il dataset reale è composto da 21610 osservazioni. Il dataset sintetico è pensato per essere usato solo nella fase di training, ed eseguire il test solo su dati reali, affinchè i risultati ottenuti fossero quanto più possibile realistici. Per questo motivo il training set è composto da 28000 osservazioni, 13000 reali e 15000 sintetiche, e il test set da 8610 osservazioni reali, con un bilanciamento training - test del 76% - 24%.

3.1.1. K = 2

Iniziamo con il presentare i risultati sul test set per $K = 2$. I grafici riportati in seguito mostrano gli score raggiunti dal modello rispetto alle stesse metriche usate nel capitolo 1 (accuratezza, sensibilità, specificità, precisione, valore predittivo negativo). In seguito sono riportati i risultati al variare di ε dal modello con training set composto dai cosiddetti dati misti (unione di dati reali e sintetici), caratterizzati dalle linee continue, confrontati con i risultati acquisiti utilizzando solo i dati reali contraddistinti dalle linee tratteggiate orizzontali. Si ricorda che ε non può assumere valori negativi.

Come si nota dall'immagine 3.1 i dati sintetici non migliorano lo score di accuratezza raggiunto in precedenza in presenza di alcun valore di ε. Tuttavia i risultati più soddifacenti vengono raggiunti in corrispondenza di alti valori del parametro, superiori a 10, al contrario di quanto accade per ε piccolo.

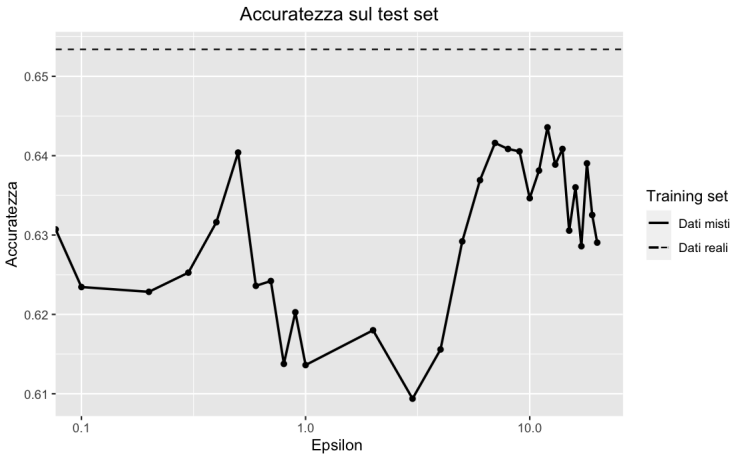

Figura 3.1: Accuratezza del modello al variare di ε, $K = 2$

Per quanto riguarda la sensibilità e la specificità il comportamento sembra essere sim-

metrico per la classe *Sales Interno* rispetto a *Call Center* e *Non gestito*: i primi infatti con l'utilizzo di dati sintetici raggiungono globalmente uno score di sensibilità maggiore, a discapito della specificità e della precisione. Ciò significa che il modello aumenta il numero di elementi classificati correttamente come *Sales Interno*, ma aumentando anche il numero di falsi positivi per questa classe che si riflette nella precisione. Per gli altri due gruppi invece la sensibilità ottenuta non è soddisfacente, ma la specificità e il valore predittivo negativo sono in forte crescita, e anche la precisione risulta molto migliore nel caso di osservazioni *Non gestite*. Da ciò si deduce che il numero assoluto di utenti classificati come *Non gestiti* correttamente diminuisce, conseguentemente al fatto che diminuisce il numero assoluto di utenti classificati come tali, ma è drasticamente calato il numero di falsi positivi, come evidenzia la specificità e la precisione; discorso molto simile per la classe *Call Center*: la precisione ottenuta è confrontabile con il modello di riferimento, mentre il valore predittivo negativo è aumentato insieme alla specificità, il che suggerisce che i falsi positivi siano diminuiti, e in proporzione lo hanno fatto maggiormente di quanto siano aumentati i falsi negativi, evidenziati dalla diminuzione della sensibilità: da ciò si deduce nel complesso che per le classe *Call Center* e *Non gestito* il modello è meno capace di classificare gli utenti positivamente e più abile negativamente.

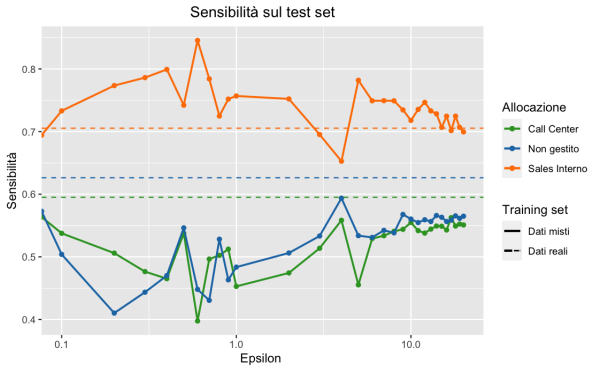

Figura 3.2: Sensibilità del modello al variare di ε, $K = 2$

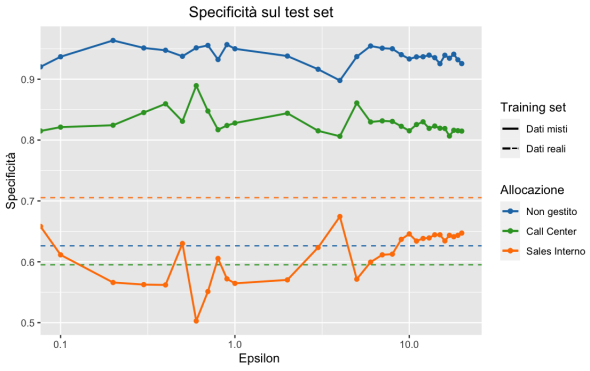

Figura 3.3: Specificità del modello al variare di ε, $K = 2$

3.1.2. K = 3

Si illustrano ora i risultati ottenuti costruendo una rete Bayesiana con un ulteriore grado di profondità rispetto al caso precedente. Da ciò è attendibile che la capacità predittiva del modello sia peggiore nel complesso poichè, da quanto emerso nell'analisi di utilità dei dati svolta nel capitolo 2, i dati sintetici provengono da una distribuzione meno simile a quella dei dati reali rispetto a una rete Bayesiana con due gradi di profondità.

I risultati ottenuti presentano delle analogie rispetto al caso precedente a livello globale,

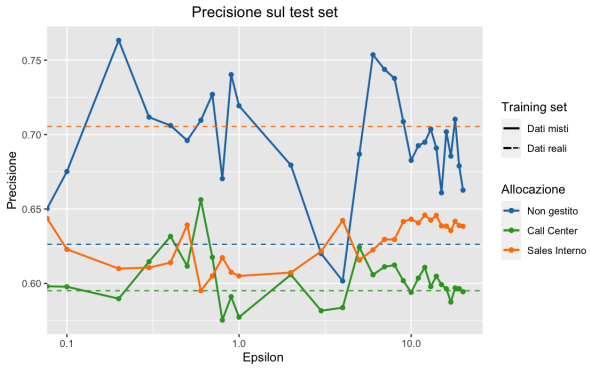

Figura 3.4: Precisione del modello al variare di ε, $K = 2$

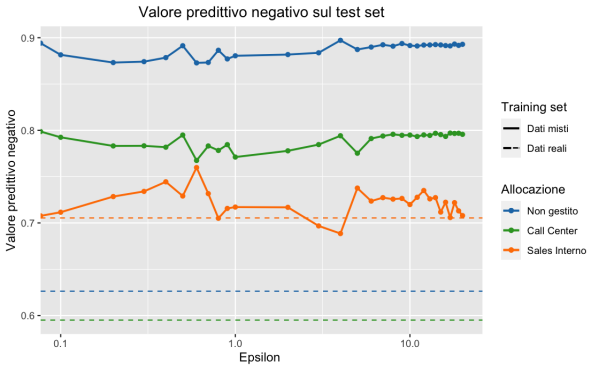

Figura 3.5: Valore predittivo negativo del modello al variare di ε, $K = 2$

mostrando anche alcune diversità. Per quanto riguarda l'accuratezza, anche in questo caso lo score ottenuto con l'utilizzo dei soli dati sintetici non è stato replicato per alcun valore di ε. Sembra più evidente invece che per valori positivi di ε l'accuratezza sia crescente, raggiungendo il massimo in corrispondenza di $\varepsilon = 0$. Ciò significa che tanto più la distanza tra la distribuzione stimata e quella di campionamento si assottiglia, tanto migliore è l'accuratezza sul test set 3.6.

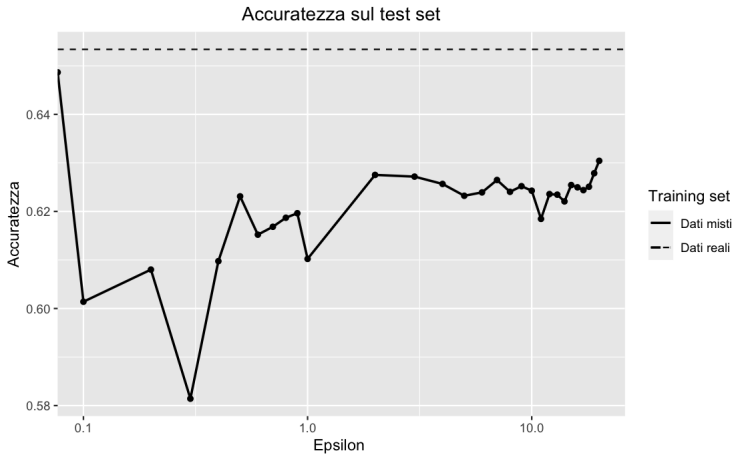

Figura 3.6: Accuratezza del modello al variare di ε, $K = 3$

Ponendo l'attenzione sulle altre metriche, il comportamento speculare tra la classe *Sales Interno* e le classi *Non gestito* e *Call Center* si ripete anche in questo caso con una dinamica che sembra ancora più marcata di quanto riscontrato in precedenza. La sensibilità degli utenti contattati dal reparto sales interno è globalmente migliorata, raggiungendo i picchi intorno all'80% per alti valori di ε, a discapito di un forte ribasso nella specificità e nella precisione. Al contrario, analogamente al caso con $K = 2$, le altre due classi si comportano in maniera opposta per quanto riguarda le prime due metriche (3.7 e 3.8),

producono un netto miglioramento sul valore predittivo negativo (3.10) e gli utenti non gestiti migliorano globalmente lo score di precisione. Inoltre è chiaro come queste tendenze vengano accentuate principalmente per alti valori di ε oppure quando tale parametro è nullo.

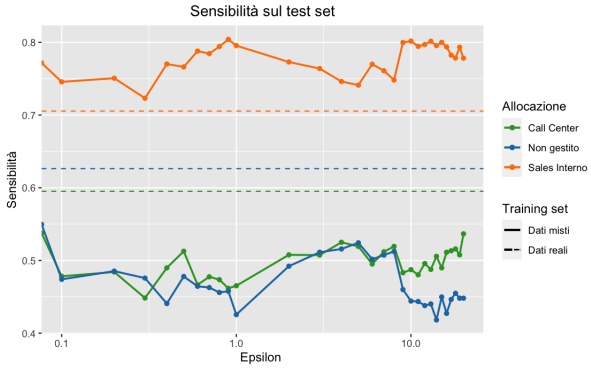

Figura 3.7: Sensibilità del modello al variare di ε, $K = 3$

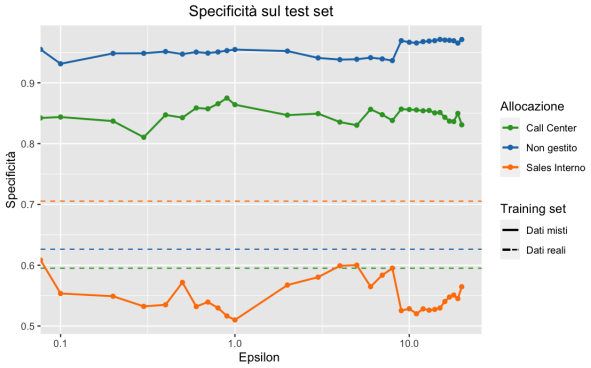

Figura 3.8: Specificità del modello al variare di ε, $K = 3$

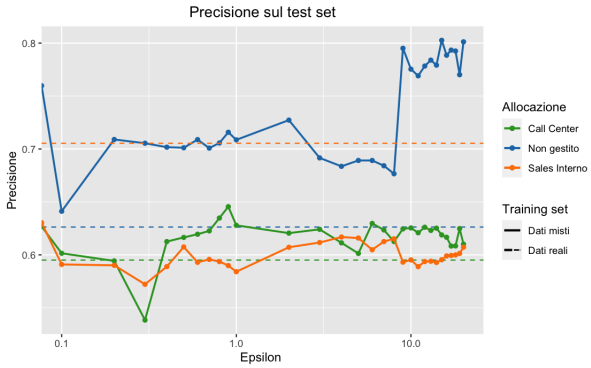

Figura 3.9: Precisione del modello al variare di ε, $K = 3$

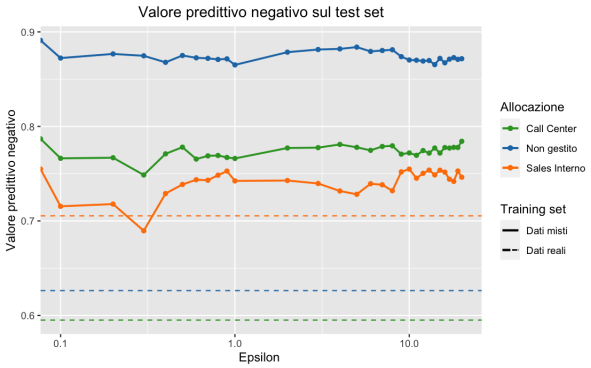

Figura 3.10: Valore predittivo negativo del modello al variare di ε, $K = 3$

In generale, una generazione di dati sintetici che replichi globalmente il database aziendale non ci consente da sola di migliorare le nostre performance predittive sull'accuratezza. Inoltre il modello assume la tendenza di classificare come *Sales Interno* un numero sempre maggiore di clienti a discapito delle rimanenti classi. È necessario dunque agire in modo più specifico sulla categoria *Non gestito*, fornendo al training set del modello un numero maggiore di utenti da questa classe e in modo più variegato, affinchè il classificatore abbia a disposizione un maggior numero di informazioni e sia più allenato nel riconoscimento di osservazioni provenienti da questo gruppo.

3.2. Sampling dal cluster di utenti *Non gestiti* al variare di ε

In questa sezione viene trattato il campionamento degli utenti in modo mirato all'interno del dataset, nel tentativo di replicare e ampliare le conoscenze rispetto alla classe degli utenti *Non gestiti*. L'obiettivo principale è quello di migliorare la capacità predittiva del modello multinomiale rispetto a questa classe senza intaccare eccessivamente le performance rispetto alle altre due categorie. Tale obiettivo permetterebbe all'azienda di ridurre i costi di gestione dei clienti in fase di vendita delle polizze, poichè consentirebbe di risparmiare le risorse aziendali che precedentemente venivano destinate erroneamente a questi prospect, allocandole in maniera più opportuna.

3.2.1. Dati sintetici classificati come *Non gestiti*

Per migliorare l'abilità del modello di classificare utenti non gestiti è stato eseguito un campionamento di 2650 osservazioni dal cluster 2 illustrato nel capitolo 2, usato interamente nel training set, in modo tale da aumentare la proporzione di osservazioni appartenenti a questa classe, passando da circa il 20% al 30%, a discapito principalmente della categoria *Sales Interno*, la cui frazione è scesa dal 45% al 38%. Questo meccanismo non solo aiuta il modello a riconoscere più facilmente i clienti da non ricontattare, ma si allinea a una dinamica aziendale sempre più verosimile con il passare del tempo: da un lato la capacità di Lokky di mantenere i clienti già acquisiti evidenziato dall'alto tasso di rinnovo delle polizze, dall'altro il bacino di nuovi utenti in aumento, costringono l'azienda a vendere sempre più prodotti senza che i clienti siano affiancati da un consulente assicurativo interno, perciò la quantità di clienti che saranno classificati come *Non gestiti* è destinata a crescere nel tempo.

La restante parte del training set è formato solo da osservazioni reali, a causa della non ottimale capacità predittiva in presenza di dati misti descritta nel paragrafo precedente. Dunque complessivamente il training set è formato da 17650 osservazioni e il test set da 6610, con un bilanciamento del 73% - 27%. I dati sono stati generati facendo variare ε sulla stessa griglia di valori del caso di dati sintetici completi, cosi come K che assume i valori 2 e 3.

È possibile notare dalle immagini sottostanti come per $K = 2$ l'accuratezza generale del classificatore rimane inferiore al quella raggiunta con i soli dati reali per ogni valore di ε. Il motivo risiede nell'andamento della sensibilità, che dimostra come effettivamente il modello riconosca in modo più preciso gli utenti *Non gestiti* a discapito della classe

Sales Interno, che essendo la categoria più numerosa ha un peso specifico maggiore sull'accuratezza. Per quanto riguarda gli utenti *Call Center* hanno uno score di sensibilità confrontabile con il classificatore di riferimento. La specficifità dei gruppi *Non gestito* e *Call Center* migliora in modo netto, senza compromettere gli utenti *Sales Interno*, così come il valore predittivo negativo, che dimostra come il modello è migliorato globalmente nel classificare in modo negativo ciascuna classe. Per quanto riguarda la precisione invece i gruppi *Non gestito* e *Call Center* raggiungono la performance del modello con i dati reali, con la seconda delle due classi in leggero miglioramento su ogni valore di ε, a discapito invece dei clienti *Sales Interno*. Da ciò si deduce come il modello abbia affinato la conoscenza generale sulle classi *Non gestito* e *Call Center*, in quanto ad un aumento della sensibilità non è conseguito un peggioramento della precisione; i risultati di specificità e valore predittivo negativo sono entrambi cresciuti, perciò il classificatore è migliorato sia nella classificazione positiva dei due gruppi che in quella negativa. Per quanto riguarda la classe *Sales Interno*, essendo la classe con la frequenza più alta, la diminuzione della sensibilità incide fortemente sull'accuratezza come discusso sopra. La contestuale diminuzione della precisione denota un aumento dei falsi positivi rispetto agli individui classificati come tali, il rapporto tra veri negativi e falsi positivi rimane costante così come quello tra i veri negativi e i falsi negativi. Da ciò si deduce che la diminuzione della sensibilità è dovuta principalmente a una diminuzione generale dei classificati posiviti, e in scala minore ad un aumento dei falsi positivi.

È infine molto importante notare come le performance su tutte le metriche presentano una forte variabilità per bassi valori di ε, mentre si stabilizzano quando ε assume valori maggiori di 10, in corrispondenza dei quali la sensibilità degli utenti *Non gestiti* è crescente nel parametro. Per questi motivi è consigliato scegliere valori di ε intorno al 20.

K=2

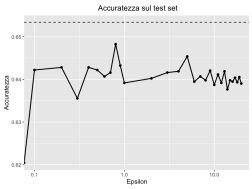

Figura 3.11: Accuratezza

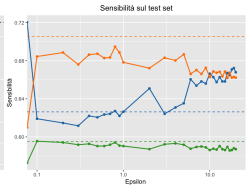

Figura 3.12: Sensibilità

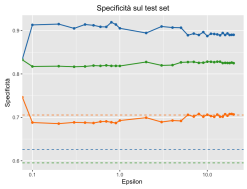

Figura 3.13: Specificità

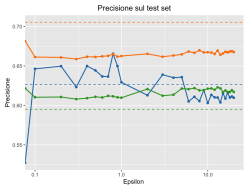

Figura 3.14: Precisione

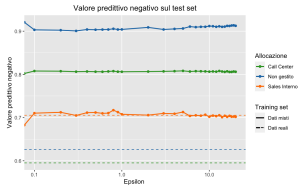

Figura 3.15: Val. predittivo negativo

Il caso di $K = 3$ invece presenta alcune particolarità: era lecito aspettarsi, per l'analisi svolta nel capitolo 2 sulla qualità dei dati sintetici, un rendimento inferiore rispetto a $K = 2$, ma in modo meno evidente di quanto riscontrato nella sezione precedente sui

dati globali, a causa dei DDPlot e dei valori di pMSE ratio, che confrontati con il corrispondente caso di $K = 2$ sono più confortanti sui dati sintetici generati. Inoltre era attendibile che le variazioni nella sensibilità seguissero il comportamento di $K = 2$, con un miglioramento della classe *Non gestiti* a discapito di *Sales Interno*. Invece i grafici mostrati in seguito presentano alcune differenze. In primo luogo è invertito l'andamento della sensibilità tra i *Sales Interno* e le categorie *Non gestito* e *Call Center*, che ha un impatto positivo sull'accuratezza soprattutto per valori di ε superiori a 10, a causa della prevalenza di osservazioni *Sales Interno* nel training set. Inoltre la sostanziale crescita della precisione rispetto al gruppo *Non gestito* denota da un lato la capacità del classificatore di non produrre falsi positivi, ma è dovuta principalmente ad una forte riduzione delle osservazioni totali classificate come positive in questa classe. Questo va in contrasto con l'obiettivo di questa sezione ed è il motivo principale per cui i questi risultati non sono soddisfacenti.

K=3

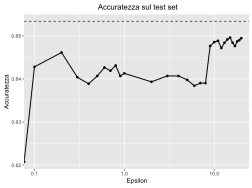

Figura 3.16: Accuratezza

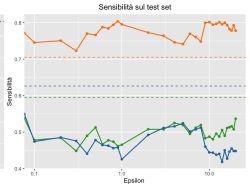

Figura 3.17: Sensibilità

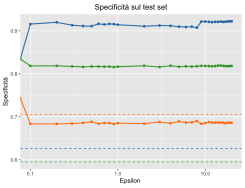

Figura 3.18: Specificità

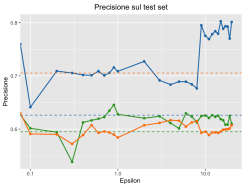

Figura 3.19: Precisione

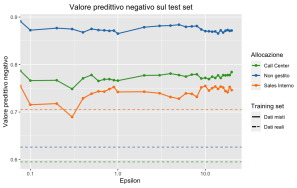

Figura 3.20: Val. predittivo negativo

4 | Conclusioni e futuri sviluppi

4.1. Conclusioni

Il presente elaborato ha descritto il processo di scelta e implementazione di un algoritmo di classificazione dei prospect dell'azienda Lokky, integrato con una fase di miglioramento delle performance predittive attraverso la generazione di dati sintetici, volti ad ampliare il database aziendale riguardo i propri clienti e rendere il classificatore più sensibile nel riconoscimento di alcuni profili commerciali più specifici. I risultati acquisiti durante questa analisi sono i seguenti:

1. Tra i modelli di classificazione considerati, il modello multinomiale logistico con tutti i regressori al primo ordine e i termini di interazione tra variabili numeriche e categoriche, non pesato, ha risposto con le performance migliori in termini di capacità predittive, mostrando buoni risultati anche in termini di aderenza ai dati.

2. La rete Bayesiana usata nell'algoritmo *PrivBayes* per la stima delle indipendenze condizionate deve avere grado 2 perchè con un numero superiore di antenati le distribuzioni congiunte variabile-ascendenti presentano un rapporto tra la scala media di informazione e la scala media di rumore troppo bassa, risultando eccessivamente mascherate dal rumore per qualunque valore di ε. Nel caso in cui $K = 2$, invece, i dati sintetici prodotti approssimano fedelmente la distribuzione congiunta dei dati reali quando $\varepsilon = 0$ o $\varepsilon > 10$.

3. L'accuratezza generale del modello multinomiale ottimale è migliore se il training set è formato solamente da dati reali rispetto a utilizzare un training set misto, composto sia da dati reali che sintetici, per entrambi i valori di K presi in analisi e per ogni valore di ε tra 0 e 20.

4. La sensibilità e la specificità della classe *Non gestito* migliorano congiuntamente se al training del modello multinomiale viene aggiunto un campione sintetico di dati provenienti da questa categoria di clienti, nel caso in cui $K = 2$ e $\varepsilon > 3$. Il parametro ε, se fissato ad un valore di circa 20, mostra una performance ottimale considerando

anche le metriche di precisione e valore predittivo negativo. Questa scelta di K e ε consentono di migliorare complessivamente i risultati sul test set, oltre che per gli utenti *Non gestito*, anche per la categoria *Call Center*, a leggero discapito di *Sales Interno*, in misura adeguata rispetto all'obiettivo di rendere il modello multinomiale logistico più sensibile al riconoscimento dei clienti *Non gestito*.

4.2. Futuri sviluppi

Il progetto in questione si inserisce in una visione aziendale più ampia che, come descritto nell'introduzione, prevede la transizione da un modello di classificazione parametrico come il modello multinomiale logistico, ad un classificatore disegnato con tecniche di machine learning, al fine di migliorare in modo consistente le performance predittive. Inoltre il supporto dell'algoritmo di generazione dei dati sintetici, che rappresenta un'ottimo potenziale nell'arricchimento del database aziendale, potrebbe incidere in modo più significativo sulla qualità del classificatore.

A livello di processo un importante miglioramento risiede nel tracciamento di eventuali errori di classificazione: come descritto nel capitolo 1, la variabile risposta è stata assegnata seguendo la logica di allocazione reale dei lead, ovvero come sono stati gestiti a livello commerciale (il manager del reparto vendite decide nella pratica la modalità di finalizzazione dell'acquisizione del cliente, destinandolo a una gestione interna, al call center, oppure di non affiancarlo ad alcun consulente aziendale). Per quanto riguarda i nuovi utenti invece l'allocazione è l'output del modello multinomiale logistico, che segue la logica di allocazione passata in quanto criterio con il quale è stato costruito il training set. Non è scontato però che tale classificazione sia ottimale, per cui è importante fornire al responsabile del reparto vendite, che è il vero fruitore dell'algoritmo, la possibilità di cambiare la categoria di assegnazione dei nuovi clienti e trasmettere questa informazione, che a tutti gli effetti è un errore di classificazione, al training del classificatore affinchè possa allinearsi con le esigenze di vendita dell'azienda.

Bibliografia

[1] A. Agresti. *Categorical Data Analysis*. John Wiley & Sons, Hoboken, NJ, 3rd edition, 2002.

[2] I. Ben-Gal. Bayesian networks. *Wiley*, page 1, 2008.

[3] D. M. Chickering, D. Heckerman, and C. Meek. Large-sample learning of bayesian networks is np-hard. *J. Mach. Learn. Res.*, 5:1287–1330, dec 2004. ISSN 1532-4435.

[4] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968. doi: 10.1109/TIT.1968.1054142.

[5] M. T. Goodrich and R. Tamassia. *Data structures and Algorithms in Java, 4th edition*. John Wiley & Sons, Hoboken, NJ, 2006.

[6] D. W. Hosmer Jr, S. Lemeshow, and R. X. Sturdivant. *Applied Logistic Regression*. John Wiley & Sons, Hoboken, NJ, 2013. doi: 10.1002/9781118548387.

[7] R. Neapolitan. *Learning Bayesian Networks*. Pearson, 2004.

[8] J. Snoke, G. M. Raab, B. Nowok, C. Dibben, and A. Slavkovic. General and Specific Utility Measures for Synthetic Data. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 181(3):663–688, 03 2018. ISSN 0964-1998. doi: 10.1111/rssa.12358. URL `https://doi.org/10.1111/rssa.12358`.

[9] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. Springer-Verlag, 1993.

[10] A. B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, 2003.

[11] M.-J. Woo, J. P. Reiter, A. Oganian, and A. F. Karr. Global measures of data utility for microdata masked for disclosure limitation. *Journal of Privacy and Confidentiality*, 1(1), Apr. 2009. doi: 10.29012/jpc.v1i1.568. URL `https://journalprivacyconfidentiality.org/index.php/jpc/article/view/568`.

[12] J. Zhang, G. Cormode, C. M. Procopuic, D. Srivastava, and X. Xiao. Privbayes: Private data release via bayesian networks. *Commun. ACM*, 2017.

A | Appendice A

A.1. Metriche utilizzate

Devianza

La devianza di un modello M è una misura della capacità di adattamento ai dati rispetto al modello interpolante $M_{Saturated}$ ciascuna osservazione, che quindi è per definizione il modello ideale in termini di goodness-of-fit. È definita attraverso una specifica trasformazione del rapporto tra la likelihood di M e la likelihood del modello saturato, nel modo seguente:

$$D(M, M_{Saturated}) = -2l(\hat{\boldsymbol{\beta}}) + 2l(\widehat{\boldsymbol{\beta}_{\boldsymbol{Saturated}}}),$$

in cui $l(\hat{\boldsymbol{\beta}})$ è la log-likelihood di M, mentre $l(\widehat{\boldsymbol{\beta}_{\boldsymbol{Saturated}}})$ è la log-likelihood del modello interpolante. Per definizione quest'ultimo è tale da assegnare una probabilità 1 alla classe di appartenenza dell'osservazione, e 0 alle altre classi, perciò ha log-likelihood nulla. Ne segue che la devianza di M si può semplicemente esprimere come:

$$D(M, M_{Saturated}) = -2l(\hat{\boldsymbol{\beta}}) = -2\sum_{i=1}^{n}\sum_{k=0}^{K-1} Y_{ik}\pi_k(\hat{\boldsymbol{\beta}}|\boldsymbol{x_i}).$$

Poichè Y_{ik} vale 1 se l'i-esima osservazione appartiene alla k-esima classe, 0 altrimenti, la devianza assume valori tanto minori tanto più è alta la probabilità stimata associata alla classe corretta. Valori bassi di devianza sono ottimali.

Pseudo R^2

Lo Pseudo R^2 di Cohen è una metrica utilizzata per valutare la bontà di adattamento ai dati nei modelli di regressione logistica. Questa metrica viene usata nei modelli lineari in cui la variabile risposta è categorica, nominale o ordinale, perciò il coefficiente di determinazione R^2 non è definito. L'idea alla base di Pseudo R^2 è quella di fornire

un'interpretazione simile a quella del coefficiente di determinazione R^2 per modelli lineari a risposta continua. Dato il modello multinomiale logistico M, M_{Null} il corrispondente modello nullo e $M_{Saturated}$ il modello interpolante tutte le osservazioni, Pseudo R^2 è definito nel modo seguente:

$$R^2_C = 1 - \frac{D(M, M_{Saturated})}{D(M_{Null}, M_{Saturated})}. \tag{A.1}$$

Tale metrica è compresa tra 0 e 1, dal momento che la devianza nulla è il massimo valore che la devianza di un modello può raggiungere, mentre nel caso di modello interpolante la sua devianza è 0. Il significato che gli è comunemente attributo è che tanto migliore sarà l'aderenza ai dati del modello M, tanto più vicino a 1 sarà Pseudo R^2.

Informazione di Akaike (AIC)

L'Informazione di Akaike è una misura di goodness-of-fit, ottenuta come trasformazione della Devianza in cui viene aggiunto un termine che penalizzi i modelli con un elevato numero di covariate. In particolare si definisce come:

$$AIC = -2l(\hat{\boldsymbol{\beta}}) + 2p,$$

dove p è il numero di regressori del modello. Anche in questo caso, come per la Devianza, sono preferiti bassi valori di AIC.

Sensibilità

La sensibilità, anche nota come True Positive Rate o tasso di veri positivi, rappresenta la proporzione dei veri positivi rilevati dal modello rispetto al totale effettivo di campioni positivi presenti nel sistema in esame. In altre parole, la sensibilità misura la capacità di un classificatore di individuare correttamente gli eventi di interesse.

La sensibilità si calcola mediante la seguente formula:

$$\text{Sensibilità} = \frac{\text{Veri Positivi}}{\text{Veri Positivi} + \text{Falsi Negativi}},$$

dove i Veri Positivi rappresentano il numero di campioni positivi correttamente rilevati dal modello, mentre i Falsi Negativi indicano il numero di campioni positivi non rilevati.

Specificità

La specificità, nota anche come True Negative Rate o tasso di veri negativi, rappresenta la proporzione di campioni negativi correttamente riconosciuti come tali rispetto al totale effettivo di campioni negativi presenti nel sistema in esame. In pratica, la specificità misura la capacità di un classificatore di distinguere correttamente gli elementi che non sono di interesse.

La specificità è definita nel seguente modo:

$$\text{Specificità} = \frac{\text{Veri Negativi}}{\text{Veri Negativi} + \text{Falsi Positivi}},$$

dove i Veri Negativi rappresentano il numero di campioni negativi correttamente rilevati dal modello, mentre i Falsi Negativi indicano il numero di campioni negativi non rilevati.

Precisione

La precisione rappresenta la proporzione delle osservazioni realmente positive tra tutte le osservazioni classificate come tali. In altre parole, misura la capacità di un sistema di restituire esiti accurati tra gli elementi riconosciuti come positivi.

La precisione si calcola come segue:

$$\text{Precisione} = \frac{\text{Veri Positivi}}{\text{Veri Positivi} + \text{Falsi Positivi}}.$$

Valore predittivo negativo

Il valore predittivo negativo rappresenta la proporzione di campioni negativi corretti tra tutte le osservazioni classificate come tali. In altre parole, misura la capacità di un sistema di fornire esiti accurati per gli elementi riconosciuti come negativi.

Il calcolo del valore predittivo negativo è il seguente:

$$\text{Valore predittivo negativo} = \frac{\text{Veri Negativi}}{\text{Veri Negativi} + \text{Falsi Negativi}}.$$

A.2. Dimostrazioni

Il seguente Lemma dimostra che $D_{KL}(P|Q)$ nella definizione 2.17 è sempre ben definita, ovvero che il termine $\int_\Omega \log(\frac{dP}{dQ})\, dP$ esiste. La seguente dimostrazione si trova in [10].

Lemma A.1. *Siano P e Q due misure di probabilità su uno spazio misurabile $(\Omega, \mathcal{F})$. Se $P \ll Q$ allora*

$$\int_\Omega (\log(\frac{dP}{dQ}))_- \, dP \leq V(P,Q) \tag{A.2}$$

dove $a_- = \max\{0, -a\}$ e $V(P,Q)$ è la Total Variation tra P e Q.

Proof. Se $P \ll Q$ abbiamo $\{q > 0\} \supseteq \{p > 0\}, \{pq > 0\} = \{p > 0\}$ Perciò possiamo scrivere

$$\int_\Omega (\log(\frac{dP}{dQ}))_- \, dP = \int_{pq>0} p(\log(\frac{p}{q}))_- \, d\nu \tag{A.3}$$

In cui ν è una misura σ-finita su $(\Omega, \mathcal{F})$ tale che sia P sia Q siano assolutamente continue rispetto a ν (tale misura esiste sempre perché si può prendere ad esempio $\nu = P + Q$). Sia $A_1 = \{q \geq p > 0\}$, abbiamo

$$\int_{pq>0} p(\log(\frac{p}{q}))_- \, d\nu = \int_{A_1} p \log(\frac{q}{p}) \, d\nu \leq \int_{A_1} (q-p) \, d\nu = Q(A_1) - P(A_1) \leq V(P,Q) \tag{A.4}$$

■

Da ciò si deduce che se $P \ll Q$ allora la Divergenza di Kullback può essere scritta come

$$D_{KL}(P|Q) = \int_\Omega p(\log(\frac{p}{q}))_+ \, d\nu - \int_\Omega p(\log(\frac{p}{q}))_- \, d\nu \tag{A.5}$$

in cui il secondo integrale è sempre finito.

Elenco delle figure

Elenco delle tabelle