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

Relazione Finale

Applicazione di modelli di classificazione per la valutazione delle prestazioni nel calcio: gli Expected Goals e gli Expected Points

Relatore Prof.ssa Manuela Cattelan
Dipartimento di Scienze Statistiche

Laureando Luca Varotto
Matricola 2054021

Anno Accademico 2023/2024

Indice

Introduzione

I gol sono da sempre considerati l'aspetto centrale del gioco del calcio, poiché determinano il risultato finale e quindi i giudizi del pubblico e della critica. Per valutare più oggettivamente le prestazioni delle squadre e dei calciatori, si cerca da anni di modellare nel modo più accurato possibile le probabilità di segnare con ogni singolo tiro, al fine di separare la componente di fortuna da quella di merito in ogni gol realizzato e gol mancato.

L'obiettivo di questa relazione è mostrare come l'applicazione di diverse tecniche statistiche possa fornire un'analisi delle prestazioni nel calcio, andando oltre il semplice risultato. In particolare, nel primo capitolo verrà introdotto il concetto di *Expected Goals* o xG, ossia la probabilità che un singolo tiro risulti in gol. Nel secondo verrà presentato il *dataset* utilizzato per stimare gli xG, che include diverse variabili esplicative dei tiri, come la distanza dalla porta, l'angolo di tiro e il tipo di azione che ha preceduto il tiro. Inoltre, qui verranno presentati i tre approcci adottati per bilanciare il numero di successi e insuccessi nel *dataset*. Nel terzo capitolo verranno presentati diversi modelli utilizzabili per stimare gli xG e ne verranno confrontati i risultati sul *dataset* in esame al variare delle tecniche di bilanciamento utilizzate. Nel quarto, verranno utilizzati gli xG stimati dal migliore dei modelli presentati nel precedente capitolo per valutare le occasioni da gol create in una singola partita e per calcolare gli *Expected Points* (xP). Le stime degli xP saranno ottenute con il metodo Monte Carlo.

Capitolo 1

La genesi degli Expected Goals

1.1 La statistica nel calcio

La genesi dell'analisi statistica nel calcio viene comunemente attribuita all'inglese Charles Reep (McMahon, 2012). A Reep, infatti, viene attribuito il merito di aver osservato e annotato tutti gli eventi avvenuti in oltre 2200 partite di calcio nella seconda metà del '900. Le analisi di Reep, tuttavia, non si concentravano sui tiri, ma sulle azioni che precedevano i goal. Dai suoi studi, Reep et al. (1968) concluse che le azioni composte da pochi passaggi in avanti erano quelle da cui era più probabile nascessero dei goal. Le sue deduzioni non risultarono del tutto corrette, ma i suoi metodi e la sua meticolosità nella registrazione degli eventi hanno posto le basi per gli analisti che lo hanno succeduto.

La forte correlazione tra il numero di tiri effettuati e i punti in classifica delle squadre portò appassionati ed esperti a concentrarsi su questo aspetto del gioco; nacque così il *Total Shots Rate* (TSR). Il TSR di una squadra in un partita si trova nel seguente modo:

$$\text{TSR} = \frac{\text{Numero di tiri effettuati}}{\text{Numero di tiri effettuati} + \text{Numero di tiri concessi}}.$$

James Grayson evidenziò la forte correlazione esistente tra il valore medio di questo indice che una squadra registra in una stagione e i punti in classifica ottenuti dalla squadra stessa (Grayson, 2012). Tuttavia il TSR ha un forte limite: a tutti i tiri viene attribuita la stessa importanza, quindi ne viene valutata solo la quantità e non la qualità.

I primi studi riguardo la qualità dei tiri avvennero tra gli anni '80 e gli anni '90 del XX secolo e riguardarono anche Charles Reep, citato in precedenza, come si può vedere in Pollard e Reep (1997). Questi si basarono, inizialmente, nel valutare la qualità dei tiri solo grazie alla distanza dalla porta e all'angolo di tiro. Negli anni successivi, i

nuovi modelli videro l'aggiunta di ulteriori variabili esplicative, ad esempio per distinguere i colpi di testa dai tiri eseguiti con altre parti del corpo (Pollard et al., 2004). Questi modelli raggiunsero i massimi livelli del calcio solo nel XXI secolo, spinti dalla rivoluzione portata da Billy Beane nel baseball americano a cavallo tra il XX e XXI secolo. L'approccio statistico di Beane e dei suo collaboratori è stato reso famoso dal libro *Moneyball* di Lewis (2003), e dall'omonimo film del 2011.

Due figure fondamentali nell'adozione dei modelli di *Expected Goals* ai massimi livelli sono state Matthew Benham e Arsène Wenger. Matthew Benham è presidente del Brentford dal 2012, club che al tempo militava nella terza divisione inglese. Negli ultimi dieci anni il club è diventato famoso per essere riuscito ad arrivare nella prima divisione avendo a disposizione un budget molto inferiore rispetto alle sue concorrenti, ma riuscendo a gestirlo in maniera più efficiente. Il club infatti, si è basato sui dati della società di consulenze *Smartodds*, sempre di proprietà di Benham, per analizzare e valutare i giocatori propri e delle squadre avversarie. Lo scopo di queste analisi era di ottenere una valutazione più oggettiva dei giocatori, in modo da poterne valutare eventuali acquisti e cessioni in maniera più razionale. Tra le varie metriche utilizzate nella fase di valutazione sono presenti gli *Expected Goals* (Tippet, 2019).

Arsène Wenger è stato il manager dell'Arsenal, club di prima divisione inglese, dal 1996 al 2018. I suoi meriti legati all'adozione degli Expected Goals nel calcio ai massimi livelli si possono riassumere in due aspetti. Il primo è tecnico e riguarda l'acquisizione da parte dell'Arsenal dell'azienda StatDNA (Hytner, 2014), una delle società di analisi e raccolta dei dati calcistici più in auge all'epoca. Questo permise al club di sviluppare uno dei modelli di *Expected Goals* maggiormente all'avanguardia. Il secondo aspetto è quello mediatico, in quanto Arsène Wenger è stato il primo manager di un club di massima divisione a parlare apertamente di *Expected Goals*. Ciò avvenne nel 2015 quando dichiarò: *"We analyse after the game the number of chances and the number of expected goals we should score with the chances we create"* (Bate e Campbell, 2015). In seguito affrontò l'argomento altre volte, per esempio a novembre 2017, quando, dopo aver perso una partita contro il Manchester City in cui la sua squadra aveva subito 3 goal realizzandone appena 1, dichiarò: *"If you look at the expected goals, it was 0.7 for them and 0.6 for us, it was a very tight game, they created very little, had very little number of shots on target, one more than us, that's all"* (Gaughan, 2017).

La definizione di *Expected Goals* risulta quindi molto intuitiva, ossia consiste nella probabilità che un singolo tiro risulti in un goal. Arsène Wenger, dopo la sconfitta contro il Manchester City, si riferiva alla versione aggregata degli *Expected Goals* in una singola partita, ossia la somma degli *Expected Goals* di ogni tiro effettuato durante la

partita stessa. Questa misura viene spesso utilizzata per valutare le prestazioni delle squadre in una singola partita, in modo da non limitare i giudizi al solo risultato finale. Tale interpretazione è necessaria perché la fortuna gioca un ruolo fondamentale in uno sport a bassi punteggi come il calcio. Infatti, capita relativamente spesso che le squadre che hanno giocato "meglio" e creato più xG perdano, mentre le squadre che hanno giocato "peggio" e creato meno xG vincano. Questa metrica aiuta quindi a distinguere la componente dovuta alla fortuna e al caso da quella dovuta al merito in ogni singolo tiro.

Capitolo 2

Il dataset

2.1 Raccolta dei dati

La quantità e la qualità dei dati è importante tanto quanto la scelta del modello. In Robberechts e Davis (2020) è stato quantificato che è necessario disporre dei tiri di 5 stagioni complete, da utilizzare per stimare e valutare il modello. Per questa relazione, sono stati quindi utilizzati i dati offerti gratuitamente da *StatsBomb* nella libreria di Python `statsbombpy`, in particolare quelli delle seguenti competizioni complete:

- Premier League, la massima divisione inglese, stagione 2015/2016;
- Bundesliga, la massima divisione tedesca, stagione 2015/2016;
- Ligue 1, la massima divisione francese, stagione 2015/2016;
- La Liga, la massima divisione spagnola, stagione 2015/2016;
- Serie A, la massima divisione italiana, stagione 2015/2016;
- Campionato mondiale di calcio, edizione 2018;
- Campionato mondiale di calcio, edizione 2022;
- Campionato europeo di calcio, edizione 2020;
- Campionato europeo di calcio, edizione 2024;
- Coppa delle nazioni africane, edizione 2023.

Variabile	**Descrizione**
Goal	1 se il tiro è un goal
Azione precedente	Tipo di azione precedente il tiro (9 livelli)
Contesto del tiro	Tipo di azione da cui è arrivato il tiro (4 livelli)
Parte del corpo	Parte del corpo usata per il tiro (4 livelli)
Tecnica di tiro	Stile del tiro effettuato (7 livelli)
Tiro di prima	1 se il tiro è al primo tocco
Contrasto aereo vinto	1 se contrasto aereo vinto prima del tiro
Tiro uno contro uno	1 se il giocatore era solo davanti al portiere
Tiro a porta vuota	1 se il tiro è a porta vuota
Sotto pressione	1 se il giocatore era marcato durante il tiro
Ampiezza	Distanza dalla linea laterale
Profondità	Distanza dalla propria linea di fondo campo
Casa	Luogo della partita: Casa, Trasferta o Campo neutro

Tabella 2.1: Variabili del *dataset* dei tiri

Questi dati sono stati impiegati sia per stimare che per valutare i modelli. In particolare le 9998 osservazioni relative alla stagione 2015/2016 di Serie A sono stati utilizzate per la fase di valutazione, mentre le rimanenti 42794 per la fase di stima.

Per ogni osservazione sono state raccolte le variabili illustrate nella Tabella 2.1. La variabile risposta è **Goal**, ossia una variabile dicotomica che vale 1 quando il tiro risulta in un goal e 0 altrimenti. Le variabili **Contrasto aereo vinto**, **Tiro uno contro uno** e **Tiro a porta vuota** sono state escluse dalle analisi in quanto aventi rispettivamente il 9.7%, 26.2% e 75.5% di valori mancanti. Invece le variabili **Ampiezza** e **Profondità** sono state trasformate, rispettivamente, in valore assoluto della distanza dalla retta passante per il centro delle due porte e distanza dalla linea di fondo avversaria. In questo modo sono state rese simmetriche le previsioni del modello rispetto al centro del campo. Il centro del campo è stato assunto pari alla mediana di **Ampiezza**, mentre la distanza tra le due porte pari al valore massimo di **Profondità**. Nella realtà, ogni campo ha dimensioni leggermente diverse, inoltre la posizione di ogni tiro è registrata manualmente dal personale tecnico di *StatsBomb*: questi due fattori introducono inevitabilmente un termine di errore nei dati. Successivamente tali variabili sono state nuovamente trasformate, tramite le formule trigonometriche, in distanza dal centro porta avversaria (**distanza dalla porta**) e angolo di tiro rispetto al centro della porta avversaria (**angolo di tiro**). La decisione è stata presa in analogia con Rajagopalan e Srid (2023) per aumentare l'interpretabilità dei risultati.

2.2 Analisi esplorativa

Della variabile risposta, ossia **Goal**, sono presenti 38221 insuccessi e 4573 successi nel *dataset* di stima. I successi rappresentano quindi solo il 10.7% delle osservazioni. Ciò può avere delle conseguenze nella fase di stima del modello; questo problema verrà affrontato più nel dettaglio nella sezione 2.3.

Le prime variabili di cui viene studiata la relazione con la variabile risposta sono le posizioni da dove sono stati effettuati i tiri. Individuata l'area di rigore avversaria come il rettangolo nero nella parte destra di ciascuna Figura, possiamo notare che i tiri risultati in goal sono concentrati dentro l'area di rigore e nella fascia più centrale del campo, come si può vedere in Figura 2.1. Al contrario i tiri non risultati in goal sono distribuiti più omogeneamente nella metà campo avversaria, di destra, occupando anche posizioni più esterne e più lontane dalla porta, come si può osservare in Figura 2.2. Si noti che al centro dell'area di rigore, soprattutto nel caso dei tiri risultanti in goal, si ha un picco: è ipotizzabile che ciò sia dovuto ai calci di rigore.

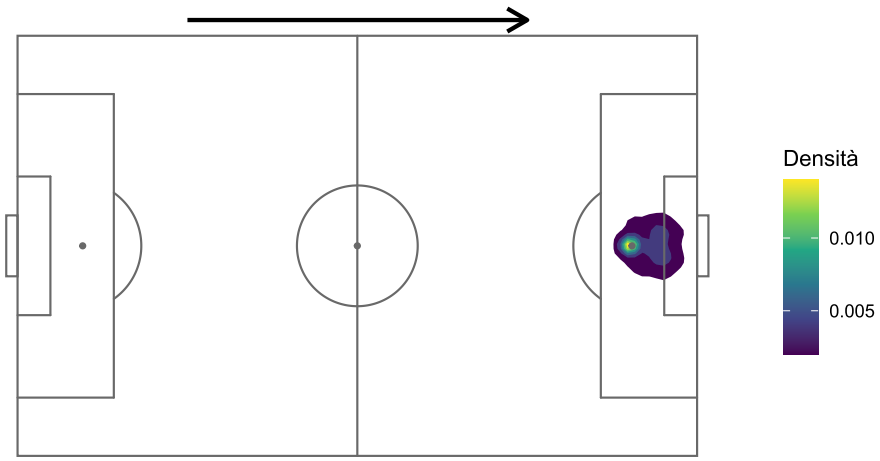

FIGURA 2.1: Densità dei tiri risultanti in goal, stimata con il metodo del nucleo.

Successivamente è stata studiata la distribuzione delle variabili esplicative al variare dei due livelli della risposta. Nel caso di **Azione precedente** osserviamo che tra i tiri risultati in goal le modalità "Contropiedi" e "Altro", hanno un peso maggiore (Figura 2.3). Cercando di approfondire a cosa corrisponde la modalità "Altro" si è notato che per 670 delle 723 osservazioni aventi questo valore è stata rilevata la modalità "Calcio di rigore" per la variabile **Contesto del tiro**.

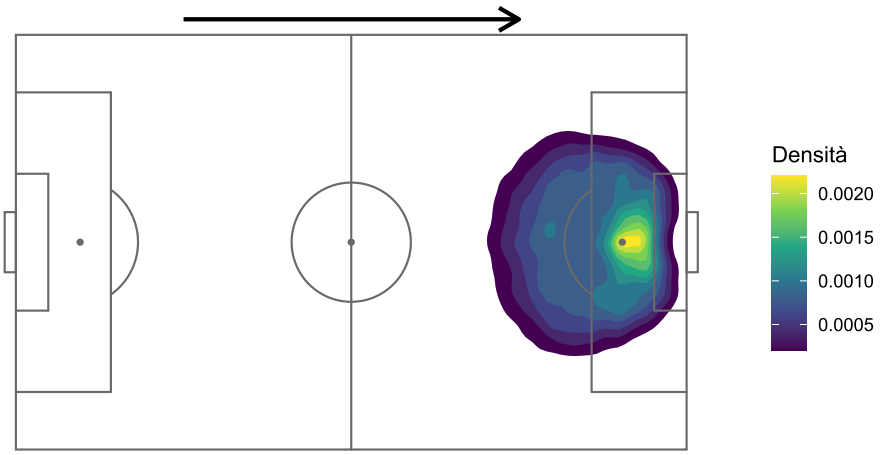

FIGURA 2.2: Densità dei tiri non risultanti in goal, stimata con il metodo del nucleo.

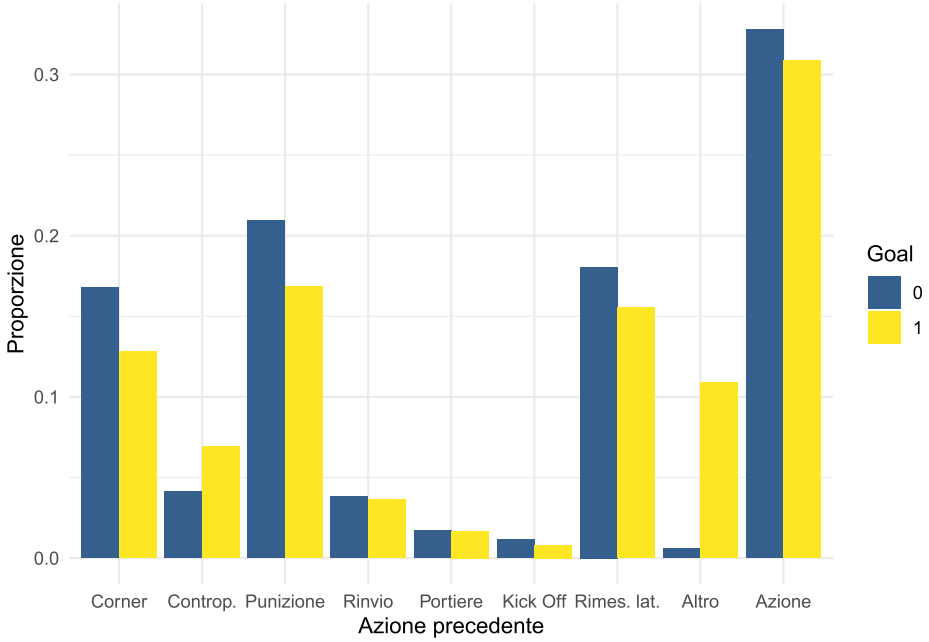

FIGURA 2.3: Grafico a barre della densità di **Azione precedente** per ognuno dei due livelli di goal.

Per la variabile **Contesto del tiro** riusciamo a vedere che nel caso dei goal la proporzione della modalità "Calci di rigore" è maggiore rispetto al caso dei non goal (Figura 2.4). Ciò, assieme a quanto visto in precedenza, ci fa assumere che i calci di rigore abbiano un effetto positivo sulla probabilità di segnare.

La variabile **Parte del corpo** graficamente non varia molto al variare dei due livelli

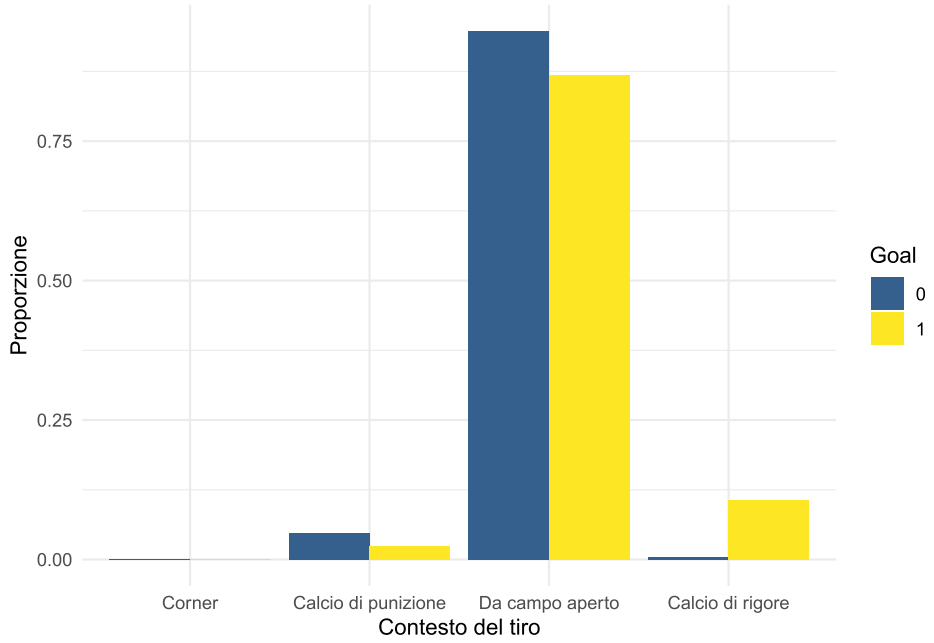

FIGURA 2.4: Grafico a barre della densità di **Contesto del tiro** per ognuno dei due livelli di goal.

di **goal** (Figura 2.5).

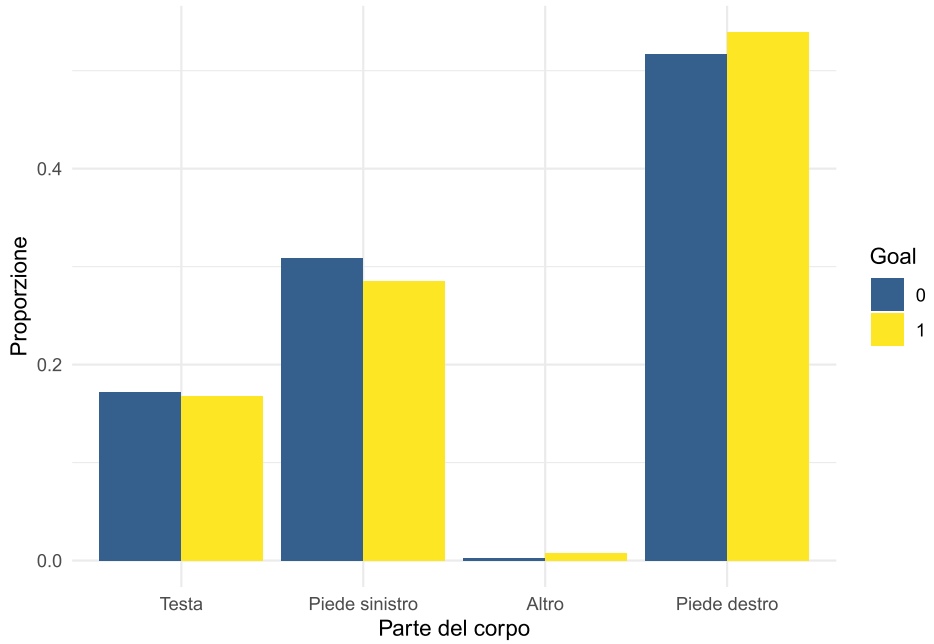

FIGURA 2.5: Grafico a barre della densità di **Parte del corpo** per ognuno dei due livelli di goal.

Per quando riguarda la variabile **Tiro di prima**, tra i goal si può notare che aumenta la proporzione della modalità "Sì" (Figura 2.6).

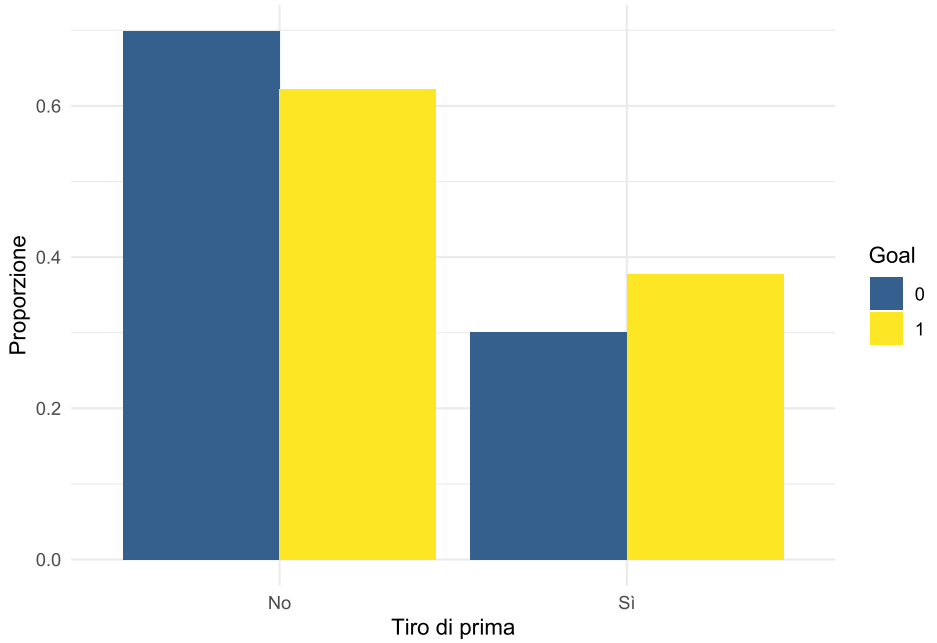

FIGURA 2.6: Grafico a barre della densità di **Tiro di prima** per ognuno dei due livelli di goal.

Graficamente sia la variabile **Tecnica di tiro** (Figura 2.7) che la variabile **Sotto pressione** (Figura 2.8) risultano avere dei cambiamenti minimi al variare della variabile risposta.

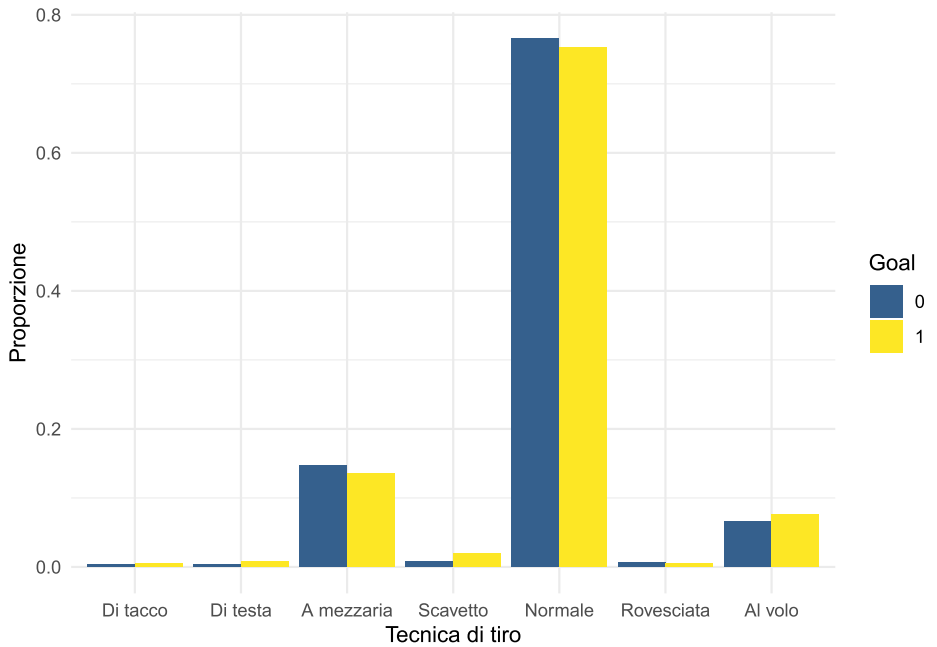

FIGURA 2.7: Grafico a barre della densità di **Tecnica di tiro** per ognuno dei due livelli di goal.

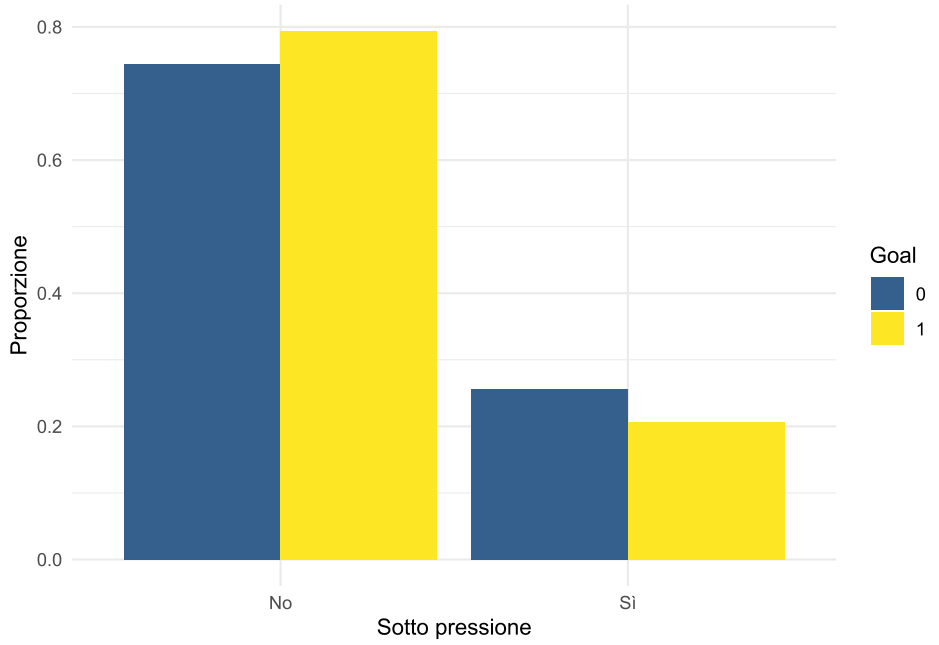

FIGURA 2.8: Grafico a barre della densità di **Sotto pressione** per ognuno dei due livelli di goal.

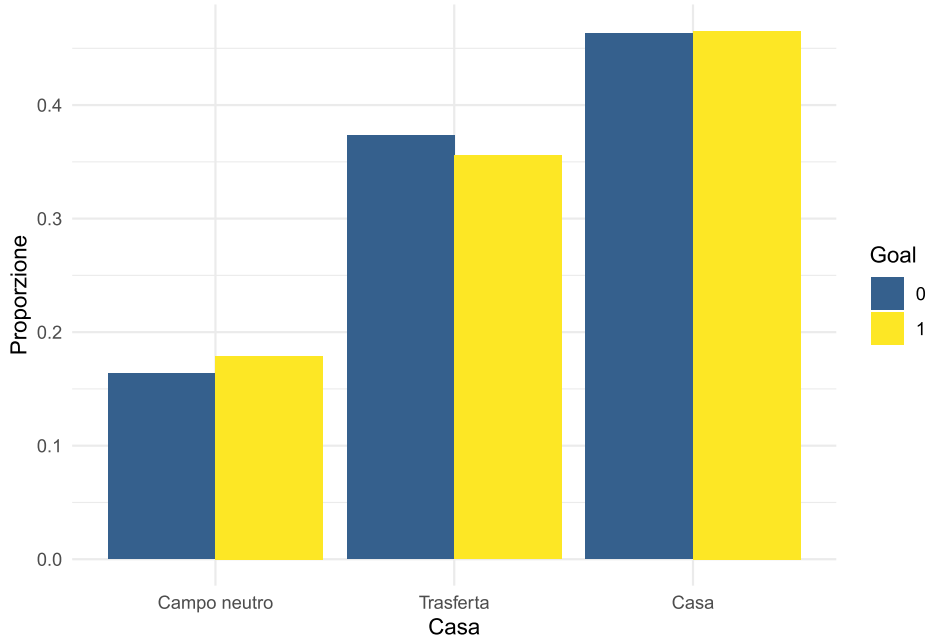

FIGURA 2.9: Grafico a barre della densità di **Casa** per ognuno dei due livelli di goal.

La variabile **Casa** ha tre livelli: "Casa" per le squadre che giocano in casa, "Trasferta" per quelle in trasferta, e "Campo Neutro" per le partite giocate su campo neutro. Si noti che nei tornei internazionali giocati tra il 2018 e il 2024, tutte le partite si sono giocate su campo neutro, quindi per queste partite il valore della variabile è "Campo Neutro" per entrambe le squadre. Graficamente (Figura 2.9) possiamo notare che la

modalità "Campo Neutro" ha una maggior peso nel caso dei goal rispetto al caso dei non goal. Al contrario la modalità "Trasferta" ha un peso minore nel caso dei goal rispetto al caso dei non goal.

2.3 Il bilanciamento del dataset

Come menzionato in precedenza solo il 10.7% delle osservazioni rappresentano dei successi. Questo può comportare dei problemi nella fase di stima dei vari modelli. Infatti, spesso capita che i modelli stimati con questo tipo di *dataset* abbiano un'elevata accuratezza (per la definizione si veda la sezione 3.1), ma una scarsa sensibilità (per la definizione si veda la sezione 3.1) (Wang et al., 2021).

Esistono diversi approcci per risolvere questo problema, in questo caso ne verranno adottati tre che agiscono direttamente sui dati. In particolare verranno confrontati i risultati delle seguenti procedure:

Undersampling con un approccio casuale, ossia la classe maggioritaria viene ridotta di numerosità estraendone un campione in maniera pseudo-casuale (Shelke et al., 2017). In questo caso sono stati estratti 4573 non goal, in modo da avere lo stesso numero di successi ed insuccessi. Tale procedura è stata eseguita in R con la versione 0.0-4 della libreria `ROSE` (Lunardon et al., 2022).

Oversampling con un approccio casuale, ossia la classe minoritaria viene ri-campionata pseudo-casualmente (Shelke et al., 2017). In questo caso sono stati ri-campionati i tiri risultati in goal, fino ad averne 38221, in modo da pareggiare il numero di successi ed insuccessi. Tale procedura è stata eseguita in R con la versione 0.0-4 della libreria `ROSE` (Lunardon et al., 2022).

SMOTE-NC ossia *Synthetic Minority Oversampling Technique - Nominal Continuous*. Come si può intuire dal nome questo è un caso particolare di *Oversampling*, la differenza rispetto a quanto visto al punto precedente è che le nuove osservazioni della classe minoritaria, quindi dei goal, non sono generate casualmente tramite ripescaggio, ma con un procedura che verrà descritta nella sezione 2.3.1. Tale procedura è stata eseguita in Python con la versione 0.12.3 della libreria `imblearn.over_sampling`.

2.3.1 SMOTE-NC

Lo SMOTE è una procedura di bilanciamento del *dataset* basata sulla generazione di osservazioni "sintetiche". Queste nuove osservazioni sono dette "sintetiche" perché non provengono da dati reali raccolti, come nel caso dell'*Oversampling* casuale, ma sono generate da un algoritmo.

In particolare lo SMOTE-NC è una variante della SMOTE proposta in Chawla et al. (2002). Questa è stata scelta poiché permette di generare sinteticamente delle osservazioni in presenza sia di variabili quantitative che categoriali, come nel caso in esame. Il funzionamento dell'algoritmo può essere diviso in due fasi: una prima fase di calcolo delle distanze tra le osservazioni già presenti e una seconda fase di generazione delle nuove osservazioni. Il calcolo delle distanze avviene attraverso i seguenti due passaggi:

1. Si trova la mediana della deviazione standard di tutte le variabili quantitative, questa quantità verrà detta Med.

2. La vicinanza tra due osservazioni, composte da p variabili quantitative e q variabili categoriali ciascuna, viene determinata con la distanza euclidea. Quindi dati due vettori $x_1 = (x_{11}, \ldots, x_{1p}, x_{1(p+1)}, x_{1(p+q)})$ e $x_2 = (x_{21}, \ldots, x_{2p}, x_{2(p+1)}, x_{2(p+q)})$, $(p+q)$-dimensionali, la distanza euclidea sarà data da:

$$d(x_1, x_2) = \sqrt{\sum_{k=1}^{p+q} (x_{1k} - x_{2k})^2}. \tag{2.1}$$

   Nel calcolarla però, verrà eventualmente sommata la quantità $(Med)^2$, all'interno della radice quadrata nella Formula 2.1, tante volte quante sono le variabili categoriali, tra le q presenti, che differiscono tra le osservazioni x_1 e x_2. La formula finale per il calcolo della distanza sarà quindi:

$$d(x_1, x_2) = \sqrt{\sum_{k=1}^{p} (x_{1k} - x_{2k})^2 + \sum_{j=p+1}^{p+q} \mathbb{I}(x_{1j}, x_{2j}) \cdot (Med)^2}, \tag{2.2}$$

   dove $\mathbb{I}(x_{1j}, x_{2j})$ è una funzione indicatrice che vale 1 se $x_{1j} \neq x_{2j}$ e 0 altrimenti.

Questi due passaggi vanno ripetuti per tutte le osservazioni. La quantità Med viene inclusa per penalizzare le differenze nello spazio delle variabili categoriali di una quantità che è legata alla differenza tipica nello spazio delle variabili continue.

In Tabella 2.2 viene riportato un esempio del calcolo della distanza tra due osservazioni x_1 e x_2. In questo caso la variabile x_{i4} è uguale tra le due osservazioni, mentre le

	Variabili continue			Variabili categoriali		
Osservazione	x_{i1}	x_{i2}	x_{i3}	x_{i4}	x_{i5}	x_{i6}
x_1	1	2	3	A	B	C
x_2	4	6	5	A	D	E

TABELLA 2.2: Esempio introduttivo allo SMOTE tratto da Chawla et al. (2002).

variabili x_{i5} e x_{i6} differiscono. La distanza tra x_1 e x_2 sarà quindi data da:

$$d(x_1, x_2) = \sqrt{(4-1)^2 + (6-2)^2 + (5-3)^2 + Med^2 + Med^2}$$

dove Med è la mediana delle deviazioni standard delle variabili x_{i1}, x_{i2} e x_{i3}, calcolata su tutto il dataset.

Una volta calcolate le distanze, la generazione di una nuova osservazione "sintetica", w^*, avviene nei seguenti passaggi:

1. Dalla classe minoritaria viene scelta casualmente un'osservazione, che chiameremo x^*, le cui variabili quantitative sono un vettore p-dimensionale che chiameremo p_{x^*}. Vengono individuati i *k-Nearest Neighbors* (*k-NN*) tra la classe minoritaria, ossia le k osservazioni della classe minoritaria più vicine a x^*. La distanza viene calcolata con il metodo descritto in precedenza.

2. Viene estratta casualmente un'osservazione tra le *k-NN* di x^*, che chiameremo z^*. Le variabili quantitative di z^* sono un vettore p-dimensionale detto p_{z^*}. I valori delle variabili quantitative di w^* sono generati lungo il segmento che unisce x^* e z^*. Quindi si trovano con la seguente formula:

$$p_{x^*} + (p_{x^*} - p_{z^*}) \cdot b,$$

   dove b è un numero pseudo-casuale generato tra 0 e 1. Questa operazione viene ripetuta tante volte quante osservazioni è necessario generare. Quindi, se per esempio è necessario triplicare la numerosità della classe minoritaria verrà ripetuta per tre volte la selezione casuale di z^* dai *k-NN* di x^*.

3. I valori delle variabili categoriali sono decisi basandosi sull'algoritmo *k-NN*. Tuttavia nel calcolo della distanza tra le osservazioni in questo caso non verrà considerata la quantità Med, in quanto i valori delle variabili categoriali sono ignoti.

La scelta dell'iper-parametro k dell'algoritmo *k-NN* è stata eseguita tramite una procedura di convalida incrociata sul modello di regressione logistica con tutte le variabili

esplicative incluse nel modello. Per i valori di k sono stati testati tutti i numeri dispari compresi tra 1 e 15, estremi inclusi. Sono stati ottenuti i migliori risultati in termini di F-score (per la definizione si veda la sezione 3.1) in corrispondenza di $k = 3$. I risultati arrotondati alla quarta cifra decimale sono riportati in Tabella 2.3.

k	1	3	5	7	9	11	13	15
F-score	0.7470	0.7471	0.7470	0.7469	0.7470	0.7470	0.7471	0.7468

TABELLA 2.3: Valore dell'indice F-score in cross-validation al variare dell'iper-parametro k.

Inoltre è stato verificato che la procedura non è influenzata dalla scala della variabili quantitative. Questo è stato fatto applicando la procedura di convalida incrociata per l'ottimizzazione dell'iper-parametro k e successivamente confrontando i risultati sul *dataset* di valutazione.

Capitolo 3

Scelta del modello

3.1 I criteri di valutazione

I migliori modelli di *Expected Goals* sono stimati e utilizzati dalle società calcistiche e dalle principali agenzie di scommesse. Tali enti sono interessati a non rendere pubblico il funzionamento dei loro modelli, al fine di mantenere un vantaggio competitivo rispetto ai concorrenti. A parte qualche rara eccezione, come Matteotti e Sotudeh (2024), non siamo quindi sicuri del funzionamento dei modelli sviluppati da questi enti, possiamo però affidarci alla letteratura scientifica per valutare diversi approcci possibili e individuarne il più efficace.

		Attuali	
		Insucesso (No goal)	Successo (goal)
Predetti	Insucesso (No goal)	Veri Negativi (VN)	Falsi Negativi (FN)
	Successo (goal)	Falsi Positivi (FP)	Veri Positivi (VP)

Tabella 3.1: Matrice di confusione

In questo capitolo i modelli utilizzati verranno presentati e valutati tramite diverse metriche, ricavate dalla matrice di confusione in Tabella 3.1, ossia:

$$\hat{\alpha} = \frac{FP}{VN + FP}, \quad \hat{\beta} = \frac{FN}{FN + VP}$$

$$\text{Specificità} = 1 - \hat{\alpha} = \frac{VN}{VN + FP}, \quad \text{Sensibilità} = 1 - \hat{\beta} = \frac{VP}{VP + FN}$$

$$\text{Precisione} = \frac{VP}{VP + FP}, \quad \text{F-score} = \frac{2 \cdot \text{Precisione} \cdot \text{Sensibilità}}{\text{Precisione} + \text{Sensibilità}},$$

$$\text{Accuratezza} = \frac{VP + VN}{VN + FN + FP + VP}.$$

Tutte le metriche appena introdotte si possono calcolare solo dopo aver fissato una soglia di separazione tra le classi. Tale soglia, compresa tra 0 e 1, stabilisce che un tiro viene considerato un goal se l'*Expected Goal* stimato è superiore alla soglia stessa, altrimenti viene ritenuto non goal.

L'ultima metrica considerata è l'AUC, ossia l'area sottostante la curva ROC (*receiver operating characteristic*). Dove la curva ROC è definita come l'insieme dei punti $(\hat{\alpha}, 1 - \hat{\beta})$, al variare della soglia di separazione tra le classi. Questa metrica è particolarmente utile perché, a differenza delle altre, può essere calcolata senza dover fissare un valore della soglia. Per questa caratteristica la curva ROC è stata usata anche nella scelta della soglia di separazione, come approfondito nella sezione 3.2.

Per ogni modello sono state stimate quattro versioni: una sul *dataset* sbilanciato, un'altra sul *dataset* bilanciato con la procedura di *undersampling* pseudo-casuale, una terza sul *dataset* bilanciato con la procedura di *oversampling* pseudo-casuale e l'ultima sul *dataset* bilanciato con la procedura di SMOTE-NC. In questa capitolo, per ogni modello verranno prima confrontate le quattro versione stimate e poi confrontate le migliori versioni di ogni modello in termini di F-score sul *dataset* di valutazione. Si è scelto di valutare i modelli rispetto all'F-score poiché, per la natura del problema, risulta più difficile ottenere un modello che classifichi correttamente i goal rispetto ad uno che classifichi correttamente i non goal.

3.2 La scelta della soglia

La scelta della soglia è un aspetto chiave dei problemi di classificazione, in quanto può cambiare radicalmente i risultati in termini di accuratezza, precisione e sensibilità (Freeman e Moisen, 2008). Più il valore della soglia è basso infatti, più osservazioni verranno stimate come successi, quindi più aumenterà la sensibilità. Al contrario più il valore della soglia è alto, più osservazioni verranno stimate come insuccessi, aumentando quindi il valore della specificità (Salvan et al., 2020).

I criteri per selezionare la soglia ottimale sono diversi, come mostrato per esempio in Zou et al. (2016). Nel caso in esame si è scelto un criterio presentato in Perkins e Schisterman (2006). Gli autori spiegano che l'angolo in alto a sinistra del grafico della curva ROC, ossia il punto $(0, 1) \in \mathbb{R}^2$, rappresenta il modello ideale, in quanto in tal punto sia la specificità che la sensibilità valgono 1, caso in cui il modello prevede correttamente tutte le osservazioni. Definendo la specificità e la sensibilità come due funzioni della soglia, ossia $p(c)$ e $q(c)$, dove c è la soglia, sceglieremo come soglia il punto

c^* che minimizza la distanza euclidea tra $(p(c), q(c))$ e il loro valore ideale, ossia (1,1). Sceglieremo quindi il valore c^* che risolve l'equazione:

$$\min_{c^* \in (0,1)} \left(\sqrt{[1 - q(c^*)]^2 + [1 - p(c^*)]^2} \right). \tag{3.1}$$

Questo primo approccio tuttavia non è particolarmente adatto a *dataset* sbilanciati. Infatti in questi casi la curva ROC può essere fuorviante e può portare a non scegliere un modello con alta accuratezza (Zou et al., 2016). Per risolvere questo problema nello stesso articolo è stata proposta la seguente correzione dell'equazione (3.1):

$$\min_{c^* \in (0,1)} \left(\sqrt{[1 - q(c^*)]^2 + r * [1 - p(c^*)]^2} \right), \text{ con } r = \frac{1 - \bar{y}}{\gamma \bar{y}}, \tag{3.2}$$

dove $\bar{y}$ è la frazione di successi sul totale della popolazione, mentre γ rappresenta il costo relativo di un falso negativo rispetto ad un falso positivo. Nel caso in esame γ è stato posto pari a 1, in tal modo il costo di un falso positivo è uguale al costo di un falso negativo. Mentre il valore $\bar{y}$ è stato posto pari alla proporzione campionaria di goal nel *dataset* di valutazione, visto che questa procedura è stata eseguita in quel *dataset*.

Il fatto di aver scelto la soglia sul *dataset* di valutazione è uno degli elementi critici di questa procedura. Infatti, questo può aver aumentato il rischio di *overfitting*, portando quindi a scegliere un modello che si adatti bene ai dati in esame, ma non si adatti altrettanto bene a ulteriori dati di stagione future. Sono state valutate due alternative, ma sono state scartate per i seguenti due motivi:

1. La soglia non è stata scelta con una procedura di convalida incrociata su ogni singolo modello poiché tale tecnica è stata utilizzata per diverse altre procedure. Utilizzarla anche per ottimizzare questo iper-parametro avrebbe aumentato comunque il rischio di over-fitting;

2. L'altra alternativa valutata è stata la creazione di un terzo *dataset*, di validazione, contenente circa il 20% delle osservazioni disponibili. In tal modo, per ogni modello, le procedure di stima, di scelta della soglia e di valutazione sarebbero avvenute su tre *dataset* diversi, riducendo il rischio di *overfitting*. Tuttavia, come evidenziato nella sezione 2.1, per stimare e testare questa tipologia di modelli è consigliabile disporre dei dati di 5 stagioni complete, quindi si è deciso di non sacrificare il 20% dei dati, ossia circa un'intera stagione, nel creare un nuovo dataset.

3.3 Regressione logistica

Uno dei modelli più frequentemente utilizzati per stimare gli *Expected Goals* è il modello di regressione logistica (Pollard et al., 2004). Sotto questo modello la realizzazione del singolo tiro, y_i, si assume sia realizzazione di $Y_i \sim Bi(1, \pi_i)$, con

$$\pi_i = \frac{\exp(\sum_{r=0}^{p} \beta_r x_{ir})}{1 + \exp(\sum_{r=0}^{p} \beta_r x_{ir})} \quad \text{e quindi: } \operatorname{logit}(\pi_i) = \log\left(\frac{\pi_i}{1-\pi_i}\right) = \sum_{r=0}^{p} \beta_r x_{ir} = \eta_i.$$

Per stimare questo modello è necessario assumere l'indipendenza delle osservazioni, quindi dei vari tiri. Questo assunto non è completamente soddisfatto nel contesto di una partita di calcio, tuttavia esso è fondamentale per semplificare la complessità del problema e per le successive applicazioni del modello stesso.

Il modello di base che possiamo stimare con questo approccio calcola la probabilità di segnare solo in funzione della distanza dalla porta e dell'angolo di tiro. L'implementazione è avvenuta in R con la versione 4.3.3 della libreria `stats`, i risultati dei modelli stimati con i quattro *dataset* sono riportati in Tabella 3.2.

	Stime dei parametri con i diversi dataset			
Parametro	**Sbilanciati**	**Undersampled**	**Oversampled**	**SMOTE-NC**
Intercetta	-0.897	1.169	1.213	1.278
Distanza di tiro	-0.128	-0.128	-0.127	-0.134
Angolo di tiro	0.013	0.014	0.013	0.014

TABELLA 3.2: Stime dei parametri della regressione logistica con solo distanza e angolo di tiro al variare del *dataset* di stima.

Un vantaggio di questo modello è che, grazie alla sua semplicità, risulta facilmente interpretabile. Infatti, definita la quota come il rapporto tra la probabilità di successo e quella di insuccesso, possiamo affermare che un aumento unitario di una variabile esplicativa, mantenendo costante l'altra, provoca una variazione del rapporto di quote di un termine pari all'esponenziale della stima del coefficiente associato alla variabile non rimasta costante. Nel caso in esame, per ogni metro in più che ci si allontana dalla porta, la quota diminuisce di $exp(-0.128) = 0.880$ volte, ovvero di circa il 12.0% nel modello stimato sui dati sbilanciati. Nei modelli stimati sui dati bilanciati mediante *Undersampling* pseudo-casuale, *Oversampling* pseudo-casuale e SMOTE-NC, il calo della quota è rispettivamente del 12.0%, 11.9% e 12.5% per ogni aumento unitario della distanza dalla porta. L'effetto dell'angolo di tiro indica invece che per ogni grado in più, quindi avvicinandosi alla zona centrale del campo, la quota aumenta dell'1.3% nel modello stimato

sui dati sbilanciati. Negli altri casi, l'aumento della quota è pari all'1.3% sia nel modello stimato sui dati bilanciati con *Undersampling* pseudo-casuale che in quello stimato sui dati bilanciati con *Oversampling* pseudo-casuale, mentre è pari all'1.4% nel modello stimato sui dati bilanciati con SMOTE-NC. Graficamente, le probabilità di segnare sono rappresentate nei quattro casi nelle Figure dalla 3.1 alla 3.4. Individuata l'area di rigore avversaria come il rettangolo bianco nella parte destra di ciascuna Figura, si può notare come al di fuori da essa, secondo tutti i modelli le probabilità di segnare sono quasi nulle, mentre al suo interno le conclusioni differiscono. Nel modello stimato sui dati sbilanciati, le probabilità di segnare all'interno dell'area di rigore risultano generalmente più basse rispetto ai tre modelli stimati sui dati bilanciati. Questa differenza è principalmente attribuibile all'intercetta, infatti nel caso in cui **Distanza di tiro** e **Angolo di tiro** siano pari a 0, la probabilità di segnare è pari a $logit(\beta_0)$, dove β_0 rappresenta l'intercetta. Le probabilità stimate in questo caso saranno rispettivamente pari a 0.290, 0.763, 0.771 e 0.782. Sebbene un tiro da tale posizione sia poco realistico a livello calcistico, esso risulta fondamentale per la natura additiva del predittore lineare η_i.

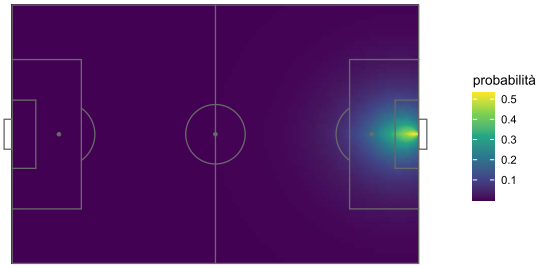

FIGURA 3.1: Grafico delle probabilità stimate sulla base del modello con solo distanza e angolo di tiro stimato sui dati sbilanciati.

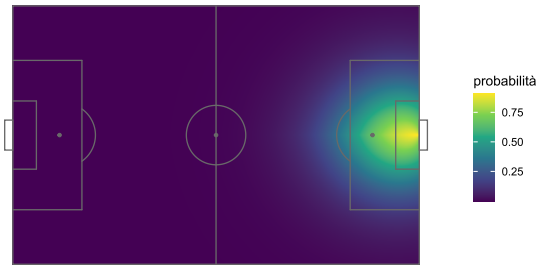

FIGURA 3.2: Grafico delle probabilità stimate sulla base del modello con solo distanza e angolo di tiro stimato sui dati bilanciati con *undersampling* pseudo-casuale.

Le metriche di valutazione dei quattro modelli sono riportate in Tabella 3.3, dove in grassetto sono evidenziati i migliori valori per ogni metrica. Si può notare che nel caso del modello stimato sul *dataset* sbilanciato è stata individuata una soglia più bassa rispetto agli altri casi. I modelli stimati sui *dataset* bilanciati invece hanno dei valori della soglia più simili tra di loro, in particolare questi sono compresi tra 0.670 e 0.700. Nonostante queste differenze nella soglia, i modelli sono analoghi per quanto riguarda AUC e F-score approssimati alla terza cifra dopo la virgola.

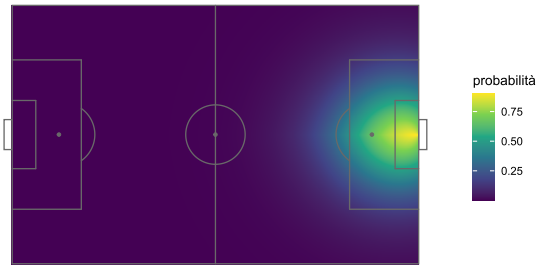

FIGURA 3.3: Grafico delle probabilità stimate sulla base del modello con solo distanza e angolo di tiro stimato sui dati bilanciati con *oversampling* pseudo-casuale.

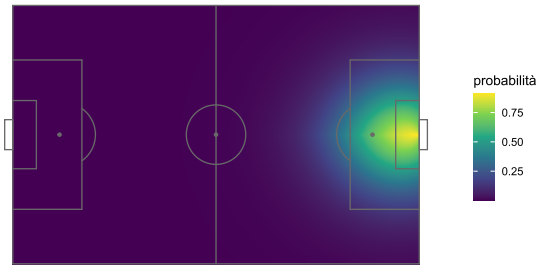

FIGURA 3.4: Grafico delle probabilità stimate sulla base del modello con solo distanza e angolo di tiro stimato sui dati bilanciati con SMOTE-NC.

Training set	AUC	Soglia	F-score	$1-\hat{\alpha}$	$1-\hat{\beta}$	Accuratezza
Sbilanciato	0.774	0.213	0.360	0.892	**0.445**	0.850
Undersampled	**0.774**	0.670	0.360	**0.895**	0.439	**0.852**
Oversampled	0.774	0.692	0.360	0.892	**0.445**	0.850
SMOTE-NC	0.774	0.700	**0.360**	0.893	**0.445**	0.850

TABELLA 3.3: Tabella delle metriche dei modelli di regressione logistica con solo distanza e angolo di tiro registrate sul *dataset* di valutazione al variare dei *dataset* di stima.

Successivamente è stata adottata una procedura *forward*, basata sull'*Akaike's information criterion* (AIC), per valutare l'aggiunta di ulteriori variabili esplicative nel modello. L'AIC di un modello M_d, dove d è il numero di parametri, si calcola come:

$$AIC(M_d) = 2 \cdot d - 2 \cdot l(\hat{\theta}^{(d)}; y),$$

dove $l(\hat{\theta}^{(d)}; y)$ è la log-verosimiglianza calcolata nella stima di massima verosimiglianza del modello M_d. La procedura *forward* così eseguita porterà ad aggiungere nuove covariate solo nel caso in cui la loro aggiunta provochi un abbassamento dell'AIC, e quindi un miglioramento della bontà di adattamento, oltre il termine di penalizzazione, pari a $2d$.

Questo modello è stato implementato in R con la versione 4.3.3 della libreria `stats`. In Tabella 3.4 sono riportate le stime dei parametri per ogni modello stimato, le linee tratteggiate servono a separare i parametri relativi alla stessa variabile esplicativa. Dalla stessa Tabella possiamo vedere che la procedura *forward* ha portato all'aggiunta di tutte

le variabili esplicative in ciascun *dataset* stimato.

	Stime dei parametri con i diversi dataset			
Parametro	**Sbilanciati**	**Unders.**[1]	**Overs.**[2]	**SMOTE-NC**
Intercetta	2.613	4.166	4.662	1.643
Azione precedente:				
Contropiede	0.948	0.935	0.952	0.851
Punizione	0.329	0.218	0.316	0.339
Rinvio dal fondo	0.477	0.449	0.444	0.085
Dal portiere	0.629	0.530	0.648	-0.267
Calcio d'inizio	0.353	-0.213	0.332	-1.206
Altro	0.419	0.433	0.409	-1.629
Rimessa laterale	0.470	0.358	0.450	0.436
Azione regolare	0.551	0.592	0.112	0.503
Tecnica di tiro:				
Di testa in tuffo	2.092	1.726	2.094	2.069
A mezz'aria	0.896	0.683	0.929	1.863
Pallonetto	2.539	2.530	2.905	2.947
Normale	1.368	1.090	1.331	2.481
Rovesciata	0.038	-0.093	0.085	-0.084
Al volo	0.853	0.626	0.862	1.770
Parte del corpo:				
Piede sinistro	1.380	1.409	1.358	1.518
Altro	0.607	1.008	0.823	-0.132
Piede destro	1.452	1.441	1.411	1.635
Contesto del tiro:				
Punizione	-4.191	-3.330	-5.397	-2.158
Azione regolare	-5.368	-4.571	-3.386	-3.428
Rigore	-3.861	-3.030	0.013	0.107
Casa:				
In trasferta	0.188	0.000	0.137	0.243
In casa	0.195	-0.263	0.144	0.236
Sotto pressione:				
Sotto pressione	-0.340	0.000	-0.303	-0.385
Tiro di prima:				
Tiro di prima	-0.116	-0.119	-0.077	-0.102
Azione precedente:				
Distanza di tiro	-0.193	-0.182	-0.182	-0.202
Angolo di tiro:				
Angolo di tiro	0.013	0.014	-4.219	0.014

TABELLA 3.4: Stime dei parametri della regressione logistica con approccio *forward* al variare del *dataset* di stima. Le linee tratteggiate separano i parametri relativi a diverse variabili esplicative.

[1]L'abbreviazione "Unders." sta per "Undersampled".
[2]L'abbreviazione "Overs." sta per "Oversampled".

In questo caso, l'abbondanza di variabili esplicative non permette un'interpretazione di ciascuna di esse attraverso il rapporto di quote, in quanto risulta poco realistico far variare una singola variabile tenendo fissate le altre. Tuttavia, è comunque possibile trarre delle conclusioni sulla significatività delle variabili, nonostante queste siano influenzate dall'elevata numerosità campionaria. Per limitare questo effetto si decide di fissare un livello di significatività pari a 0.01. Si nota che in nessuno dei quattro casi le stime dei coefficienti relativi alla modalità "Rovesciata" della variabile **Tecnica di tiro** e alla modalità "Other" della variabile **Azione precedente** risultano significativamente diverse da 0. Ricordando che le modalità di riferimento delle due variabili sono rispettivamente "Colpo di tacco" e "Corner", si può concludere che, a parità di altre esplicative, la probabilità di segnare con un colpo di tacco è uguale a quella di segnare con una rovesciata, e che, a parità di altre esplicative, la probabilità di segnare sugli sviluppi di un'azione da calcio d'angolo è uguale alla probabilità di segnare sugli sviluppi di un'azione "Other".

In Tabella 3.5 sono riportate le metriche di valutazione per ogni modello stimato. Si noti che si è ottenuto un miglioramento di tutte le metriche rispetto alla Tabella 3.3. Anche in questo caso si può osservare una grande differenza nelle soglie individuate nel caso dei modelli stimati sui *dataset* bilanciati e non.

Training set	AUC	Soglia	F-score	$1-\hat{\alpha}$	$1-\hat{\beta}$	Accuratezza
Sbilanciato	**0.816**	0.188	**0.424**	0.900	0.526	0.864
Undersampled	0.815	0.644	0.419	**0.901**	0.515	0.864
Oversampled	0.815	0.649	0.424	0.900	0.523	**0.865**
SMOTE-NC	0.812	0.658	0.424	0.898	**0.530**	0.863

TABELLA 3.5: Tabella delle metriche dei modelli di regressione logistica con un approccio *forward* registrate sul *dataset* di valutazione al variare dei *dataset* di stima.

3.4 Regressione regolarizzata: il Lasso

Il *Lasso* è una tecnica di regolarizzazione che consiste nell'aggiunta di un vincolo allo spazio parametrico. In particolare viene aggiunto un vincolo di tipo valore assoluto all'equazione di stima dei coefficienti della regressione logistica, in questo modo sono penalizzati i valori alti dei parametri. La funzione da massimizzare per ottenere le stime dei coefficienti di regressione logistica diventa quindi:

$$\max_{\beta_0,\beta} \left\{ \sum_{i=1}^{N} [y_i(\beta_0 + \beta^T x_i) - \log(1 + e^{\beta_0+\beta^T x_i})] - \lambda \sum_{j=1}^{p} |\beta_j| \right\},$$

che possiamo riscrivere come

$$\min_{\beta_0,\beta} \left\{ \sum_{i=1}^{N} [y_i(\beta_0 + \beta^T x_i) - \log(1 + e^{\beta_0 + \beta^T x_i})] \right\} \text{ soggetta al vincolo: } \sum_{j=1}^{p} |\beta_j| \leq t.$$

Geometricamente questo tipo di vincolo trasla le stime dei coefficienti verso 0, troncandole quando arrivano a 0 (Tibshirani, 1996). Grazie al fatto che tipicamente diversi coefficienti sono stimati come nulli questo approccio è molto utile, infatti in tal modo si ottiene una selezione delle variabili più importanti, aumentando l'interpretabilità dei risultati. Si noti che nel termine di penalità non è incluso il parametro β_0, ossia l'intercetta, in quanto non viene penalizzato. Inoltre le variabili esplicative vengono standardizzate, ossia vengono portate a media nulla e varianza unitaria, in modo da evitare che la scala delle variabili influenzi i risultati (Hastie et al., 2009).

Il problema per il caso in esame è la presenza di fattori a più di due livelli, ossia le variabili **Azione precedente**, **Contesto del tiro**, **Parte del corpo**, **Tecnica di tiro** e **Casa**. Per adattare un modello di regressione logistica è quindi necessario trasformare ciascuna di queste variabili qualitative con k livelli in $k-1$ variabili dicotomiche. In questi casi però la penalizzazione *Lasso* non riuscirà a fare una selezione delle variabili, in quanto la penalizzazione riguarderà solo il coefficiente relativo ad una singola modalità della variabile e non a tutte le modalità della stessa. Tuttavia l'eventuale annullamento di uno dei coefficienti indicherà la mancanza di differenza tra la modalità di una variabile e la modalità di riferimento rappresentata dall'intercetta. Per le cinque variabili le modalità di riferimento sono, rispettivamente, "Calcio d'angolo", "Calcio d'angolo", "Colpo di testa", "Colpo di tacco" e "Campo neutro".

Questo modello è stato implementato in R utilizzando la versione 4.1-8 della libreria `glmnet` (Friedman et al., 2022). Oltre alla stima dei parametri, è stata eseguita una procedura di convalida incrociata per determinare il valore ottimale di λ. In particolare, per ogni *dataset* di stima sono stati selezionati due valori di λ: $\lambda_{\min}$ e $\lambda_{(1\text{ s.e.})}$, che corrispondono alle due linee tratteggiate verticali in Figura 3.5. $\lambda_{\min}$ è il valore di λ che minimizza l'errore in convalida incrociata, rappresentato dalla linea tratteggiata più a sinistra in Figura 3.5. Invece, $\lambda_{(1\text{ s.e.})}$ è il massimo valore di λ per cui la devianza corrispondente è inferiore di al massimo una deviazione standard rispetto alla devianza associata a $\lambda_{\min}$, rappresentato dalla linea tratteggiata più a destra in Figura 3.5.

In Tabella 3.6 sono riportate le metriche relative al λ che ha ottenuto i migliori risultati sul *dataset* di valutazione. Possiamo notare che si ha il valore di λ più piccolo nel caso del modello stimato sui dati sbilanciati, quindi in questo caso la penalizzazione è minore. Il valore massimo di λ si è ottenuto invece sui dati bilanciati con la procedura

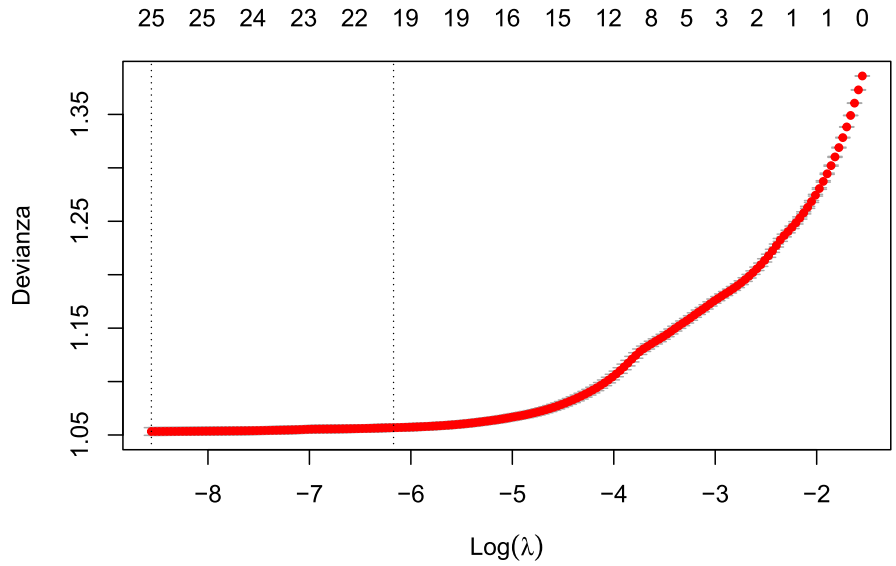

FIGURA 3.5: Grafico dell'errore in convalida incrociata al variare del parametro $\log(\lambda)$ per il modello stimato sui dati bilanciati con SMOTE-NC.

di *undersampling*, quindi in questo caso la penalizzazione è maggiore. Anche in questo caso si può osservare che il valore della soglia è simile nei tre modelli stimati sui dati bilanciati. Mentre la soglia del modello stimato sui dati sbilanciati è molto inferiore alle altre tre. Inoltre è possibile osservare che si ha un calo di tutte le metriche, tranne la sensibilità, rispetto ai modelli stimati con procedura *forward*. Quindi i modelli con penalità di tipo *Lasso* funzionano meglio solo nel prevedere il corretto valore dei goal osservati, ossia dei successi osservati.

Training set	$\log(\lambda)$	AUC	Soglia	F-score	$1-\hat{\alpha}$	$1-\hat{\beta}$	Accuratezza
Sbilanciato	-7.898	**0.815**	0.162	0.416	0.886	0.548	0.854
Undersampled	-4.913	0.810	0.488	0.416	**0.894**	0.528	**0.859**
Oversampled	-5.836	0.814	0.454	0.418	0.886	**0.550**	0.854
SMOTE-NC	-6.391	0.812	0.424	**0.422**	0.892	0.543	0.859

TABELLA 3.6: Tabella delle metriche dei modelli di regressione logistica con penalizzazione di tipo *Lasso* sul *dataset* di valutazione al variare dei *dataset* di stima.

In Tabella 3.7 sono invece riportati i coefficienti stimati al variare del dataset, anche in questo caso i parametri relativi alla stessa variabile esplicativa sono stati separati da linee tratteggiate. Possiamo vedere che l'unico coefficiente che risulta nullo per ogni modello stimato è quello relativo alla modalità "Calcio di punizione" della variabile **Contesto del tiro**. Possiamo quindi concludere che secondo tutti modelli, a parità

delle altre esplicative, la probabilità di segnare direttamente da un calcio d'angolo è uguale a quella di segnare da un calcio di punizione.

	Stime dei parametri con i diversi dataset			
Parametro	**Sbilanciati**	**Unders.**[1]	**Overs.**[2]	**SMOTE-NC**
Intercetta	-0.661	1.810	1.703	1.393
Azione precedente:				
Contropiede	0.871	0.593	0.719	0.773
Punizione	0.250	0	0.070	0.249
Rinvio dal fondo	0.390	0.019	0.176	0
Dal portiere	0.530	0.008	0.335	-0.232
Calcio d'inizio	0.226	-0.101	0	-1.063
Rimessa laterale	0.340	0.071	0.172	0
Altro	0.469	0.434	0	0.430
Azione regolare	0.398	0.068	0.236	0.350
Tecnica di tiro:				
Di testa in tuffo	1.183	0.276	0.888	0.085
A mezz'aria	0.052	0	0	0.049
Pallonetto	1.671	1.253	1.725	1.012
Normale	0.519	0.222	0.353	0.653
Rovesciata	-0.725	-0.293	-0.625	-1.670
Al volo	0.001	0	-0.021	0
Parte del corpo:				
Piede sinistro	1.299	0.881	1.138	1.354
Altro	0.554	0.239	0.539	-0.017
Piede destro	1.370	0.927	1.193	1.473
Contesto del tiro:				
Punizione	0	0	0	0
Azione regolare	-1.152	-1.005	-1.154	-1.231
Rigore	0.410	0.227	0.671	0.565
Casa:				
In trasferta	0.146	0	0.004	0.162
In casa	0.154	0	0.015	0.155
Sotto pressione:				
Sotto pressione	-0.336	-0.233	-0.286	-0.368
Tiro di prima:				
Tiro di prima	-0.076	0	0	-0.037
Azione precedente:				
Distanza di tiro	-0.187	-0.150	-0.166	-0.191
Angolo di tiro:				
Angolo di tiro	0.012	0.010	0.011	0.013

TABELLA 3.7: Stime dei parametri della regressione logistica con penalità di tipo *Lasso* al variare del *dataset* di stima. Le linee tratteggiate separano i parametri relativi a diverse variabili esplicative.

[1] L'abbreviazione "Unders." sta per "Undersampled".
[2] L'abbreviazione "Overs." sta per "Oversampled".

3.5 Regressione regolarizzata: il Grouped Lasso

Per eseguire la selezione delle variabili avendo dei fattori a più livelli è necessario considerare un altro tipo di penalizzazione, come il *Grouped Lasso*, poiché questo approccio consente di raggruppare i parametri relativi alla stessa variabile esplicativa penalizzandoli contemporaneamente (Hastie et al., 2009). Nel dettaglio i p parametri sono divisi in G gruppi e con p_g si indicherà la numerosità del gruppo g. L'equazione da risolvere per trovare le stime dei coefficienti di regressione logistica sarà quindi:

$$\min_{\beta \in \mathbb{R}^p} \left(l(\beta) + \lambda \sum_{g=1}^{G} \sqrt{p_g} ||\beta_g||_2 \right), \tag{3.3}$$

dove $l(\beta)$ è la funzione di log-verosimiglianza, ossia:

$$l(\beta) = \sum_{i=1}^{n} \left(y_i \eta_i - \log(1 + \exp(n_i)) \right),$$

come visto in precedenza l'intercetta non sarà penalizzata (Meier et al., 2008).

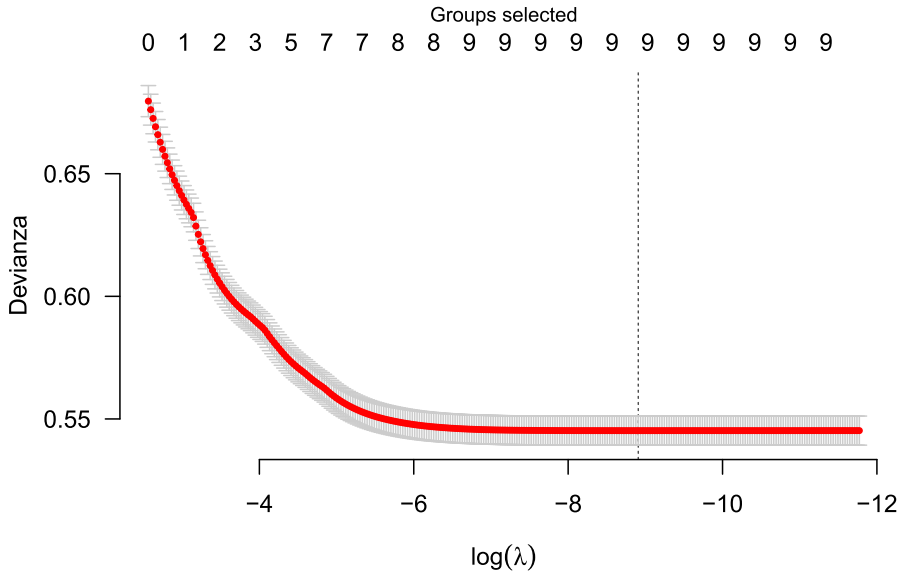

FIGURA 3.6: Grafico dell'errore in convalida incrociata al variare del parametro $\log(\lambda)$ per il modello stimato sui dati sbilanciati.

Questo modello è stato implementato in R con la versione 3.4.0 della libreria `grpreg` (Breheny et al., 2021). Il numero G di gruppi è stato posto uguale a 9, ossia al numero di variabili esplicative presenti. L'iper-parametro λ, che regola la penalizzazione, anche in questo caso è stato ottimizzato con una procedura di convalida incrociata in 10 blocchi.

In particolare è stato scelto il valore λ che corrisponde al modello con il minimo errore in convalida incrociata. In Figura 3.6 è riportato un esempio del criterio per la scelta di λ.

Training set	log(λ)	AUC	Soglia	F-score	$1-\hat{\alpha}$	$1-\hat{\beta}$	Accuratezza
Sbilanciato	-8.908	**0.816**	0.158	**0.418**	0.880	**0.567**	0.850
Undersampled	-7.046	0.815	0.633	0.417	0.898	0.520	0.862
Oversampled	-10.711	0.816	0.618	0.418	0.889	0.544	0.856
SMOTE-NC	-10.750	0.790	0.600	0.406	**0.900**	0.495	**0.862**

TABELLA 3.8: Tabella delle metriche dei modelli di regressione logistica con penalizzazione di tipo *Grouped Lasso* sul *dataset* di valutazione al variare dei *dataset* di stima.

In Tabella 3.8 sono riportate invece le metriche per i quattro modelli stimati. Possiamo osservare che la procedura di convalida incrociata ha portato a scegliere il valore maggiore di λ, e quindi il maggior valore della penalizzazione, per il modello stimato sui dati bilanciati con *undersampling*. Si noti che sui dati bilanciati con SMOTE-NC il modello con penalità *Grouped Lasso* ottiene risultati inferiori rispetto al modello con penalità *Lasso*, tranne che per la specificità.

In Tabella 3.9 sono riportate le stime dei parametri al variare del dataset. Le linee tratteggiate separano i parametri relativi allo stesso gruppo. Possiamo vedere che nessun gruppo di coefficienti è stato stimato pari a 0, quindi nessuna variabile esplicativa è stata rimossa.

3.6 Alberi di classificazione

L'approccio che utilizza gli alberi binari di classificazione si basa sul partizionare lo spazio delle covariate per poi stimare le probabilità di successo in base alla proporzione campionaria dei successi in ogni partizione dello spazio (Hastie et al., 2009).

Gli alberi di classificazione si possono applicare in qualsiasi contesto in cui sono date n osservazioni composte dalla variabile risposta discreta y_i e dal vettore delle esplicative p-variato $x_i = (x_{i1}, \ldots, x_{ip}) \in \Phi$, $i = 1, \ldots, n$, dove $\Phi \subseteq \mathbb{R}^p$ è lo spazio delle covariate. L'obbiettivo degli alberi di classificazione è di dividere lo spazio Φ in M regioni $R_1, R_2, \ldots, R_M$ in modo da minimizzare una funzione di perdita. Nel caso in esame gli alberi sono stati fatti crescere liberamente, ponendo come unico vincolo il fatto che su ogni foglia, ossia su ogni nodo terminale, ci dovessero essere almeno 5 osservazioni.

	Stime dei parametri con i diversi dataset			
Parametro	**Sbilanciati**	**Unders.**[1]	**Overs.**[2]	**SMOTE-NC**
Intercetta	2.558	3.996	4.660	1.642
Azione precedente:				
Contropiede	0.932	0.896	0.951	0.851
Punizione	0.323	0.216	0.315	0.339
Rinvio dal fondo	0.466	0.432	0.444	0.086
Dal portiere	0.614	0.514	0.647	-0.267
Calcio d'inizio	0.343	-0.189	0.332	-1.204
Rimessa laterale	0.408	0.414	0.409	-1.344
Altro	0.602	0.740	0.175	0.503
Azione regolare	0.459	0.346	0.450	0.436
Tecnica di tiro:				
Di testa in tuffo	2.051	1.620	2.091	2.065
A mezz'aria	0.880	0.643	0.927	1.860
Pallonetto	2.502	2.370	2.901	2.943
Normale	1.340	1.028	1.329	2.478
Rovesciata	0.040	-0.099	0.085	-0.084
Al volo	0.836	0.587	0.861	1.768
Parte del corpo:				
Piede sinistro	1.357	1.354	1.357	1.516
Altro	0.605	1.003	0.823	-0.131
Piede destro	1.428	1.384	1.409	1.633
Contesto del tiro:				
Punizione	-4.124	-3.214	-4.215	-2.154
Azione regolare	-5.270	-4.404	-5.392	-3.424
Rigore	-3.812	-3.074	-3.448	-0.177
Casa:				
In trasferta	0.179	0.101	0.136	0.243
In casa	0.187	0.089	0.143	0.236
Sotto pressione:				
Sotto pressione	-0.337	-0.258	-0.303	-0.385
Tiro di prima:				
Tiro di prima	-0.107	-0.102	-0.076	-0.102
Azione precedente:				
Distanza di tiro	-0.192	-0.178	-0.182	-0.202
Angolo di tiro:				
Angolo di tiro	0.013	0.013	0.013	0.014

TABELLA 3.9: Stime dei parametri della regressione logistica con penalità di tipo *Grouped Lasso* al variare del *dataset* di stima. Le linee tratteggiate separano i parametri relativi allo stesso gruppo.

Successivamente gli alberi sono stati potati annullando quelle divisioni che hanno provocato un calo della funzione di perdita inferiore alla quantità *cp* (*complexity-parameter*).

[1]L'abbreviazione "Unders." sta per "Undersampled".
[2]L'abbreviazione "Overs." sta per "Oversampled".

Quindi maggiore è stato scelto il valore di *cp* minore è il numero di divisioni nell'albero.

Gli alberi di classificazione sono stati implementati in R con la versione 4.1.23 della libreria `rpart` (Therneau e Atkinson, 2023). Nel caso in esame, si è scelto di creare due alberi per ogni *dataset* di stima, cambiando la misura della funzione di perdita. Le due funzioni di perdita adottate sono:

$$\text{Indice di Gini} : \sum_{k \neq k'} \hat{p}_{mk}\hat{p}_{mk'} = \sum_{k=1}^{K} \hat{p}_{mk}(1 - \hat{p}_{mk}),$$

$$\text{Entropia incrociata} : -\sum_{k=1}^{K} \hat{p}_{mk} \log \hat{p}_{mk},$$

dove $\hat{p}_{mk}$ rappresenta la probabilità stimata che per una osservazione nella regione m si osservi la k-esima modalità della variabile risposta. Nel caso in esame si ha $K = 2$. Successivamente è stato studiato l'errore in convalida incrociata al variare dell'iperparametro *cp* ed è stato scelto il valore di *cp* che ha minimizzato tale errore.

Le metriche di valutazione dei modelli stimati sono presentate in Tabella 3.10 e in Tabella 3.11. Si può osservare che i modelli stimati con l'Entropia incrociata ottengono una maggiore accuratezza rispetto ai corrispondenti modelli stimati con l'indice di Gini. Questo è dovuto soprattutto a un aumento della specificità. Al contrario, la sensibilità passando dall'indice di Gini all'entropia incrociata cala in 3 casi su 4. A livello di F-score si ottengono valori peggiori rispetto ai modelli di regressione logistica stimati in precedenza, dunque si ritiene che questo approccio non sia riuscito a migliorare le previsioni del modello.

Training set	*cp*	AUC	Soglia	F-score	$1 - \hat{\alpha}$	$1 - \hat{\beta}$	Accuratezza
Sbilanciato	$6.56 \cdot 10^{-4}$	0.662	0.126	0.365	**0.923**	0.387	**0.872**
Undersampled	$2.40 \cdot 10^{-3}$	0.771	0.720	0.383	0.868	**0.534**	0.836
Oversampled	$8.72 \cdot 10^{-6}$	0.665	0.822	0.321	0.890	0.390	0.843
SMOTE-NC	$5.23 \cdot 10^{-5}$	**0.780**	0.786	**0.384**	0.890	0.487	0.852

TABELLA 3.10: Tabella delle metriche al variare dei *dataset* di stima degli alberi di classificazione con indice di Gini come funzione di perdita.

Gli alberi di classificazione inoltre permettono di avere un'interpretazione dell'importanza delle variabili in base al miglioramento dell'adattamento del modello, come indicato dagli autori della libreria di R `rpart` in Therneau e Atkinson (2023). Nel dettaglio essi quantificano l'importanza di una variabile come la somma dei miglioramenti della funzione di perdita, ossia della somma dei cali della funzione di perdita nei nodi in cui appare tale variabile. L'importanza delle variabili al variare dei quattro *dataset* di

Training set	*cp*	AUC	Soglia	F-score	$1-\hat{\alpha}$	$1-\hat{\beta}$	Accuratezza
Sbilanciato	$1.09 \cdot 10^{-3}$	0.750	0.187	0.354	**0.967**	0.282	**0.902**
Undersampled	$2.40 \cdot 10^{-3}$	**0.775**	0.732	0.389	0.885	**0.505**	0.850
Oversampled	$1.74 \cdot 10^{-5}$	0.660	0.811	0.318	0.894	0.380	0.846
SMOTE-NC	$5.75 \cdot 10^{-5}$	0.771	0.788	**0.390**	0.893	0.489	0.854

TABELLA 3.11: Tabella delle metriche al variare dei *dataset* di stima degli alberi di classificazione con Entropia incrociata come funzione di perdita.

stima, nel caso dell'albero la cui funzione di impurità massimizzi l'F-score, è rappresentata da Figura 3.7 a Figura 3.10. Possiamo notare che le interpretazioni dell'importanza delle variabili sono molto diverse tra di loro. Nel caso dei dati sbilanciati risultano molto più importanti le variabili **Contesto di tiro**, **Azione precedente** e **Distanza dalla porta**, mentre le altre 6 hanno una importanza molto inferiore. Nel caso dei dati bilanciati con *undersampling* pseudo-casuale possiamo vedere che sono state utilizzate solo 6 delle 9 variabili esplicative nella realizzazione delle divisioni. Ciò è dovuto anche al fatto che avendo meno osservazioni è stato possibile effettuare meno divisioni. In questo caso la variabile **Distanza dalla porta** è quella con importanza maggiore, oltre tre volte ciascuna delle altre. Nel caso degli alberi stimati sui dati bilanciati con *oversampling* pseudo-casuale e con SMOTE-NC i risultati invece sono graficamente simili. In entrambi i casi le tre variabili più importanti sono **Distanza dalla porta**, **Angolo di tiro** e **Azione precedente**.

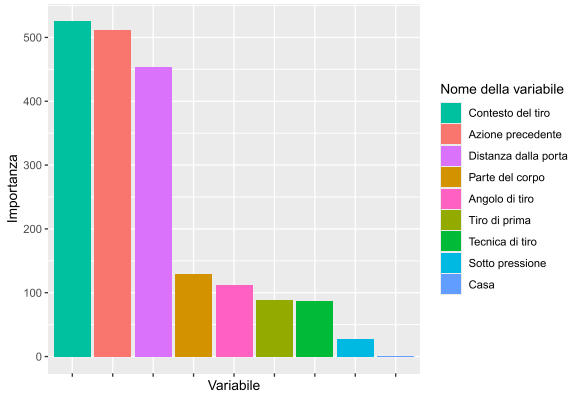

FIGURA 3.7: Grafico dell'importanza delle variabili sulla base dell'albero di classificazione stimato con Indice di Gini sul *dataset* sbilanciato.

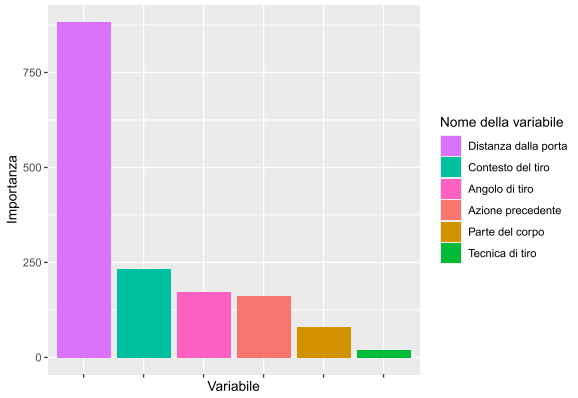

FIGURA 3.8: Grafico dell'importanza delle variabili sulla base dell'albero di classificazione stimato con Entropia incrociata sul *dataset* bilanciato con *undersampling*.

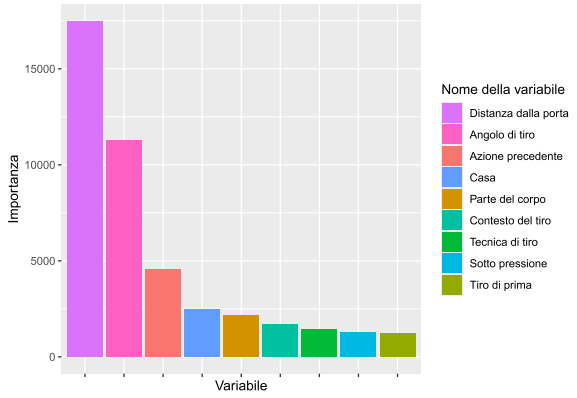

FIGURA 3.9: Grafico dell'importanza delle variabili sulla base dell'albero di classificazione stimato con Indice di Gini sul *dataset* bilanciato con *oversampling*.

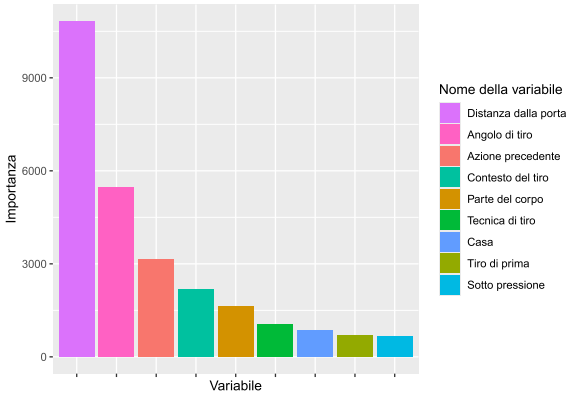

FIGURA 3.10: Grafico dell'importanza delle variabili sulla base dell'albero di classificazione stimato con entropia incrociata sul *dataset* bilanciato con SMOTE-NC.

3.7 Random Forest

Gli alberi di classificazione, tuttavia, presentano alcuni limiti, per esempio sono molto sensibili a piccole variazioni nei dati (Zhang e Singer, 2010). Inoltre, sono modelli caratterizzati da bassa distorsione ma alta varianza (Hastie et al., 2009). Per migliorare questi aspetti, sono state sviluppate le *Random Forest*. Queste si basano su un approccio aggregativo che utilizza un insieme di classificatori che lavorano assieme per prevedere il valore della variabile risposta di ogni osservazione (More e Rana, 2017). Tale approccio è ispirato al *bagging* (*bootstrap aggregating*), con una differenza chiave: nelle *Random Forest*, non vengono utilizzati tutti i predittori in ogni modello, ma solo un campione di essi scelto pseudo-casualmente.

Nel dettaglio l'algoritmo usato è quello disponibile nella versione 4.7-1.1 della libreria `randomForest` di R. Questo si basa su quello proposto in Breiman (2001b) e scritto inizialmente in Fortran. Può essere riassunto nei seguenti passaggi:

1. Viene estratto un campione *bootstrap* (quindi con reinserimento) dai dati, di dimensioni uguale a quella dei dati;

2. Viene fatto crescere l'albero su questo campione ripetendo i seguenti due passaggi per ogni nodo:

   - Vengono selezionate m variabili pseudo-casualmente tra le p variabili esplicative. Tipicamente m è pari a $\log(p)$ o a $\sqrt{p}$ (Zhang e Singer, 2010);

- Viene effettuata la miglior divisione possibile considerando le m variabili estratte;

3. La crescita dell'albero si arresta quando ogni foglia raggiunge la dimensione minima fissata, solitamente posta pari a 1.

La procedura viene ripetuta tante volte quanti alberi si desidera avere nella foresta. Una particolarità di questo approccio è che gli alberi della foresta non vengono potati, al contrario di quanto effettuato nella sezione 3.6. Il fatto di non potare gli alberi non comporta un aumento del rischio di *overfitting*, come mostrato in Breiman (2001a). Inoltre in Breiman (2001b), si è osservato che neanche il troppo elevato numero di alberi in una foresta è un fattore che aumenta il rischio di *overfitting*.

Un aspetto importante delle *Random Forest* sono le stime *out-of-bag* (OOB). Queste stime sono ottenibili avendo applicato una procedura di *bagging*. Infatti, avendo scelto un diverso campione *bootstrap* per l'stima di ogni singolo albero, una generica osservazione z_i, formata da una variabile risposta y_i e da un vettore di esplicative x_i, è stata usata per stimare solo una frazione degli alberi. Possiamo quindi utilizzare gli alberi in cui z_i non appare per ottenere una previsione del valore di y_i.

In Breiman (2001b) è stato stimato che circa un terzo delle osservazioni non sia presente in ogni nuovo campione bootstrap, quindi le stime OOB si basano su circa un terzo degli alberi presenti nella foresta. Quest'ultima caratteristica tuttavia rende gli errori OOB sovra-stimati, in quanto l'errore in una *Random Forest* decresce al crescere del numero di alberi.

Un'altra proprietà delle stime OOB è che sono non distorsione, al contrario delle stime ottenute con una procedura di convalida incrociata (Breiman, 2001b). Quindi nelle *Random Forest* le stime OOB sono preferibili alle stime in convalida incrociata. Tuttavia affinchè le stime OOB siano non distorte è necessario superare il numero di alberi per cui l'errore nel *dataset* di valutazione converge.

	Errore OOB al variare del *dataset* di stima			
m	Sbilanciato	Undersampled	Oversampling	SMOTE-NC
2	9.67%	27.75%	-	-
3	9.67%	27.95%	8.72%	18.86%
4	9.84%	28.89%	3.41%	16.89%
6	10.09%	29.48%	2.25%	16.94%
9	-	-	2.50%	-

TABELLA 3.12: Errore OOB delle *Random Forest* stimate al variare dell'iperparametro m.

L'iperparametro m introdotto in precedenza è stato ottimizzato basandosi sulle stime dell'errore OOB. In particolare, è stata utilizzata la funzione `tuneRF` della libreria `randomForest`, che, partendo da un dato valore di m (in questo caso 4), cerca il valore che minimizza l'errore OOB. Il valore iniziale di m viene moltiplicato o diviso per 1.5 finché queste variazioni comportano una riduzione dell'errore OOB. L'algoritmo si arresta quando nessuna variazione di m provoca una diminuzione dell'errore OOB. I risultati sono riportati nella Tabella 3.12.

Training set	m	AUC	Soglia	F-score	$1-\hat{\alpha}$	$1-\hat{\beta}$	Accuratezza
Sbilanciato	2	0.779	0.136	0.391	0.893	0.489	0.850
Undersampled	3	**0.804**	0.716	**0.416**	**0.894**	**0.527**	**0.859**
Oversampled	6	0.768	0.278	0.362	0.873	0.488	0.836
SMOTE-NC	4	0.777	0.602	0.352	0.878	0.462	0.839

TABELLA 3.13: Tabella delle metriche al variare dei *dataset* di stima delle *Random Forest*.

I risultati sul *dataset* di valutazione dei modelli stimati sono riportati in Tabella 3.13. Possiamo osservare che, per la prima volta, le soglie scelte nei tre modelli stimati sui dati bilanciati differiscono di oltre un punto decimale. Questo perché la soglia del modello stimato sui dati bilanciati con *oversampling* casuale risulta molto più bassa di quanto visto in precedenza. Inoltre, è importante sottolineare che, per la prima volta, un modello stimato su un *dataset* riesce a ottenere valori superiori in tutte le metriche rispetto agli altri. Questo risultato si ottiene nel modello stimato sui dati bilanciati con *undersampling*.

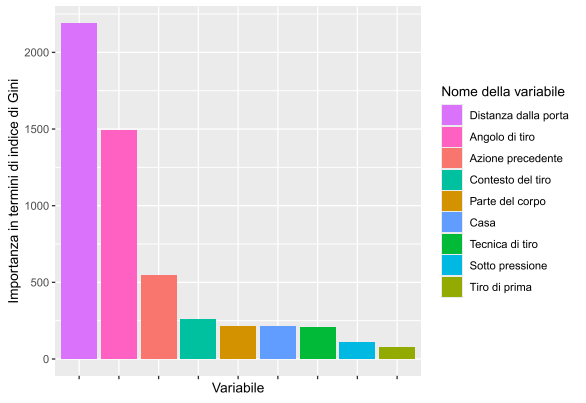

FIGURA 3.11: Grafico dell'importanza delle variabili sulla base della *Random Forest* stimata sul *dataset* sbilanciato.

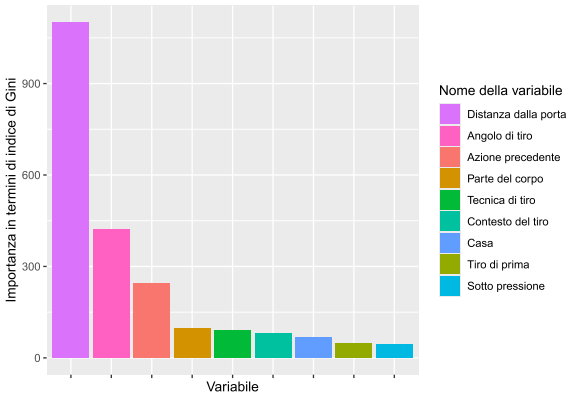

FIGURA 3.12: Grafico dell'importanza delle variabili sulla base della *Random Forest* stimata sul *dataset* bilanciato con *undersampling*.

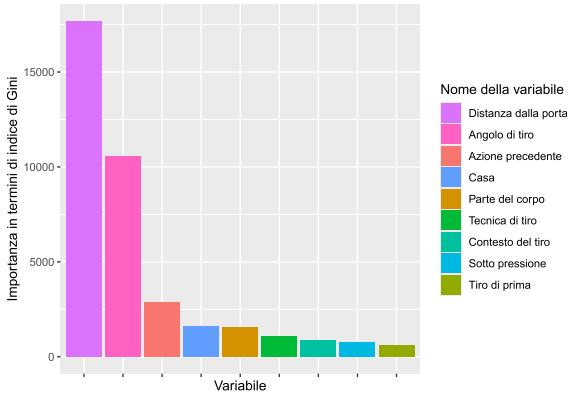

FIGURA 3.13: Grafico dell'importanza delle variabili sulla base della *Random Forest* stimata sul *dataset* bilanciato con *oversampling*.

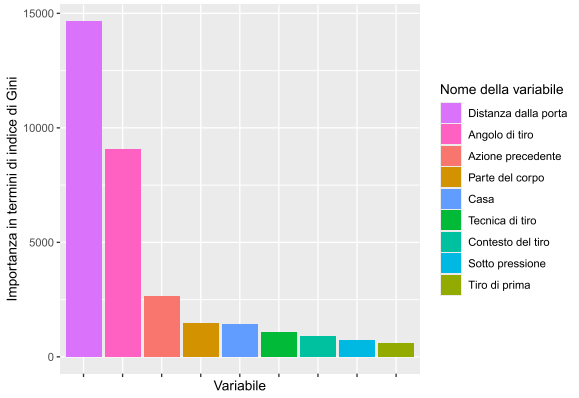

FIGURA 3.14: Grafico dell'importanza delle variabili sulla base della *Random Forest* stimata sul *dataset* bilanciato con SMOTE-NC.

Ai fini interpretativi le *Random Forest* permettono di ottenere una misura dell'importanza delle variabili in maniera analoga a quanto visto per gli alberi di classificazione. Quindi l'importanza di una variabile in un albero è pari alla somma dei cali della funzione di perdita provocati dalle divisioni in cui viene utilizzata questa variabile. L'importanza di una variabile nella *Random Forest* è pari alla media dell'importanza di ogni variabile tra tutti gli alberi nella foresta. Nei casi in esame l'importanza delle variabili è mostrata da Figura 3.11 a Figura 3.14. I risultati sono simili tra loro, infatti in ciascuno dei 4 casi le variabili più importanti sono "Distanza dalla porta", "Angolo di tiro" e "Azione precedente". Possiamo notare che i risultati sono simili a quelli degli alberi di classificazione stimati su i dati bilanciati con *oversampling* pseudo-casuale e SMOTE-NC.

3.8 Confronto tra i migliori modelli

In quest'ultima sezione verranno confrontati i casi migliori per F-score di ogni modello presentato in precedenza. I risultati sono riassunti in Tabella 3.14. Possiamo notare che sui 7 modelli presentati, in ben 4 casi il migliore è stato quello stimato sui dati bilanciati con SMOTE-NC, mentre in nessun caso il migliore è stato stimato sui dati bilanciati con *oversampling* casuale. Sul *dataset* bilanciato con *undersampling* si è ottenuta la miglior *Random Forest*, mentre sul *dataset* sbilanciato è stato ottenuto il miglior modello in due occasioni.

Valutando la bontà di adattamento del modello di regressione logistica con solo distanza dalla porta e angolo di tiro si trova che stime dei 3 coefficienti risultano significative con p-value minore dell'1%. Il test del log-rapporto di verosimiglianza con il modello con solo intercetta porta a preferire il modello corrente.

Modello	*Dataset*	AUC	Soglia	F-score	$1-\hat{\alpha}$	$1-\hat{\beta}$	Acc.[1]
Logistica: d+a[2]	SMOTE-NC	0.774	0.700	0.360	0.893	0.445	0.850
Logistica: Forw.[3]	Sbilanciato	0.815	0.188	**0.424**	**0.900**	0.526	**0.864**
Logistica: Lasso	SMOTE-NC	0.812	0.424	0.422	0.892	0.543	0.859
Logistica: G.L.[4]	Sbilanciato	**0.816**	0.158	0.418	0.880	**0.567**	0.850
Albero Gini	SMOTE-NC	0.780	0.786	0.384	0.890	0.487	0.852
Albero Entropia	SMOTE-NC	0.771	0.788	0.390	0.893	0.489	0.854
Random Forest	*Unders.*[5]	0.804	0.716	0.416	0.894	0.527	0.859

TABELLA 3.14: Tabella delle metriche del caso migliore di ogni modello in termine di F-score. In grassetto sono evidenziati i massimi osservati per ciascuna metrica.

Il miglior modello di regressione logistica stimato con procedura *forward* include tutte le variabili esplicative nel modello. L'analisi della varianza indica che tutte le variabili sono significative con p-value$<$0.01, tranne per la variabile Tiro di prima il cui p-value era compreso tra 0.01 e 0.02. Il test del log-rapporto di verosimiglianza indica un significativo miglioramento rispetto al modello con solo intercetta e rispetto a quello con solo distanza e angolo di tiro. Anche l'AIC porta a preferire il modello stimato con procedura *forward*. Si noti inoltre che si è ottenuto un miglioramento di tutte le metriche di valutazione. Si ha perciò una conferma del miglioramento del modello, anche se non è possibile affermare se l'aumento di queste metriche sia significativo.

Il modello migliore di regressione logistica con penalità di tipo *Lasso* si è ottenuto in corrispondenza di $\log(\lambda)$ pari a -6.391. Si noti che si è ottenuto un calo di tutte le metriche tranne che per la sensibilità rispetto al miglior modello stimato con procedura *forward*. Inoltre sono stati stimati come nulli i coefficienti relativi alla modalità "Rinvio dal fondo" e alla modalità "Altro" della variabile **Azione precedente**. Anche i coefficienti relativi alla modalità "Punizione" della variabile **Contesto del tiro** e alla modalità "Al volo" della variabile **Tecnica di tiro** sono stati stimati pari a 0. Questo implica che non è stata individuata come significativa la differenza tra queste modalità e le modalità di riferimento rappresentate dall'intercetta. Ricordiamo che sia per la variabile **Azione precedente** che per la variabile **Contesto di tiro** la modalità di

[1] L'abbreviazione "Acc." sta per "Accuratezza".
[2] Regressione logistica con solo distanza e angolo di tiro come variabile esplicative.
[3] L'abbreviazione "Forw." sta per "Forward".
[4] La sigla "G.L." sta per "*Grouped Lasso*".
[5] L'abbreviazione "*Unders.*" sta per "*Undersampled*".

riferimento è "Calcio d'angolo", mentre per la variabile **Tecnica di tiro** la modalità di riferimento è "Colpo di tacco".

Il modello migliore di regressione logistica con penalità di tipo *Grouped Lasso* si è ottenuto in corrispondenza di $\log(\lambda)$ pari a -8.908. Nonostante la penalizzazione nessuno dei gruppi di coefficienti è stato posto pari a 0. Confrontando il modello con i risultati precedenti possiamo vedere che sia rispetto alla procedura *Lasso* che rispetto alla procedura *forward* si ha un aumento dell'AUC ma un calo dell'F-score.

L'albero di classificazione che massimizza l'F-score è quello stimato sui dati bilanciati con procedura SMOTE-NC e con l'entropia incrociata come funzione di perdita. Rispetto ai modelli parametrici precedenti si nota un netto calo dell'AUC, che è risultato inferiore anche di quello del modello con solo distanza dalla porta e angolo di tiro. Anche l'F-score cala notevolmente, mentre l'accuratezza è maggiore rispetto al *Grouped Lasso*. Tale aumento si può spiegare soprattutto grazie all'aumento della specificità.

La *Random Forest* migliore è stata stimata sui dati bilanciati con *Undersampling* pseudo-casuale. Si noti che si ha un miglioramento di tutte le metriche rispetto al miglior albero di classificazione, nonostante ciò il modello non ha superato i risultati ottenuti con i migliori modelli di regressione logistica.

Capitolo 4

Applicazioni alla valutazione delle prestazioni

4.1 Il miglior modello

Nel capitolo precedente si è visto che il miglior modello in termini di F-score è la regressione logistica stimata con procedura *forward* sui dati originali. Gli *Expected Goals* stimati sul *dataset* di valutazione da questo modello sono stati utilizzati in questo capitolo nell'ambito di due applicazioni degli *Expected Goals*.

In entrambe le applicazioni che seguono è stato necessario assumere l'indipendenza dei tiri, tuttavia questo assunto non è sempre valido in una partita di calcio, come già menzionato nella sezione 3.3, ma è necessario per semplificare la natura del problema.

4.2 Analisi di una partita

Come anticipato nella sezione 1.1, le stime ottenute degli *Expected Goals* vengono usate per valutare le prestazioni sia nel lungo, che nel breve periodo. In questa sezione vedremo come applicare le stime degli *Expected Goals* per valutare le prestazioni calcistiche nel breve periodo. Useremo quindi le probabilità stimate per analizzare una partita appartenente al test set, ossia Napoli-Juventus del 26 settembre 2015, conclusasi per 2-1 a favore del club partenopeo.

Possiamo vedere che sono stati realizzati 26 tiri in questa partita, 14 da parte del Napoli e 12 da parte della Juventus. Le posizioni da cui sono stati effettuati i tiri sono state rappresentate in Figura 4.1. In particolare i pallini rappresentano i tiri non

risultati in goal, mentre le stelle i tiri risultati in goal. Inoltre il colore blu rappresenta i tiri effettuati dal Napoli, mentre il bianco quelli effettuati dalla Juventus.

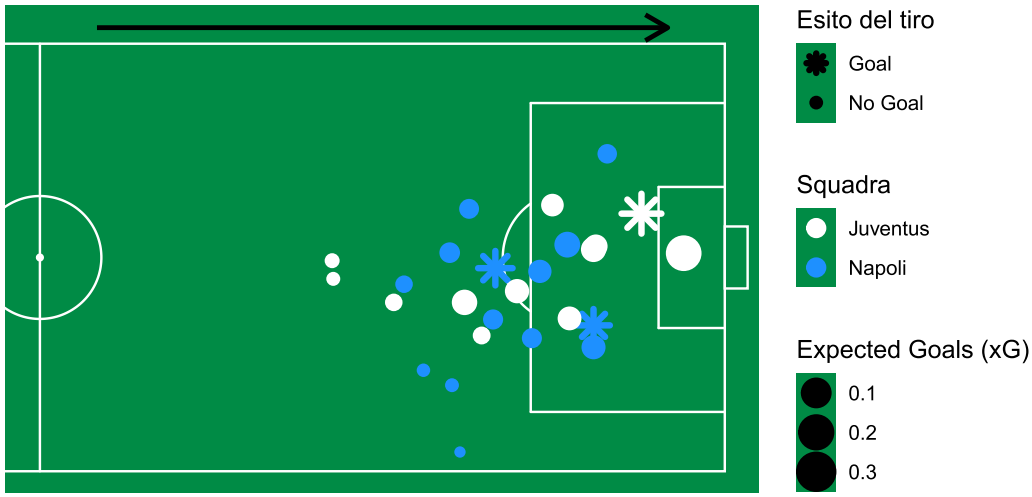

FIGURA 4.1: Mappa dei tiri di Napoli-Juventus della stagione 2015/2016.

Usando i risultati del modello stimato osserviamo che solo un tiro ha superato la soglia di 0.188 fissata dal modello scelto, ossia il pallino bianco più vicino alla porta in Figura 4.1. Il modello ha quindi stimato un solo goal, nella realtà i goal sono stati 3 e sono arrivati da tutt'altre conclusioni, emerge quindi una discordanza tra le prestazioni previste dal modello e la realtà. Inoltre, sommando gli *Expected Goals* dei singoli tiri per ogni squadra, si osserva che il Napoli ha totalizzato 0.74 xG mentre la Juventus 1.16 xG.

Dalla Figura 4.2, invece, possiamo vedere gli xG cumulati al variare del tempo. Questo ci permette di capire come si sono distribuite le occasioni da goal nel corso della singola partita. Nel caso in esame di Napoli-Juventus possiamo osservare la grande differenza a livello di xG ottenuti dalla Juventus tra il primo e secondo tempo, ossia tra i primi 45 minuti e i successivi. In particolare nel primo tempo ha cumulato poco più di 0.15 xG, mentre nel secondo circa 1 xG. Questa differenza può essere dovuta al fatto di aver subito un goal nel primo tempo, e quindi dalla necessità di dover recuperare il risultato nel secondo tempo.

Si può dunque concludere che, secondo il modello stimato, la Juventus avrebbe meritato di segnare più goal rispetto al Napoli. Tuttavia, il risultato finale della partita,

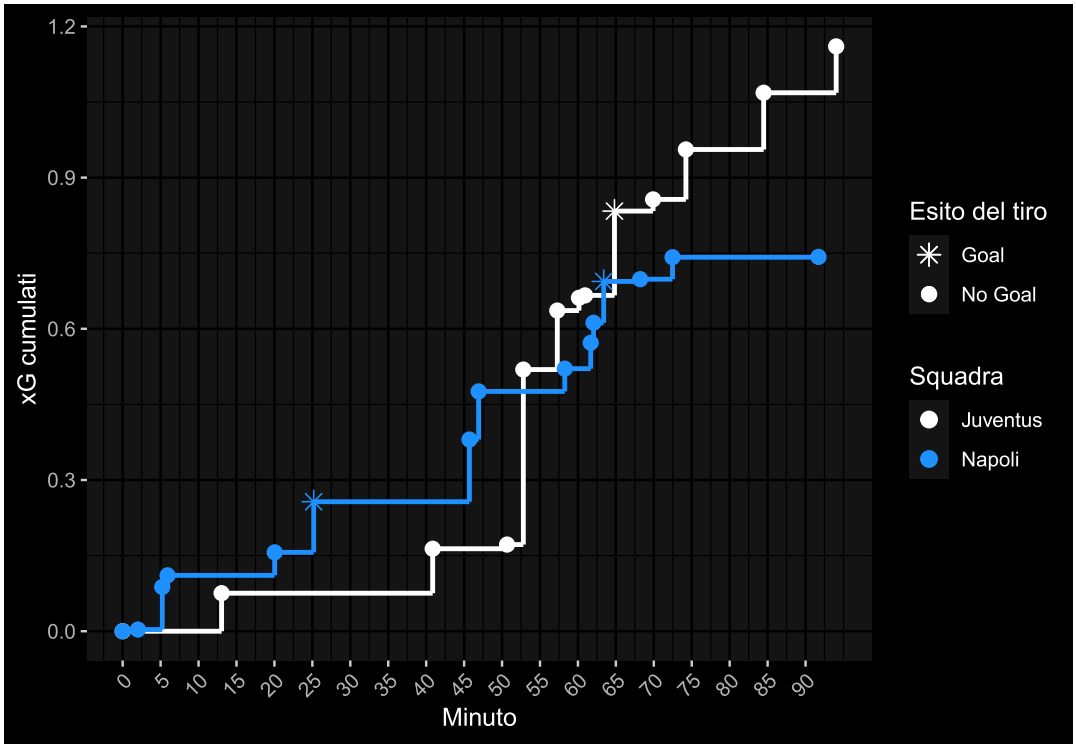

FIGURA 4.2: Mappa degli xG cumulati di Napoli-Juventus della stagione 2015/2016.

terminata 2-1 a favore dei campani, non rispecchia questa analisi. Come evidenziato dalle dichiarazioni di Arsène Wenger riportate nel primo capitolo, ciò accade piuttosto frequentemente. Gli *Expected Goals* infatti corrispondono più a una valutazione delle prestazioni e del merito di una squadra piuttosto che a un riflesso preciso del risultato.

4.3 Gli Expected Points

Gli *Expected Points* sono definiti nel seguente modo:

$$\begin{aligned}\text{Expected Points} = &(\text{Punti per la vittoria} \times \text{Probabilità di vittoria})+\\ &+(\text{Punti per il pareggio} \times \text{Probabilità di pareggio})+\\ &+(\text{Punti per la sconfitta} \times \text{Probabilità di sconfitta}).\end{aligned} \tag{4.1}$$

Interpretando questa quantità con un approccio frequentista, si può affermare che gli *Expected Points* riflettono il numero medio di punti che una squadra otterrebbe se la partita si giocasse un numero elevato di volte. Nella pratica, per calcolare questa quantità è necessario sostituire i punti per la vittoria, il pareggio e la sconfitta con i rispettivi valori di 3, 1 e 0. Successivamente, bisogna stimare le tre probabilità richieste, utilizzando i

metodi di Monte Carlo (Tippet, 2019).

Tali metodi comprendono una vasta gamma di approcci utilizzati per risolvere diversi tipi di problemi nelle scienze, nell'ingegneria e nell'informatica. Essi si basano sulla generazione di numeri pseudo-casuali per stimare le quantità di interesse (Kroese e Rubinstein, 2012).

La procedura di stima delle probabilità richieste si basa sulla seguente procedura:

1. Si stimano gli *Expected Goals*;

2. Si effettuano B simulazioni dei goal totali realizzati in una data partita. Nel dettaglio, per ogni tiro effettuato nella realtà si genererà il valore della variabile risposta **Goal** tramite simulazione da una Bernoulliana con probabilità di successo pari al rispettivo *xG*. Infine, per ognuna delle B simulazioni, si sommano i valori di **Goal** stimati per ogni squadra;

3. Per ognuna delle B simulazioni si stima la frazione di partite in cui una squadra ha vinto, ha perso e ha pareggiato. Tali frazioni corrispondono alle stime delle probabilità richieste.

Fissando B=10000 e applicando questo metodo al caso di Napoli-Juventus, analizzata nella Sezione 4.2, si stima che il Napoli ha una probabilità di vittoria pari al 22.0%, la probabilità di pareggio è del 31.2%, mentre la probabilità di vittoria della Juventus è del 46.8% circa. Quindi il Napoli ha totalizzato 0.973 xP, mentre la Juventus 1.715 xP. Nella realtà il Napoli ha ottenuto 3 punti, mentre la Juventus 0. Quindi, secondo il modello stimato, la Juventus ha ottenuto 1.715 punti in meno di quanti ne ha meritati mentre il Napoli ne ha ottenuti 2.027 in più di quanti ne ha meritati.

Applicando questo approccio alle partite dell'intero campionato e successivamente sommando i risultati per ogni squadra si può ottenere una stima della classifica reale. Il confronto tra risultati reali e stimati è riportato in Tabella 4.1. Le posizioni che permettono la qualificazione alla Champions League sono colorate in blu, quelle all'Europa League in verde, mentre quelle che portano alla retrocessione in rosso. Possiamo quindi osservare una certa discordanza tra i risultati reali e i risultati stimati dal modello. Per esempio la Juventus, che nella realtà ha vinto il campionato con un distacco di 9 punti, è stimata come seconda in termini di *Expected Points*. Le tre squadre qualificate in Europa League secondo il modello sono diverse rispetto a quelle che si sono qualificate nella realtà. Mentre il Frosinone che nella realtà è arrivato ultimo risulta 13esimo per *Expected Points*.

	Risultati reali		Risultati stimati	
Posizione	Squadra	Punti	Squadra	xP
1	Juventus	91	Napoli	75.6
2	Napoli	82	Juventus	75.3
3	AS Roma	80	Fiorentina	67.3
4	Inter	67	AS Roma	66.2
5	Fiorentina	64	AC Milan	61.4
6	Sassuolo	61	Lazio	58.0
7	AC Milan	57	Torino	56.8
8	Lazio	54	Inter	56.8
9	Chievo	50	Sassuolo	54.0
10	Empoli	46	Atalanta	49.9
11	Genoa	46	Udinese	49.6
12	Torino	45	Genoa	49.1
13	Atalanta	45	Hellas Verona	47.0
14	Bologna	42	Carpi	46.2
15	Sampdoria	40	Chievo	44.5
16	Palermo	39	Empoli	43.0
17	Udinese	39	Sampdoria	39.9
18	Carpi	38	Bologna	39.0
19	Frosinone	31	Palermo	37.8
20	Hellas Verona	28	Frosinone	29.6

Tabella 4.1: Confronto tra classifica reale e stimata

Conclusioni

Da questa relazione è emerso che il metodo migliore per stimare gli *Expected Goals*, tra quelli provati, è l'approccio parametrico della regressione logistica, in particolare quello senza alcuna penalizzazione. Tutte le variabili esplicative analizzate sono risultate significative nelle analisi, sia nelle procedure regolarizzate che in quelle non.

La variabile esplicativa più importante è **distanza dalla porta**, come emerso dall'analisi degli alberi di classificazione e delle *Random Forest*. Inoltre, il modello di regressione logistica con solo distanza dalla porta e angolo di tiro come esplicative ha raggiunto un F-score inferiore di meno di un decimo inferiore rispetto al miglior modello.

I limiti degli *Expected Goals*

Nell'ultimo capitolo sono emerse delle discordanze tra le analisi effettuate con il modello e quanto accaduto nella realtà. Questo è dovuto in primo luogo ai limiti del modello stimato, il quale ha ottenuto un F-score di 0.424 e una specificità di poco superiore al 50%. Pertanto, il modello fatica a prevedere accuratamente i valori della variabile risposta, soprattutto per la classe dei goal.

Un secondo limite riguarda l'assunto di indipendenza, che, come già menzionato nel corso della relazione, non è sempre rispettato. Per migliorare questo aspetto, potrebbe essere opportuno includere nel modello delle variabili per valutare lo stato di forma delle squadre e dei calciatori. Anche l'aggiunta di un fattore temporale potrebbe portare un miglioramento. Infatti, capita spesso che in una stessa azione di gioco si realizzino più conclusioni, brevemente distanziate nel tempo; per queste non risulta ragionevole assumere l'indipendenza.

Delle ulteriori limitazioni sono legate ai dati a disposizione. Infatti i dati offerti sono raccolti da personale umano, quindi è ragionevole assumere che sia presente un termine di errore nel registrare i risultati e, soprattutto, nella stima della distanza dalla porta. Inoltre, è ipotizzabile che l'aggiunta di ulteriori variabili esplicative non disponibili tra quelle offerte gratuitamente possa apportare ulteriori miglioramenti al modello. Alcune

di queste potrebbero essere la posizione del portiere al momento del tiro, l'altezza a cui è stato colpito il pallone o il numero di difensori tra l'attaccante e la porta al momento della conclusione.

La penultima criticità riguarda l'associare la probabilità di segnare esclusivamente alla realizzazione di un tiro. Sebbene il leggendario ex giocatore e allenatore olandese Johan Cruijff abbia detto che *"You have got to shoot, otherwise, you can't score"* (Keel, 2016), questo non è sempre vero. I goal possono derivare da autogol della squadra avversaria o da situazioni che non si concretizzano in un tiro, e quindi corrispondono a 0 xG secondo i modelli visti finora, ma rappresentano comunque occasioni da goal.

Infine, è importante ricordare che queste discordanze tra gli *Expected Goals* e i risultati reali delle partite dipendono anche dalla natura del modello, in quanto gli *Expected Goals* sono nati soprattutto per valutare le prestazioni nelle partite di calcio, le quali spesso sono discordanti con il risultato reale delle stesse.

Possibili sviluppi

In letteratura sono stati esplorati diversi modelli per stimare gli *Expected Goals*. Ad esempio, in Rajagopalan e Srid (2023) è stato applicato un approccio di *Boosting* agli alberi di classificazione, ottenendo risultati migliori rispetto alle *Random Forest*.

Un ulteriore approccio che sta emergendo negli ultimi anni è quello basato sull'analisi delle immagini del momento in cui viene effettuato il tiro. Questo metodo sembra particolarmente promettente perché consente di superare il limite di associare gli *Expected Goals* esclusivamente alla realizzazione di un tiro. Inoltre, considerata la moltitudine di immagini registrate a fini televisivi durante le partite, esso può aiutare a superare il problema dell'errore umano nel registrare le variabili esplicative. Tuttavia, i modelli visti in questa relazione non sono adatti a questo tipo di analisi, in quanto cambia completamente la natura del dato a disposizione. Un esempio di questo approccio è presentato in Matteotti e Sotudeh (2024), dove è stato utilizzato un modello basato sulle *Convolutional Neural Networks* che ha ottenuto un AUC di 0.801.

Bibliografia

Bate, A. e Campbell, R. (2015). Expected goals explained: The analysis that is changing the game. URL: https://www.skysports.com/football/news/11661/10907419/expected-goals-explained-the-analysis-that-is-changing-the-game. (Ultimo accesso 25 Luglio 2024).

Breheny, P., Zeng, Y. e Kurth, R. (2021). grpreg: Regularization Paths for Regression Models with Grouped Covariates. URL: https://cran.r-project.org/web/packages/grpreg/index.html.

Breiman, L. (2001a). *Random Forest*. Springer.

Breiman, L. (2001b). Random forests. *Machine learning* **45**: 5–32.

Chawla, N. V., Bowyer, K. W., Hall, L. O. e Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research* **16**: 321–357.

Freeman, E. A. e Moisen, G. G. (2008). A comparison of the performance of threshold criteria for binary classification in terms of predicted prevalence and kappa. *Ecological modelling* **217**(1-2): 48–58.

Friedman, J., Hastie, T., Tibshirani, R., Narasimhan, B., Tay, K., Simon, N., Qian, J. e Yang, J. (2022). glmnet: Lasso and Elastic-Net Regularized Generalized Linear Models. URL: https://cran.r-project.org/web/packages/glmnet/index.html.

Gaughan, J. (2017). Arsenal boss Arsene Wenger claims Manchester City aren't "unstoppable" and blasts "atrocious" Michael Oliver (but has he got it very wrong on his "expected goals" claim?). URL: https://www.dailymail.co.uk/sport/football/article-5078067/

Arsene-Wenger-claims-Manchester-City-aren-t-unstoppable.html. (Ultimo accesso 25 Luglio 2024).

Grayson, J. (2012). Another post about TSR. URL: https://jameswgrayson.wordpress.com/2012/07/15/another-post-about-tsr. (Ultimo accesso 25 Luglio 2024).

Hastie, T., Tibshirani, R. e Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.

Hytner, D. (2014). Arsenal's 'secret' signing: club buys £2m revolutionary data company. URL: https://www.theguardian.com/football/2014/oct/17/arsenal-place-trust-arsene-wenger-army-statdna-data-analysts#:~:text=Arsenal%20bought%20StatDNA%2C%20the%20US,mention%20them%20by%20their%20name. (Ultimo accesso 25 Luglio 2024).

Keel, T. (2016). Johan Cruyff's best quotes: The game-changing wisdom of a true football legend. URL: https://www.eurosport.com/football/johan-cruyff-s-best-quotes-the-game-changing-wisdom-of-a-true-football-legend_sto5366190/story.shtml. (Ultimo accesso 27 Luglio 2024).

Kroese, D. P. e Rubinstein, R. Y. (2012). Monte Carlo methods. *Wiley Interdisciplinary Reviews: Computational Statistics* **4**(1): 48–58.

Lewis, M. (2003). *Moneyball: The Art of Winning an Unfair Game*. W. W. Norton & Company.

Lunardon, N., Menardi, G. e Torelli, N. (2022). Package 'ROSE'. URL: https://cran.r-project.org/web/packages/ROSE/ROSE.pdf.

Matteotti, M. e Sotudeh, H. (2024). The Power of Pixels: Exploring the Potential of CNNs for Expected Goals (xG) in Football. URL: https://www.researchgate.net/publication/382456974_The_Power_of_Pixels_Exploring_the_Potential_of_CNNs_for_Expected_Goals_xG_in_Football. (Ultimo accesso 27 Luglio 2024).

McMahon, B. (2012). The Most Important Soccer Performance Analyst You Have Never Heard Of. URL: https://www.forbes.com/sites/bobbymcmahon/2012/10/21/the-most-important-soccer-performance-analyst-you-have-never-heard-of. (Ultimo accesso 25 Luglio 2024).

Meier, L., Van De Geer, S. e Bühlmann, P. (2008). The group lasso for logistic regression. *Journal of the Royal Statistical Society Series B: Statistical Methodology* **70**(1): 53–71.

More, A. e Rana, D. P. (2017). Review of random forest classification techniques to resolve data imbalance. In *2017 1st International conference on intelligent systems and information management (ICISIM)*. IEEE, 72–78.

Perkins, N. J. e Schisterman, E. F. (2006). The inconsistency of "optimal" cut-points obtained using two criteria based on the receiver operating characteristic curve. *American journal of epidemiology* **163**(7): 670–675.

Pollard, R., Ensum, J. e Taylor, S. (2004). Estimating the probability of a shot resulting in a goal: The effects of distance, angle and space. *International Journal of Soccer and Science* **2**(1): 50–55.

Pollard, R. e Reep, C. (1997). Measuring the effectiveness of playing strategies at soccer. *Journal of the Royal Statistical Society Series D: The Statistician* **46**(4): 541–550.

Rajagopalan, A. e Srid, R. (2023). Football Performance Evaluation. URL: `https://www.researchgate.net/publication/373697963_Football_Performance_Evaluation`. (Ultimo accesso 28 Luglio 2024).

Reep, C. e Benjamin, B. (1968). Skill and chance in association football. *Journal of the Royal Statistical Society. Series A (General)* **131**(4): 581–585.

Robberechts, P. e Davis, J. (2020). How data availability affects the ability to learn good xG models. In *Machine Learning and Data Mining for Sports Analytics: 7th International Workshop, MLSA 2020, Co-located with ECML/PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings 7*. Springer, 17–27.

Salvan, A., Sartori, N. e Pace, L. (2020). *Modelli Lineari Generalizzati*. Springer.

Shelke, M. S., Deshmukh, P. R. e Shandilya, V. K. (2017). A review on imbalanced data handling using undersampling and oversampling technique. *International Journal of Recent Trends in Engineering and Research* **3**(4): 444–449.

Therneau, T. M. e Atkinson, E. J. (2023). An Introduction to Recursive Partitioning Using the RPART Routines. URL: `https://cran.r-project.org/web/packages/rpart/vignettes/longintro.pdf`.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology* **58**(1): 267–288.

Tippet, J. (2019). The Expected Goals Philosophy. ISBN: 978-1089883180.

Wang, L., Han, M., Li, X., Zhang, N. e Cheng, H. (2021). Review of classification methods on unbalanced data sets. *Ieee Access* **9**: 64606–64628.

Zhang, H. e Singer, B. H. (2010). *Recursive Partitioning and Applications*. Springer.

Zou, Q., Xie, S., Lin, Z., Wu, M. e Ju, Y. (2016). Finding the best classification threshold in imbalanced classification. *Big Data Research* **5**: 2–8.