UNIVERSITÀ DEGLI STUDI DELL'INSUBRIA

DIPARTIMENTO DI SCIENZE TEORICHE E APPLICATE

CORSO DI STUDIO TRIENNALE IN
INFORMATICA

Moneyball nel Calcio Contemporaneo: Analisi Big Data e Machine Learning per la definizione di modelli predittivi dei risultati di match calcistici.

Relatore:
Prof. Davide Tosi

Tesi di Laurea di:
Filippo Alzati
Matricola 745495

Anno accademico:
2022/2023

"Volere è potere”

Indice

Introduzione

Negli ultimi anni, il mondo dello sport ha subito una trasformazione epocale guidata da concetti innovativi e metodologie avanzate. La mia decisione di esplorare l'applicazione di analisi dati nel contesto del calcio contemporaneo è stata strettamente influenzata da una rivoluzionaria filosofia nata nel baseball professionistico: **Moneyball**.[1]

Moneyball, originariamente formulato da Billy Beane e Paul DePodesta presso gli Oakland Athletics, ha ridefinito le dinamiche tradizionali nella valutazione dei giocatori, introducendo un approccio basato su dati statistici avanzati. Questa filosofia ha dimostrato che l'utilizzo intelligente di dati può portare a decisioni strategiche più efficaci e a una migliore performance sportiva.

La mia ricerca mira a estendere queste innovazioni al **mondo del calcio**, dove l'analisi dei dati e il machine learning possono offrire nuove prospettive nella valutazione delle squadre, nella previsione degli esiti delle partite e nella pianificazione strategica. In questo contesto, Moneyball agisce come faro guida, ispirando l'adozione di metodologie avanzate per ottenere vantaggi competitivi nel calcio contemporaneo.[2]

‘’Lo scopo non deve essere comprare giocatori: lo scopo deve essere comprare vittorie’’ Peter Brand che interpreta Paul DePodesta in “Moneyball - L’Arte di Vincere”.

Obiettivi della Ricerca

Il nucleo centrale di questa ricerca si concentra sull'esplorare l'efficacia dell'analisi dei dati nel contesto del calcio, con l'obiettivo di determinare se tale approccio possa offrire previsioni accurate sull'esito delle partite. L'indagine si propone di analizzare come l'uso di strumenti, l'analisi statistica e il machine learning, possano influenzare la capacità di predizione nelle competizioni calcistiche.

Gli obiettivi specifici della ricerca comprendono:

Ricerca e Prelievo dei Dati tramite Web Scraping: La fase di ricerca e prelievo dati sarà una componente cruciale del nostro lavoro. Utilizzeremo tecniche di web scraping per acquisire informazioni dettagliate e aggiornate sulle squadre, i giocatori e le condizioni di gioco. L'accuratezza di questa fase influenzerà direttamente la validità del nostro modello predittivo.

Data Cleaning: Parallelamente al web scraping, dedicheremo attenzione al processo di data cleaning. La pulizia accurata dei dati è fondamentale per garantire che il nostro modello si basi su informazioni corrette e affidabili. Affronteremo sfide come la gestione di dati mancanti, la rimozione di duplicati e la normalizzazione delle variabili.

Valutare la Rilevanza delle Variabili: Esaminare attentamente la rilevanza delle variabili selezionate nell'ambito dell'analisi dei dati per identificare i fattori che hanno un impatto significativo sull'esito di una partita di calcio.

Sviluppare un Modello Predittivo: Creare un modello predittivo basato sull'analisi dei dati e su algoritmi di machine learning, mirando a prevedere con precisione l'esito di una partita calcistica.

Testare l'Affidabilità del Modello: Sottoporre il modello predittivo a una verifica e validazione, testando la sua affidabilità e precisione attraverso dati storici e risultati effettivi delle partite.

Esplorare Potenziali Sfide e Limitazioni: Identificare e analizzare possibili sfide e limitazioni connesse all'utilizzo dell'analisi dei dati nel contesto del calcio, fornendo un'analisi critica delle restrizioni potenziali del metodo.

Proporre Possibili Miglioramenti: Sulla base delle conclusioni raggiunte, suggerire potenziali miglioramenti o sviluppi futuri nell'applicazione dell'analisi dei dati per la previsione degli esiti delle partite di calcio.

Questi obiettivi mirano a offrire una visione completa e ben argomentata sull'efficacia dell'analisi dei dati nel contesto del calcio, contribuendo a consolidare la comprensione di come l'innovazione tecnologica possa influenzare l'interpretazione e la previsione degli eventi sportivi.

Capitolo 1

Panoramica sull'Analisi dei Dati

L'analisi dei dati assume un ruolo cruciale nell'attuale era digitale, delineando un processo intricato che comprende l'esplorazione, la pulizia, l'interpretazione e la presentazione dei dati per ottenere informazioni significative.
Questa pratica ha acquisito crescente importanza a causa della sempre maggiore disponibilità di dati e delle avanzate tecnologie informatiche che permettono di scoprire significati e modelli utili. In un mondo in cui le informazioni abbondano, l'analisi dei dati diventa una chiave per estrarre conoscenze e prendere decisioni.

Cos'è l'Analisi dei Dati

L'analisi dei dati non è semplicemente un processo tecnico; è una scienza che coinvolge la trasformazione di dati grezzi in una narrazione significativa. Attraverso l'applicazione di metodologie statistiche, algoritmi di machine learning e sofisticati strumenti di visualizzazione, l'analisi dei dati si propone di rivelare insight nascosti e prendere decisioni migliori.

Big Data

I "*big data"* sono insiemi enormi di informazioni così grandi e complessi che diventa difficile gestirli e capirli usando i metodi tradizionali. Questi dati, che

spesso includono una grande quantità di informazioni diverse e arrivano molto velocemente, richiedono nuovi modi di essere trattati.
Con la diffusione di dispositivi digitali e tecnologie come l'IOT[1], stiamo generando sempre più dati in tempo reale.

Analizzare i big data significa non solo occuparsi di questa grande quantità di informazioni, ma anche cercare di trovare modelli e informazioni interessanti all'interno di esse. Questo modo di gestire i dati ha cambiato molte cose, come la gestione delle imprese, la ricerca scientifica e la sanità e si sta insinuando anche nel mondo dello sport. L'obiettivo è prendere decisioni più sagge e prevedere i problemi prima che si presentino. La sfida ora è capire come usare al meglio questi big data in modo innovativo e al tempo stesso proteggere la nostra privacy e garantire la sicurezza delle informazioni.[3]

Figure Professionali nell'Analisi dei Dati

1. Data Analyst

Il Data Analyst si colloca al crocevia tra dati grezzi e informazioni significative. Attraverso competenze statistiche avanzate e una profonda comprensione degli obiettivi aziendali, il Data Analyst traduce numeri complessi in report comprensibili. Il suo ruolo è fondamentale per fornire un quadro chiaro e dettagliato della situazione attraverso l'analisi approfondita dei dati.

[1] Internet of Things (IoT): L'Internet delle Cose è un concetto che si riferisce alla connessione di dispositivi fisici alla rete Internet, consentendo loro di comunicare e scambiare dati tra loro. Questa interconnessione include una vasta gamma di oggetti, come elettrodomestici, veicoli, sensori e dispositivi industriali, che possono raccogliere e condividere informazioni.1

2. Data Scientist

Il Data Scientist è il narratore delle storie nascoste nei dati. Con una formazione multidisciplinare che comprende statistica, programmazione e conoscenza del dominio, questo professionista crea modelli predittivi e algoritmi avanzati. La sua missione è quella di scoprire nuove prospettive e fornire insight strategici che guidano le decisioni aziendali.

3. Data Engineer

Mentre il Data Analyst e il Data Scientist si concentrano sulla comprensione e sull'utilizzo dei dati, il Data Engineer è l'architetto che costruisce le fondamenta. Responsabile della progettazione e dello sviluppo di sistemi di gestione dati robusti, il Data Engineer garantisce che le informazioni siano facilmente accessibili e gestite in modo sicuro.

Interazioni tra Professioni

La collaborazione tra Data Analyst, Data Scientist e Data Engineer è sinergica. Il Data Analyst fornisce il contesto e la comprensione iniziale, il Data Scientist crea modelli avanzati e il Data Engineer costruisce l'infrastruttura che li supporta. Insieme, formano un team in grado di affrontare sfide complesse di analisi dei dati.

Impatto e Crescita dell'Analisi dei Dati

L'analisi dei dati ha rivoluzionato la nostra capacità di prendere decisioni informate in una vasta gamma di settori. Dal miglioramento delle operazioni aziendali alla previsione delle tendenze di mercato, il suo impatto è onnipresente. Questo capitolo si conclude riflettendo sulla crescita esplosiva dell'analisi dei dati e anticipando il suo ruolo cruciale nell'ambito della previsione degli esiti delle partite di calcio.

Capitolo 2

Machine Learning

Il Machine Learning (ML) è una branca dell'intelligenza artificiale che si occupa di sviluppare algoritmi e modelli in grado di far apprendere ai computer senza essere esplicitamente programmati. In altre parole, il Machine Learning consente alle macchine di acquisire conoscenza e migliorare le loro prestazioni attraverso l'esperienza e l'analisi dei dati.

L'approccio fondamentale del Machine Learning si discosta dal tradizionale paradigma di programmazione, dove gli sviluppatori definiscono regole e istruzioni specifiche per risolvere un determinato problema. Al contrario, nel Machine Learning, si forniscono ai computer dati di input e si permette loro di imparare autonomamente i modelli sottostanti, identificando pattern, relazioni e tendenze nei dati stessi.[4][5]

Esistono diverse categorie di modelli di ML, vediamone due nel dettaglio.

Apprendimento Supervisionato:

Nel modello di apprendimento supervisionato, il processo di addestramento coinvolge l'utilizzo di un dataset che contiene esempi di input e le corrispondenti etichette di output desiderate.

L'obiettivo principale è far apprendere al modello la relazione tra gli input e gli output in modo che, una volta addestrato, possa fare previsioni accurate su nuovi dati.[6]

Addestramento: Durante l'addestramento, il modello riceve coppie di input e output desiderato. Ad esempio, se stiamo cercando di creare un modello per riconoscere immagini di gatti, il dataset conterrà immagini di gatti insieme all'etichetta "gatto".

Apprendimento: Il modello cerca pattern nei dati che possano collegare gli input alle etichette di output. Questo processo di apprendimento consente al modello di generalizzare e fare previsioni accurate su nuovi dati, anche se non sono stati visti durante l'addestramento.

Previsione: Una volta addestrato, il modello può essere utilizzato per fare previsioni su nuovi dati, producendo un output previsto basato sugli input forniti.

Tipi di Modelli Supervisionati:

Ci sono diversi tipi di modelli supervisionati, ognuno dei quali si adatta a contesti e tipi di dati specifici. Alcuni dei modelli più comuni includono:

Regressione Lineare:

Utilizzato per predire un valore continuo.

Adatto quando c'è una relazione lineare tra le features e l'output desiderato.

Supponiamo di voler esplorare la relazione tra le dimensioni delle case e i relativi prezzi per sviluppare un modello che possa stimare il prezzo di una casa in base alla sua dimensione. Questo è un problema comune in statistica e machine learning, dove cerchiamo di adattare una linea ai dati in modo da poter fare previsioni su nuovi dati.

Nel nostro esercizio, abbiamo raccolto dati su diverse case, dove la dimensione delle case è rappresentata da '**X**' e i prezzi delle case sono rappresentati da '**y**'. Ora, utilizzando la libreria scikit-learn[2] in Python, vogliamo addestrare un modello di regressione lineare su questi dati, in modo da poter prevedere il prezzo di una casa in base alle sue dimensioni. Andiamo avanti con l'addestramento del modello e l'analisi dei risultati.

[2] Scikit-learn è una libreria Python open-source per il machine learning, ampiamente utilizzata per la sua semplicità ed efficacia. Offre strumenti per la preparazione dei dati, la creazione di modelli e la valutazione delle prestazioni, rendendola una risorsa essenziale per chiunque si occupi di apprendimento automatico.

```
# Suddividi i dati in set di addestramento e test
X_train, X_test, y_train, y_test = train_test_split(X, y,
    test_size=0.2, random_state=42)

# Crea un modello di regressione lineare
model = LinearRegression()

# Addestra il modello sui dati di addestramento
model.fit(X_train, y_train)
```

fig.1 : Script addestramento con regressione lineare

test_size = 0.2 specifica che il 20% dei dati verrà utilizzato come set di test, e ***random_state*** garantisce la riproducibilità della suddivisione dei dati.

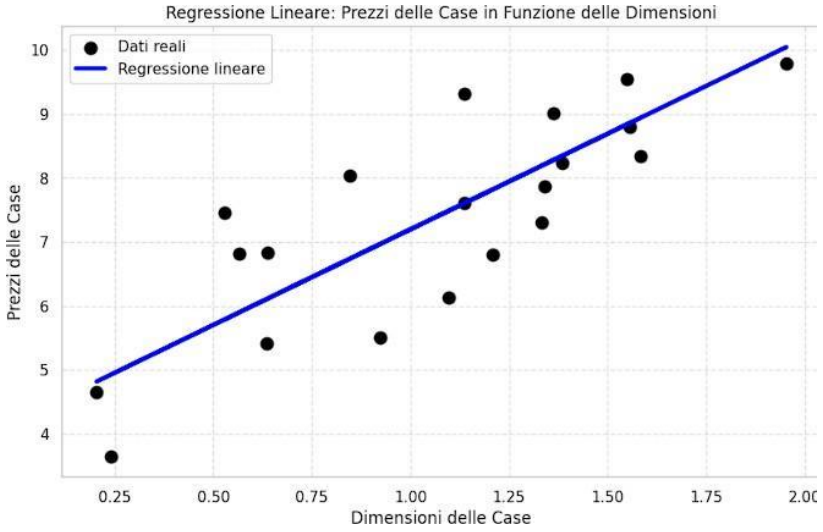

fig.2 : Grafico regressione lineare

Nel grafico, la dispersione dei dati attorno alla linea di regressione indica quanto il modello si adatti ai dati reali. Se la dispersione è bassa e la linea segue bene i punti, possiamo concludere che la regressione lineare è un modello efficace per questo particolare insieme di dati.

Regressione Logistica:

Utilizzato per problemi di classificazione binaria.
Produce un output tra **0** e **1**, interpretato come probabilità.

Supponiamo di avere un dataset contenente informazioni su studenti universitari e vogliamo costruire un modello per prevedere se uno studente passerà o no un esame in base al numero di ore di studio. Utilizzeremo la regressione logistica per affrontare questo problema di classificazione binaria.

```
# Suddividi i dati in set di addestramento e test
X_train, X_test, y_train, y_test = train_test_split(X, y,
    test_size=0.2, random_state=42)

# Crea un modello di regressione logistica
logistic_reg = LogisticRegression()

# Addestra il modello sui dati di addestramento
logistic_reg.fit(X_train, y_train)
```

fig.3 : Script addestramento con regressione logistica

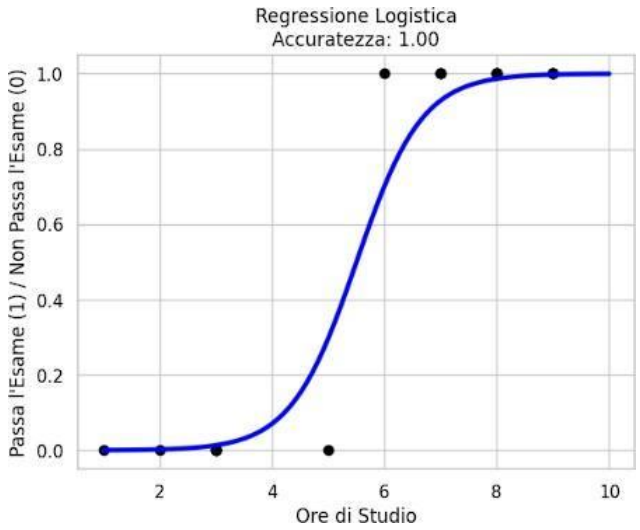

fig.4 : Grafico regressione logistica

La "**decision boundary**" (confine decisionale) in un problema di classificazione con la regressione logistica rappresenta il confine tra le diverse classi predette dal modello. In un contesto binario come l'esempio sopra (superare o non superare l'esame), la decision boundary separa lo spazio delle caratteristiche in due regioni: una in cui il modello predice la classe positiva (1) e un'altra in cui predice la classe negativa (0).

Nel grafico, la decision boundary è rappresentata dalla linea blu. Ogni punto sulla linea indica un punto nello spazio delle caratteristiche in cui la probabilità di appartenenza a una delle classi è del 50%. Oltre la linea blu, il modello predice la classe 1 (superare l'esame), mentre al di sotto della linea predice la classe 0 (non superare l'esame).

Support Vector Machines (SVM):

Utilizzati per problemi di classificazione o regressione.

Cercano di trovare un iperpiano ottimale per separare i dati.

In questo esempio, i dati sono generati casualmente, e l'SVM cerca di separarli con un iperpiano lineare. I support vectors sono evidenziati, e l'iperpiano è tracciato nel grafico.

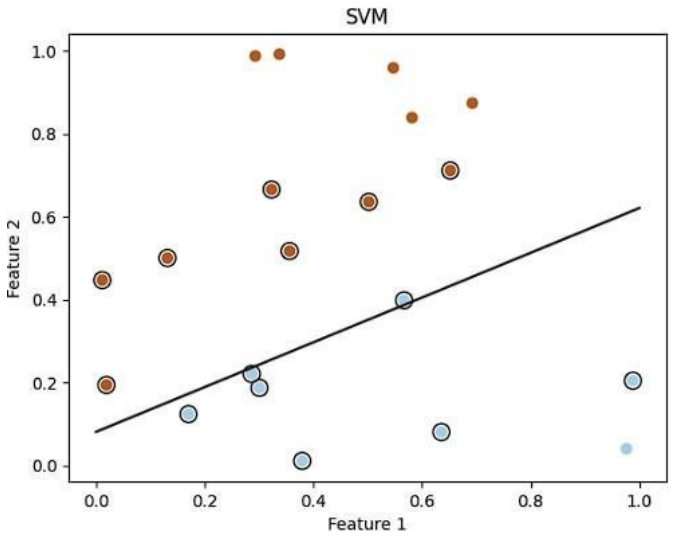

fig.5 : Grafico SVM

I punti blu e rossi rappresentano i dati generati casualmente. Ogni punto ha due coordinate (Feature 1 e Feature 2) e un colore associato (0 o 1) che indica la classe a cui appartiene.

L'obiettivo dell'SVM è trovare un iperpiano che separi al meglio le due classi. In questo caso, essendo un iperpiano lineare, si tratta di una retta.

L'iperpiano è rappresentato dalla linea nera nel grafico. Questa linea cerca di separare i punti blu da quelli rossi. L'equazione della retta è ottenuta in base ai pesi e al termine noto dell'iperpiano calcolati dal modello SVM.

I punti più vicini all'iperpiano e che contribuiscono alla definizione della sua posizione sono chiamati **support vectors**.

Alberi Decisionali e Random Forest:

Alberi decisionali sono strutture ad albero che guidano le decisioni.[7]

Random forest è un insieme di alberi decisionali.

Adatti per problemi di classificazione e regressione.

Supponiamo di avere un dataset che contiene informazioni su diverse caratteristiche fisiche e abitudini alimentari di un gruppo di persone, insieme a un'etichetta che indica se ciascuna persona è in buona o cattiva salute.

```
# Suddividi i dati in set di addestramento e test
X_train, X_test, y_train, y_test = train_test_split(X, y,
    test_size=0.2, random_state=42)

# Creare e addestrare il modello di Random Forest
model = RandomForestClassifier(n_estimators=100,
    random_state=42)

# Addestra il modello sui dati di addestramento
model.fit(X_train, y_train)
```

fig.6 : Script addestramento random forest

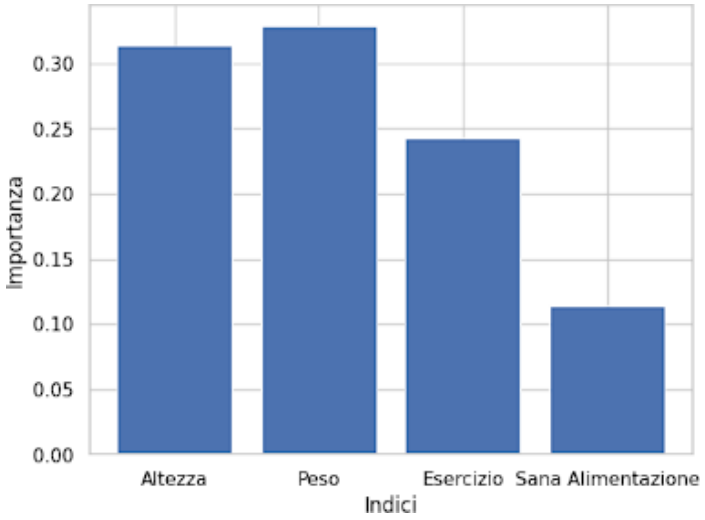

fig.7 : Grafico peso metriche

In questo esercizio, stiamo creando un modello di Random Forest per prevedere lo stato di salute delle persone in base a diverse caratteristiche. Il codice include la creazione di un dataset **fittizio**, la suddivisione in set di training e test, la creazione del modello, la valutazione delle prestazioni e la visualizzazione dell'importanza delle features. Si tratta di un esempio semplificato, e nei dati reali **si dovrebbero utilizzare set di dati più ampi e rappresentativi**.

Reti Neurali:

Le Reti Neurali Artificiali (ANN) sono componenti interconnesse (neuroni) che trasformano un insieme di input in un output desiderato, cercando di imitare una rete neurale biologica.

Purucker fu uno dei primi a studiare la previsione dei risultati nelle partite sportive utilizzando ANNs. Ha raccolto dati dalle prime otto giornate della National Football League (NFL), inclusi cinque fattori: yards guadagnati,

yards corse, margine di palla persa, tempo di possesso e quote di scommesse. Ha utilizzato un Percettrone Multistrato (**MLP**)[3] ANN addestrato con l'**algoritmo di retropropagazione**[4] e ha raggiunto un'accuratezza del 61% rispetto all'accuratezza del 72% degli esperti del settore.[8][9]

K-Nearest Neighbors (K-NN):

L'idea di base di KNN è piuttosto intuitiva. Immagina di avere un insieme di dati con etichette di classe associate e un nuovo punto di dati di cui vuoi predire l'etichetta di classe. KNN fa quanto segue:

Calcolo della Distanza: Calcola la distanza tra il nuovo punto di dati e tutti i punti di addestramento. La distanza può essere calcolata utilizzando diverse metriche, come la distanza euclidea o la distanza di Manhattan[5].

Selezione dei Vicini: Identifica i "K" punti di addestramento più vicini al nuovo punto di dati. "K" è un parametro che deve essere specificato in anticipo e rappresenta il numero di vicini da considerare.

---

[3] Percettrone: È un tipo di rete neurale semplificato, basato sul neurone artificiale, che può prendere più input, assegnare loro dei pesi e produrre un output. Un singolo strato di questi neuroni costituisce il percettrone.
Multistrato: Indica che la rete è composta da più strati di percettroni. Gli strati includono uno strato di input, uno o più strati nascosti e uno strato di output. L'aggiunta di strati nascosti consente alla rete di apprendere rappresentazioni più complesse.

[4] È un metodo di addestramento per reti neurali che si basa sulla minimizzazione dell'errore tra le previsioni della rete e gli output desiderati. La retropropagazione calcola l'errore e lo propaga all'indietro attraverso la rete, aggiornando i pesi in modo che l'errore sia ridotto.

[5] La distanza di Manhattan deve il suo nome al layout rettilineo delle strade a Manhattan. Questa distanza è la somma delle differenze assolute tra le coordinate dei punti $d = |x_2 - x_1| + |y_2 - y_1|$

Voto a Maggioranza: Per la classificazione, determina l'etichetta di classe più comune tra i K vicini (voto a maggioranza). Per la regressione, calcola la media dei valori delle etichette dei K vicini.

Assegna l'Etichetta: Assegna al nuovo punto di dati l'etichetta di classe determinata dalla votazione a maggioranza.

L'aspetto cruciale di KNN è la scelta di "K". Un valore di "K" troppo piccolo potrebbe rendere il modello sensibile al rumore nei dati, mentre un valore troppo grande potrebbe rendere il modello meno flessibile.

Apprendimento Non Supervisionato:

Nel modello di apprendimento non supervisionato, il dataset di addestramento non contiene etichette di output. L'obiettivo principale è far emergere pattern o strutture nascoste nei dati senza indicazioni specifiche su cosa cercare.

Addestramento: Il modello riceve solo gli input, senza informazioni sul risultato desiderato. Ad esempio, se stiamo analizzando dati di vendite, il modello riceverà solo informazioni sui prodotti venduti, senza indicazioni su quali prodotti siano di successo o meno.

Apprendimento: Il modello cerca automaticamente pattern o cluster nei dati. Questo tipo di apprendimento è spesso utilizzato per scoprire relazioni intrinseche, raggruppamenti naturali o strutture nascoste.

Analisi dei Dati: Una volta addestrato, il modello può essere utilizzato per analizzare e comprendere la struttura intrinseca dei dati senza una guida specifica.

Per concludere, l'apprendimento supervisionato richiede esempi etichettati per addestrare il modello a fare previsioni, mentre l'apprendimento non supervisionato cerca pattern o strutture senza etichette specifiche di output.

Tipi di Modelli Non Supervisionati

K-Means Clustering:

Raggruppa i dati in cluster basati sulla similarità.

Supponiamo di lavorare per un'applicazione dedicata al calcio, e vogliamo utilizzare K-Means per raggruppare gli utenti in base alle loro preferenze e interazioni con l'app. I dati potrebbero includere il numero di partite seguite, il tempo trascorso sulle sezioni dedicate alle squadre preferite, e il tipo di contenuti preferito (ad esempio, notizie, statistiche, video degli highlights).

Implementiamo l'elbow method per determinare il numero ottimale di cluster (k) da utilizzare in un modello di clustering K-means.

Nell'output del grafico, si cerca un punto in cui la diminuzione dell'errore rallenta significativamente, formando un "gomito" nel grafico.

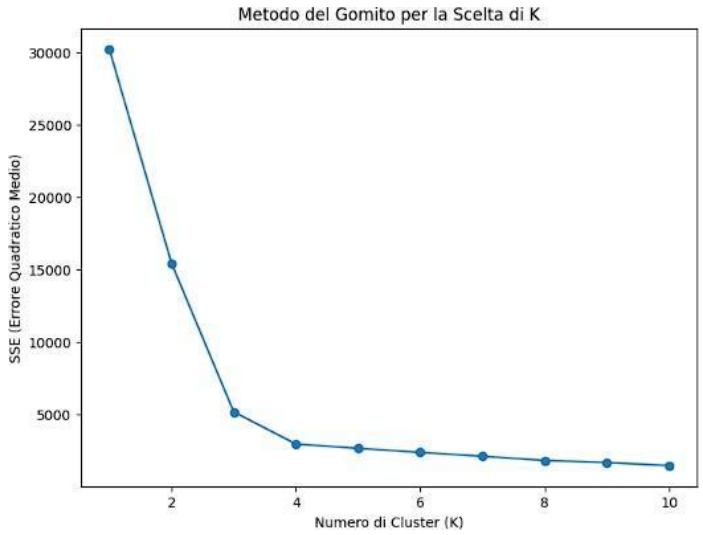

fig.8 : Grafico elbow method

Come si nota nel grafico, il valore individuato di ***K=3*** corrisponde al punto in cui la curva dell'Errore Quadratico Medio (SSE) mostra un chiaro gomito.

Applichiamo ora il modello:

```
# Addestramento del modello K-Means con K=3
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans.fit(X)

# Otteniamo le etichette di cluster e i centri dei cluster
labels = kmeans.labels_
centers = kmeans.cluster_centers_
```

fig.9 : Script addestramento K-means

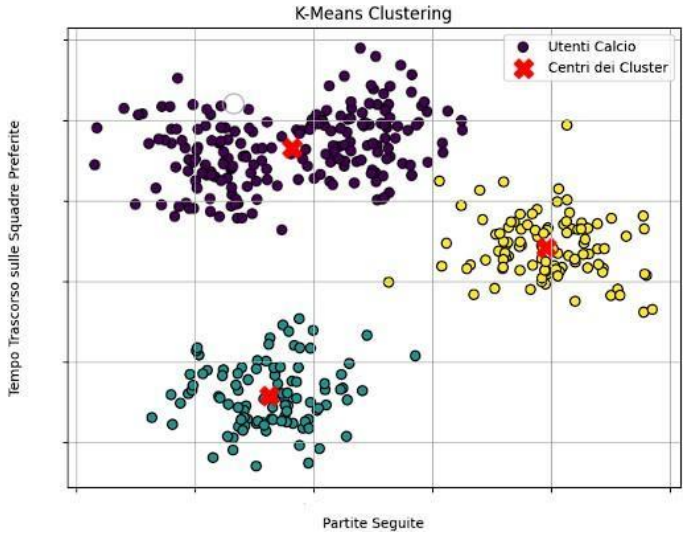

fig.10 : Grafico K-means clustering

In questo esempio, i dati casuali rappresentano il coinvolgimento degli utenti nell'app, con il numero di partite seguite e il tempo trascorso sulle squadre preferite come feature. I cluster formati da **K-Means** potrebbero identificare gruppi di utenti con comportamenti simili, e i centri dei cluster potrebbero rappresentare profili tipici di interazione con l'app per ciascun gruppo. Queste informazioni potrebbero essere utilizzate per personalizzare le notizie, le notifiche o i contenuti multimediali sulla piattaforma di calcio.

Analisi delle Componenti Principali (PCA):

La **PCA** (Analisi delle Componenti Principali) è una potente tecnica di riduzione della dimensionalità utilizzata nell'ambito dell'analisi dei dati e del machine learning.

Quando affrontiamo un dataset con un gran numero di variabili o feature, la PCA ci aiuta a focalizzarci sulle informazioni più significative attraverso la ricerca di nuove variabili, chiamate "componenti principali". Queste componenti principali sono combinazioni lineari delle variabili originali e sono ordinate in modo decrescente in base alla quantità di varianza che spiegano nei dati.

Capitolo 3

Moneyball

Prima di addentrarci nel nostro campo di utilizzo, ossia il calcio, è bene capire dove nasce questo approccio.

Moneyball è stato introdotto dal general manager degli Oakland Athletics, Billy Beane, negli anni 2000. L'obiettivo principale era trovare modi più efficienti ed economici per assemblare una squadra competitiva, considerando le limitate risorse finanziarie della squadra.

L'innovazione chiave di Moneyball nel baseball è stata l'uso intensivo delle statistiche avanzate e l'analisi dei dati per valutare le prestazioni dei giocatori in modo più accurato rispetto alle valutazioni tradizionali.

Beane e il suo team hanno introdotto metriche che misurano la capacità di un giocatore di raggiungere una base, e altri indicatori statistici, per valutare il vero contributo di un giocatore alla squadra.

Moneyball ha sfidato le convenzioni del reclutamento dei giocatori, consentendo agli Oakland Athletics di selezionare talenti trascurati da altre squadre ma che mostravano prestazioni elevate secondo le nuove metriche analitiche.[10]

L'approccio ha portato gli Oakland Athletics a ottenere risultati sorprendenti

e ha influenzato significativamente il modo in cui molte squadre sportive, di baseball e non, considerano e utilizzano l'analisi dei dati per prendere decisioni.

Come funziona Moneyball

Selezione delle Statistiche Chiave:

Nel contesto del metodo Moneyball, il processo inizia con un'attenta individuazione delle statistiche più significative per valutare le prestazioni di un giocatore. Queste nuove metriche avanzate vengono spesso preferite alle tradizionali, come la media battuta o i fuoricampo, poiché forniscono una visione più precisa delle abilità di un giocatore. Il sistema analizza in profondità i dati, considerando non solo le statistiche e le caratteristiche fisiche dei giocatori, ma anche la loro capacità di adattarsi a un determinato contesto tattico e ambientale.

Identificazione dei Giocatori Sottovalutati:

Un elemento cardine del Moneyball consiste nell'individuare giocatori spesso trascurati dal mercato, ma con un potenziale significativo. Questi atleti, sebbene meno conosciuti o non vincolati da contratti costosi, rivelano il loro vero valore attraverso statistiche avanzate. L'analisi approfondita consente di scovare queste *"hidden gems"* nel vasto panorama del baseball.

Costruzione del Roster:

Gli Oakland Athletics, guidati da Billy Beane, mettono in pratica queste analisi nella selezione dei giocatori per il loro roster. La squadra si concentra su atleti che, in base alle statistiche chiave, offrono un eccellente rapporto tra il loro valore e il costo associato. Questo approccio consente loro di massimizzare le risorse disponibili, creando un roster che ottimizza il valore complessivo. La gestione della squadra si impegna nel bilanciare giocatori con eccellenza in diverse abilità, come il lancio, la difesa e la capacità di avanzare sulle basi.

Flessibilità Tattica:

Moneyball non si limita alla selezione dei giocatori ma implica anche l'adozione di una strategia di gioco basata sui dati. Gli Athletics mirano a guadagnare basi attraverso un approccio di lancio paziente, la capacità di raggiungere le basi e una difesa solida. Questa strategia cerca di sfruttare appieno le abilità dei giocatori selezionati, creando opportunità per segnare punti e vincere partite.

Risultati:

Nonostante il budget limitato, gli Athletics hanno dimostrato il successo dell'approccio Moneyball raggiungendo i playoff in diverse occasioni. Ciò sottolinea la possibilità di competere con squadre dotate di risorse finanziarie notevolmente superiori, grazie all'utilizzo intelligente di analisi

avanzate basate sui dati.

La Sabermetrica

Il baseball ha guadagnato popolarità in diversi paesi, e negli ultimi decenni, l'analisi statistica nel baseball, conosciuta come sabermetrica, ha rivoluzionato l'approccio valutativo nei confronti di giocatori e squadre. Questa rivoluzione è stata avviata da Bill James negli anni '70, un appassionato senza esperienza professionale nel baseball, che ha coniato il termine *"sabermetrics"* per descrivere l'analisi dettagliata del gioco basata su dati di prestazioni anziché sulle statistiche tradizionali come la media battuta.

Le statistiche sabermetriche utilizzano misure più avanzate, fornendo una visione più accurata delle prestazioni di un giocatore rispetto alle statistiche tradizionali. Anche se non sono comunemente presenti nei tabellini di gioco, queste metriche sono utilizzate dagli esperti per valutare i giocatori in modo più approfondito, contribuendo a cambiare la percezione del valore dei giocatori e a migliorare la comprensione del baseball.

Billy Beane e la sua dirigenza focalizzata sulla sabermetrica hanno sostenuto statistiche come la percentuale di arrivo in base (**OBP**) e la percentuale di slugging (**SP**) perché credevano che queste misure rappresentassero i due componenti essenziali della creazione dell'offesa.

Queste statistiche tengono conto di vari eventi, tra cui basi su ball (**BB**)[6], colpi da lancio (**HB**)[7], e situazioni specifiche come i *"sacrifice fly"* (**SF**)[8].

Una statistica tradizionale spesso considerata dai sabermetrici è quella delle basi rubate, anche se sostengono che questa misura può distorcere il vero valore di un giocatore. In alternativa, sono state sviluppate statistiche più complesse come Fielding Runs Above Average (**FRAA**) e Ultimate Zone Rating (**UZR**) per valutare le abilità difensive di un giocatore in modo più accurato, considerando fattori come le dimensioni e la forma del parco da gioco.

Nel contesto della prestazione del lancio, statistiche come **WHIP** (valore delle basi per inning lanciato) e **K/9** (strikeout per nove inning) sono diventate cruciali nell'analisi sabermetrica. Queste metriche consentono di valutare le abilità di un lanciatore in modo più obiettivo, senza dipendere dalle circostanze di gioco come le vittorie, che possono essere influenzate dalla capacità offensiva della squadra.

In sintesi, l'analisi sabermetrica ha apportato un cambiamento significativo nell'approccio al baseball, fornendo strumenti più sofisticati per valutare le prestazioni individuali e di squadra.[11]

---

[6] Si verifica una base su ball quando il lanciatore effettua quattro lanci in zona non valida di strike allo stesso battitore, senza che questi metta in gioco la palla con una battuta o venga eliminato con uno strikeout.

[7] Il *"hit by pitch"* (HBP) è un evento in cui un battitore o i suoi vestiti / attrezzature (diverse dalla mazza) sono colpiti direttamente da un lancio del lanciatore.

[8] Manovra tattica che consente alla squadra in battuta di realizzare un punto, facendo arrivare a casa un corridore che si trova in base. È detta *"di sacrificio"* perché il battitore sacrifica il proprio turno di battuta,

facendosi eliminare *"al volo"*, per consentire la realizzazione del punto alla propria squadra.

Capitolo 4

Introduzione al progetto : Metriche

Per poter applicare un modello di ML sul calcio sarebbe prima necessario individuare delle **KPI** andando a definire quella che potrebbe essere la Sabermetrica ma per il mondo calcistico. Gli indicatori chiave di prestazione (KPI) sono un insieme di indicatori misurabili che svolgono un ruolo vitale nel capire quanto un sistema stia effettivamente raggiungendo i suoi obiettivi. Questi indicatori sono molto importanti per monitorare e valutare le prestazioni di un sistema, organizzazione, azienda o brand.

I club di calcio definiscono indicatori chiave di prestazione a diversi livelli per misurare il loro successo nel raggiungere gli obiettivi. A livello elevato, si concentrano sulla performance complessiva del club, mentre gli indicatori a livello inferiore si focalizzano sui processi nei dipartimenti come addestramento, marketing o logistica.

Metriche utilizzate:

Nella fase iniziale del nostro progetto, riconosciamo l'importanza di identificatori specifici nel modellare in modo preciso le prestazioni calcistiche.

Maggiore è il numero di identificatori che integriamo nel nostro modello,

maggiore sarà la sua precisione. Questo concetto è parallelo alla qualità dei dati raccolti, che rappresentano un aspetto fondamentale nella costruzione di un modello affidabile.

Per la nostra analisi, abbiamo estratto dati da ogni partita per ogni squadra presente dalla stagione 2019/2020 fino a quella 2023/2024, nella **Serie A** italiana. Questi dati, prelevati accuratamente da **fbref.com**[9], sono così suddivisi:

- ***GF*** = Gol fatti
- ***GA*** = Gol subiti
- ***Sh*** = Tiri totali
- ***SoT*** = Tiri in porta
- ***Dist*** = Distanza media dei tiri
- ***FK*** = Punizioni
- ***PK*** = Rigori segnati
- ***PKatt*** = Rigori calciati
- ***Cmp%*** = Percentuale passaggi andati a buon fine
- ***CrsPA*** = Cross in area
- ***Poss*** = Possesso palla

identificatori temporali e di contesto:

- ***venue_code*** (1 per la squadra di casa, 0 per gli ospiti)
- ***opp_code*** (Codice identificativo per ogni squadra in trasferta)
- ***hour*** (Orario di Gioco)

[9] FBref è un sito web dedicato alle statistiche e analisi avanzate nel mondo del calcio. Siti di questo tipo

forniscono dettagliate statistiche sui giocatori, squadre e competizioni

- ***day_code*** (Giorno)

Questi dati, provenienti da fbref.com e focalizzati sulla Serie A, forniranno la base per la creazione di un modello di machine learning mirato a prevedere i risultati delle partite calcistiche.

Con il progresso del progetto, esploreremo ulteriori identificatori e ottimizzeremo il modello per migliorare la sua capacità predittiva.

Fase 1 : Prelievo dei dati

Per la fase iniziale abbiamo sviluppato uno script dedicato allo scraping delle metriche dalla pagina web di fbref.com. Questo script, scritto in **Python**, è progettato per navigare la struttura complessa del sito web, identificando gli elementi HTML chiave estraendo le metriche calcistiche desiderate.

Durante lo sviluppo dello script, abbiamo utilizzato librerie fondamentali come BeautifulSoup[12] per effettuare il parsing dell'HTML e navigare attraverso la pagina.

Inoltre, è stata necessaria una logica per il salvataggio ordinato dei dati estratti in un file **CSV**[10], rendendo le informazioni facilmente accessibili per fasi successive del progetto.

[10] Un file CSV (Comma-Separated Values) è un formato di file che viene utilizzato per rappresentare dati tabulari sotto forma di testo semplice. In un file CSV, i dati sono organizzati in colonne separate da virgole (o da un altro delimitatore, come il punto e virgola). Ogni riga del file rappresenta una record o un'istanza di dati.

Lo script è progettato per iterare su più squadre, ottenendo i dati delle partite per ciascuna squadra in ogni stagione. L'iterazione su squadre e stagioni si svolge in modo sequenziale, con una transizione alle stagioni precedenti dopo aver completato l'iterazione per tutte le squadre nella stagione corrente.

Dopodichè si effettua un **raggruppamento** per squadra in modo da avere una tabella ordinata.

Concludendo questa fase, lo script di scraping è ora in grado di raccogliere periodicamente dati aggiornati da fbref.com, fornendo una solida base di informazioni per l'analisi successiva e la costruzione del modello di machine learning.

Ecco l'output ottenuto:

	Date	Time	Comp	Round	Day	Venue	Result	GF	GA	Opponent	...	Sh	SoT	Dist	FK	PK	PKatt	Cmp%	CrsPA	Season	Team
0	2023-08-19	20:45	Serie A	Matchweek 1	Sat	Home	W	2.0	0.0	Monza	...	22.0	3.0	17.2	1.0	0	0	88.0	3.0	2024	Internazionale
1	2023-08-28	20:45	Serie A	Matchweek 2	Mon	Away	W	2.0	0.0	Cagliari	...	17.0	3.0	15.9	0.0	0	0	84.4	5.0	2024	Internazionale
2	2023-09-03	18:30	Serie A	Matchweek 3	Sun	Home	W	4.0	0.0	Fiorentina	...	20.0	10.0	14.5	1.0	1	1	81.7	4.0	2024	Internazionale
3	2023-09-16	18:00	Serie A	Matchweek 4	Sat	Home	W	5.0	1.0	Milan	...	13.0	6.0	15.2	1.0	1	1	80.9	3.0	2024	Internazionale
5	2023-09-24	12:30	Serie A	Matchweek 5	Sun	Away	W	1.0	0.0	Empoli	...	23.0	5.0	16.8	0.0	0	0	83.3	6.0	2024	Internazionale
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
36	2020-07-19	19:30	Serie A	Matchweek 34	Sun	Away	L	1	2	Brescia	...	15.0	6.0	14.8	1.0	0	0	78.4	0.0	2020	SPAL
37	2020-07-22	21:45	Serie A	Matchweek 35	Wed	Home	L	1	6	Roma	...	11.0	3.0	17.8	2.0	0	0	84.8	0.0	2020	SPAL
38	2020-07-26	19:30	Serie A	Matchweek 36	Sun	Home	D	1	1	Torino	...	7.0	1.0	18.2	0.0	0	0	82.7	2.0	2020	SPAL
39	2020-07-29	19:30	Serie A	Matchweek 37	Wed	Away	L	0	3	Hellas Verona	...	14.0	5.0	20.8	3.0	0	0	82.2	2.0	2020	SPAL
40	2020-08-02	18:00	Serie A	Matchweek 38	Sun	Home	L	1	3	Fiorentina	...	8.0	4.0	22.2	0.0	0	0	85.4	1.0	2020	SPAL

3222 rows × 29 columns

fig.11 : CSV iniziale

Abbiamo ottenuto un CSV di 3222 scontri di Serie A, ogni riga contiene **27*** colonne, queste colonne, oltre le metriche sopra citate, contengono:

- ***date***
- ***time***
- ***round*** = numero giornata
- ***day*** = giorno della settimana
- ***venue*** = se la squadra in "team" gioca in casa o in trasferta
- ***result*** = W (vittoria) D (pareggio) L (sconfitta)
- ***opponent*** = avversario
- ***attendance*** = numero di spettatori
- ***captain*** = nome capitano
- ***formation*** = modulo
- ***referee*** = nome arbitro
- ***team*** = squadra di cui facciamo riferimento

Dopo un ulteriore pulizia e aggiunta degli **identificatori temporali e di contesto**, otteniamo il seguente CSV :

	Date	Time	Round	Day	Venue	Result	GF	GA	Opponent	xG	...	PKatt	Cmp%	CrsPA	Season	Team	target	venue_code	opp_code	hour	day_code
0	2023-08-19	20:45	Matchweek 1	Sat	Home	W	2.0	0.0	Monza	2.7	...	0	88.0	3.0	2024	Internazionale	1	1	17	20	5
1	2023-08-28	20:45	Matchweek 2	Mon	Away	W	2.0	0.0	Cagliari	0.9	...	0	84.4	5.0	2024	Internazionale	1	0	4	20	0
2	2023-09-03	18:30	Matchweek 3	Sun	Home	W	4.0	0.0	Fiorentina	3.7	...	1	81.7	4.0	2024	Internazionale	1	1	8	18	6
3	2023-09-16	18:00	Matchweek 4	Sat	Home	W	5.0	1.0	Milan	2.6	...	1	80.9	3.0	2024	Internazionale	1	1	16	18	5
5	2023-09-24	12:30	Matchweek 5	Sun	Away	W	1.0	0.0	Empoli	1.6	...	0	83.3	6.0	2024	Internazionale	1	0	7	12	6
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
36	2020-07-19	19:30	Matchweek 34	Sun	Away	L	1.0	2.0	Brescia	1.7	...	0	78.4	0.0	2020	SPAL	0	0	3	19	6
37	2020-07-22	21:45	Matchweek 35	Wed	Home	L	1.0	6.0	Roma	0.8	...	0	84.8	0.0	2020	SPAL	0	1	20	21	2
38	2020-07-26	19:30	Matchweek 36	Sun	Home	D	1.0	1.0	Torino	0.4	...	0	82.7	2.0	2020	SPAL	0	1	26	19	6
39	2020-07-29	19:30	Matchweek 37	Wed	Away	L	0.0	3.0	Hellas Verona	0.7	...	0	82.2	2.0	2020	SPAL	0	0	11	19	2
40	2020-08-02	18:00	Matchweek 38	Sun	Home	L	1.0	3.0	Fiorentina	0.5	...	0	85.4	1.0	2020	SPAL	0	1	8	18	6

3222 rows × 32 columns

fig.12 : CSV arricchito da fornire in input al modello

***Sarebbero 29 ma "Comp" e "Note" verranno poi eliminate perché inutili.**

La colonna TARGET

Come notiamo nel secondo output, oltre agli **identificatori temporali e di contesto** è presente una colonna **target**, nel contesto dell'analisi dei dati calcistici, l'aggiunta di una colonna target è una mossa cruciale per la preparazione dei dati al fine di applicare con successo modelli di machine learning supervisionati. In particolare, questa colonna assume un ruolo di primaria importanza nel fornire al modello una chiara indicazione dell'obiettivo che deve cercare di predire.

La sua presenza consente al modello di apprendimento automatico di connettere le caratteristiche specifiche di ciascuna partita con il risultato finale, ovvero se la squadra ha **vinto (1)** o ha ottenuto un **risultato diverso (0)**.

Questo processo di supervisione è essenziale per l'allenamento del modello. Attraverso il dataset, il modello impara a riconoscere i modelli e le relazioni tra le diverse metriche. Questa conoscenza acquisita sarà quindi utilizzata per fare previsioni su nuovi dati, aiutando a prendere decisioni più informate.

Questa colonna target diventa il cuore del processo di addestramento e valutazione. Suddividendo il dataset in un set di allenamento e un set di test, il modello può essere addestrato su un sottoinsieme dei dati e successivamente valutato sulla sua capacità di generalizzare e fare previsioni accurate su dati non visti.

Capitolo 5

Ricerca e testing del modello ottimale

Nel processo di sviluppo di un modello di machine learning, una delle fasi cruciali è l'esplorazione e il test dei diversi modelli disponibili per il problema specifico che si sta affrontando. Questo capitolo è dedicato a esaminare il percorso seguito per identificare e valutare i modelli, fornendo una panoramica dettagliata del processo decisionale che ha portato alla selezione del modello finale.

Nel processo di sviluppo del modello per predire l'esito delle partite di calcio nel dataset fornito, è stata condotta un'analisi approfondita dei modelli disponibili per trovare la soluzione più adatta alle caratteristiche dei dati e agli obiettivi del progetto. Durante questa fase, sono stati esaminati diversi modelli, ciascuno con i suoi vantaggi e svantaggi. È importante notare che la selezione del **Random Forest** non è avvenuta a caso, ma è stata il risultato di una valutazione comparativa che ha escluso altri modelli in base alle loro prestazioni e alle esigenze specifiche del problema.

Random Forest : Vantaggi

Ho scelto Random Forest per questo compito per diversi motivi che si adattano bene alle caratteristiche del dataset iniziale e all'obiettivo di individuare l'esito di ogni scontro.

Gestione di dati eterogenei:

Il dataset contiene informazioni su molte variabili diverse, inclusi i nomi delle squadre, risultati, metriche di gioco e altri fattori. Random Forest è in grado di gestire dati eterogenei e combinare efficacemente informazioni da diverse fonti per fare previsioni.

Flessibilità nelle relazioni non lineari:

Le relazioni tra le variabili nel calcio possono essere complesse e non lineari. Random Forest è in grado di catturare queste relazioni non lineari, consentendo al modello di adattarsi ai pattern complessi presenti nei dati.

Robustezza contro overfitting:

Random Forest è nota per la sua capacità di gestire il problema dell'overfitting, che si verifica quando un modello si adatta troppo ai dati di addestramento e perde la capacità di generalizzare su nuovi dati. In un

contesto di predizione del risultato delle partite di calcio, è fondamentale evitare l'overfitting per garantire che il modello possa fare previsioni accurate su partite future non osservate.

Interpretabilità delle feature di importanza:

Random Forest fornisce una stima della rilevanza delle variabili nel processo decisionale del modello. Questo può essere particolarmente utile nel contesto del calcio, dove allenatori e analisti sono spesso interessati a capire quali fattori influenzano maggiormente l'esito di una partita.

Medie Mobili

Durante lo sviluppo del modello, sono stati condotti diversi test per valutare l'efficacia. Tra questi test, sono stati inclusi due approcci che hanno arricchito il dataset con l'aggiunta di medie mobili. Questi due approcci sono stati ritenuti particolarmente significativi in quanto hanno portato a risultati promettenti e rilevanti per la previsione.

Le medie mobili, in termini semplici, sono una tecnica statistica utilizzata per analizzare i dati temporali, come le prestazioni delle squadre di calcio nel corso della stagione. In questo contesto, una media mobile rappresenta la media dei risultati passati di una squadra su un determinato periodo di tempo. Ad esempio, una media mobile a 10 partite per una metrica "X" rappresenta il valore di quella metrica nelle ultime 10 partite disputate.

L'aggiunta di medie mobili al dataset consente al modello di considerare non solo i risultati delle singole partite, ma anche le tendenze e le dinamiche nel rendimento delle squadre nel tempo.

Questo può essere particolarmente utile nel calcio, dove le squadre possono attraversare periodi di forma eccezionale o di difficoltà che influenzano le loro prestazioni future.

I risultati ottenuti dai test che hanno arricchito il dataset con medie mobili hanno dimostrato di fornire miglioramenti significativi nelle prestazioni del modello rispetto agli altri approcci testati. Pertanto, questi test saranno i principali focus di questa analisi, poiché hanno fornito informazioni più promettenti e rilevanti per il problema specifico della previsione degli esiti delle partite di calcio.

1° Test

Dataset utilizzato :

Il dataset utilizzato per il primo test è una versione arricchita del dataset precedentemente mostrato. In aggiunta alle metriche spiegate nei capitoli sopra, questo dataset include le medie mobili su tutte le statistiche relative agli aspetti della partita, indicate con ***nome_metrica_rolling***. Le medie mobili sono calcolate su un periodo di **10** partite e rappresentano una media ponderata delle metriche registrate nelle partite precedenti.

Ecco il dataset :

Date	Time	Round	Day	Venue	Result	GF	GA	Opponent	xG	...	GA_rolling	Sh_rolling	SoT_rolling	Dist_rolling	FK_rolling	PK_rolling	PKatt_rolling	Cmp%_rolling	CrsPA_rolling	Poss_rolling
2019-11-03	12:30	Matchweek 11	Sun	Home	L	0.0	2.0	Cagliari	1.2	...	1.6	20.3	8.1	17.35	0.8	0.3	0.3	81.55	2.0	54.9
2019-11-10	15:00	Matchweek 12	Sun	Away	D	0.0	0.0	Sampdoria	0.6	...	1.6	20.1	7.1	17.30	0.9	0.3	0.3	82.10	2.1	54.7
2019-11-23	15:00	Matchweek 13	Sat	Home	L	1.0	3.0	Juventus	2.7	...	1.3	18.4	6.5	17.48	1.1	0.3	0.3	82.46	2.2	55.3
2019-11-30	15:00	Matchweek 14	Sat	Away	W	3.0	0.0	Brescia	2.7	...	1.5	18.8	6.7	17.56	1.1	0.2	0.3	82.19	2.5	55.2
2019-12-07	15:00	Matchweek 15	Sat	Home	W	3.0	2.0	Hellas Verona	2.4	...	1.3	19.3	7.4	17.15	0.8	0.2	0.3	82.29	2.8	55.6
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
2022-05-01	12:30	Matchweek 35	Sun	Away	L	1.0	2.0	Juventus	0.6	...	1.9	9.8	2.6	17.35	0.8	0.0	0.1	69.96	1.9	42.1
2022-05-05	18:00	Matchweek 20	Thu	Away	L	1.0	2.0	Salernitana	1.1	...	2.0	10.2	2.7	17.91	1.0	0.0	0.1	72.76	2.0	43.7
2022-05-08	15:00	Matchweek 36	Sun	Home	W	4.0	3.0	Bologna	2.8	...	2.1	9.7	2.7	17.17	1.0	0.0	0.1	74.53	2.2	44.8
2022-05-14	20:45	Matchweek 37	Sat	Away	D	1.0	1.0	Roma	0.3	...	2.1	10.4	3.1	17.92	1.0	0.1	0.3	75.77	2.6	44.4
2022-05-22	21:00	Matchweek 38	Sun	Home	D	0.0	0.0	Cagliari	1.0	...	1.8	9.3	2.9	18.97	0.9	0.1	0.2	74.91	2.3	43.0

fig.13 : CSV con medie mobili

Successivamente, ho definito una funzione chiamata ***make_predictions*** che si occupa di effettuare le predizioni utilizzando un classificatore Random Forest. Questa funzione suddivide il dataset in un set di addestramento e uno di test, utilizzando i dati fino al 1° maggio 2023 per addestrare il modello e i dati successivi per testarlo. Il modello viene addestrato sui parametri specificati, come il numero di estimatori (n_estimators), la suddivisione minima dei campioni (min_samples_split), e lo stato random (random_state). Successivamente, vengono effettuate le predizioni sul set di test e valutate rispetto ai risultati effettivi delle partite.

In sintesi, questo script esegue un processo completo che va dall'elaborazione dei dati, con l'aggiunta delle medie mobili, alla creazione e valutazione di un modello di predizione basato su Random Forest, utilizzando il dataset elaborato.

Valutazione dei risultati

Accuracy (Accuratezza): L'accuracy misura la proporzione di previsioni corrette rispetto al totale delle previsioni effettuate. È una metrica intuitiva e facile da interpretare, in quanto fornisce una panoramica generale delle prestazioni del modello. Tuttavia, l'accuracy potrebbe non essere sufficiente per fornire una valutazione completa, specialmente in presenza di classi sbilanciate. Ad esempio, se abbiamo un dataset in cui una classe è molto più comune dell'altra, un modello che predice sempre la classe maggioritaria potrebbe ancora ottenere un'accuracy elevata, ma sarebbe inefficace nel riconoscere la classe minoritaria. Pertanto, l'accuracy da sola potrebbe non essere sufficiente per fornire una valutazione accurata delle prestazioni del modello, specialmente in contesti in cui le classi sono sbilanciate.

ROC AUC (Area Under the Receiver Operating Characteristic curve): La curva ROC e l'Area Under the ROC curve (ROC AUC) sono metriche utilizzate per valutare la capacità discriminante di un modello di classificazione. La curva ROC rappresenta il tasso di vero positivo rispetto al tasso di falso positivo al variare della soglia di classificazione. L'Area Under the ROC curve fornisce una misura della capacità del modello di distinguere tra le classi positive e negative. Un valore ROC AUC vicino a 1 indica che il modello ha una buona capacità discriminante, mentre un valore vicino a 0.5 indica che il modello è poco meglio di un modello casuale. La ROC AUC è particolarmente utile quando si hanno classi sbilanciate o quando è necessario valutare la capacità di classificazione del modello a diverse soglie di classificazione.

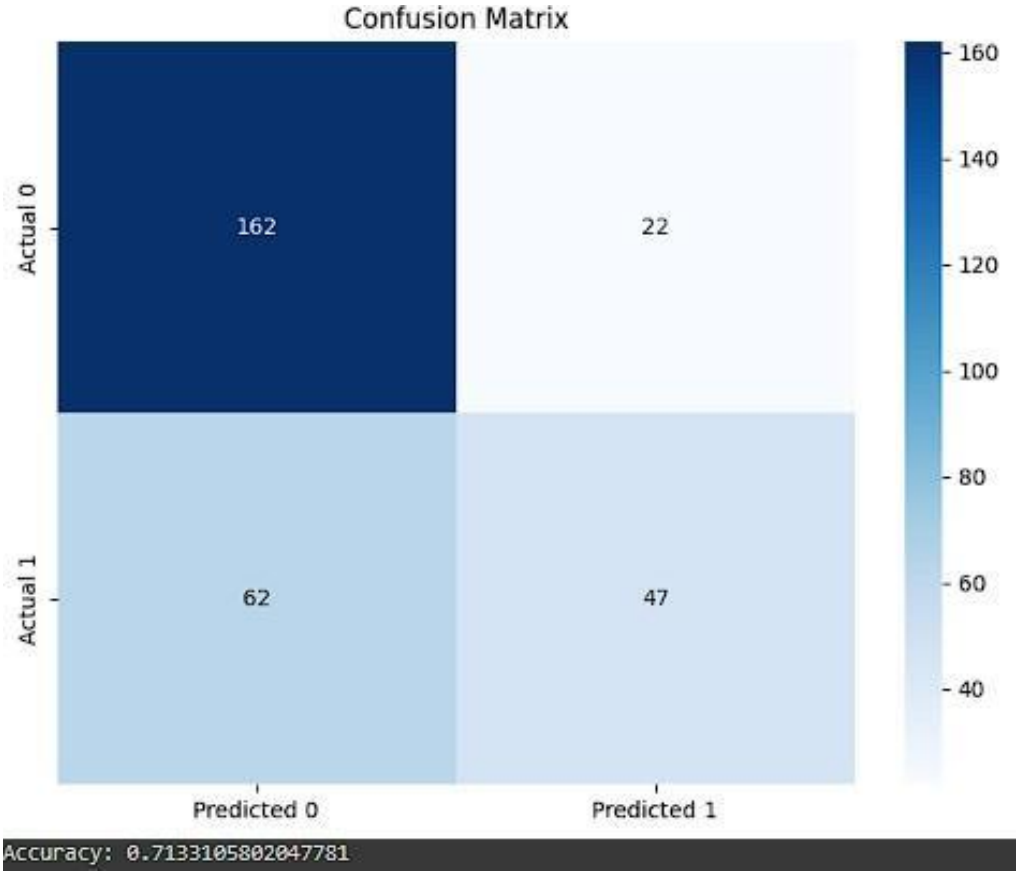

fig.14 : Matrice di confusione primo test

La matrice di confusione è uno strumento utile per valutare le prestazioni di un modello di classificazione. Essa mostra il numero di previsioni corrette e erronee fatte dal modello su un set di dati di test, confrontando le previsioni effettuate dal modello con i valori effettivi delle classi.[13] Nella matrice di confusione, i valori sono organizzati come segue:

Actual 1 (Predicted 1): Rappresenta i veri positivi (**True Positives**, TP), ossia i casi in cui il modello ha previsto correttamente che un'istanza appartiene alla classe positiva (1).

Actual 0 (Predicted 0): Rappresenta i veri negativi (**True Negatives**, TN), ossia i casi in cui il modello ha previsto correttamente che un'istanza appartiene alla classe negativa (0).

Predicted 1 (Colonna 1): Rappresenta le previsioni fatte dal modello per la classe positiva (1).

Predicted 0 (Colonna 0): Rappresenta le previsioni fatte dal modello per la classe negativa (0).

In riferimento all'accuracy riportata, il valore di **0.7133105802047781** indica la proporzione di previsioni corrette fatte dal modello rispetto al totale delle previsioni effettuate. Ciò significa che il 71.33% delle previsioni fatte dal modello è corretto, evidenziando un discreto livello di precisione nelle sue previsioni complessive.

Inoltre, notiamo un'alta precisione nell'indovinare i negativi, indicando un alto numero di veri negativi. Questo significa che il modello è molto efficace nel prevedere correttamente i casi in cui una partita non finisce con una vittoria.

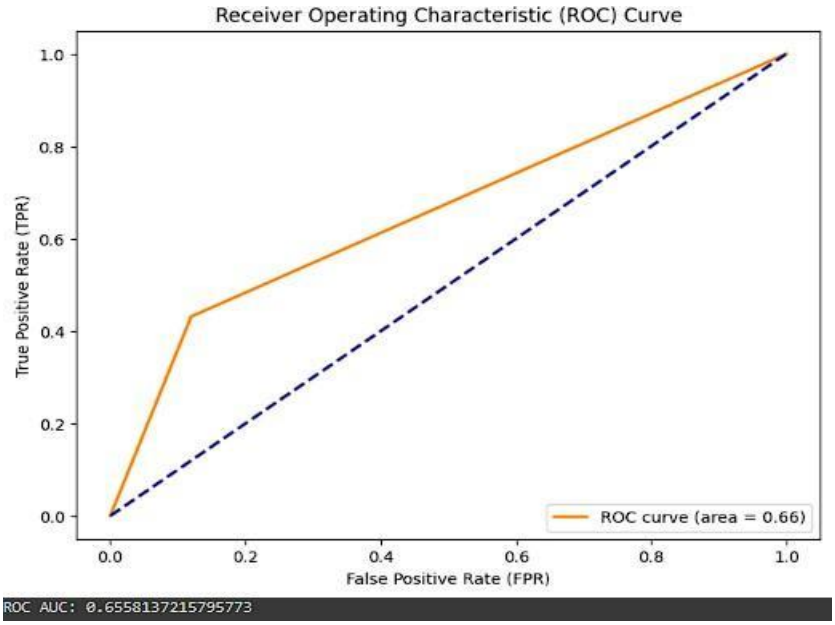

fig.15 : ROC primo test

questo test potrebbe essere considerato discreto, ci sono sempre opportunità di miglioramento e iterazione per affinare ulteriormente il modello e ottenere prestazioni migliori, vediamone una…

2° Test

Per il secondo test, ho deciso di arricchire ulteriormente il dataset includendo i valori di mercato di ogni giocatore di Serie A della stagione 2022/2023. Questi dati sono stati ottenuti da transfermarkt.it, una fonte autorevole nel mondo del calcio. Ho sommato i valori di mercato di ciascun giocatore all'interno di ciascuna squadra per ottenere il valore complessivo della rosa. Ho ritenuto che questo approccio fosse significativo, poiché il valore della rosa di una squadra può influenzare in modo sostanziale le sue prestazioni sul campo.

Integrare questa informazione nel modello mi ha permesso di catturare meglio la forza relativa delle squadre, fornendo così una panoramica più completa delle variabili che influenzano gli esiti delle partite di calcio. Con l'aggiunta di questo nuovo parametro, sono fiducioso che il modello avrà una maggiore capacità predittiva, consentendomi di ottenere risultati più accurati nelle previsioni degli esiti delle partite.

Step 1

- Prelievo valore di mercato di ogni singolo giocatore per squadra

	#	Giocatori	Nato il	Naz	Squadra attuale	Valore di mercato	Ruolo	Squadra
0	30	Wladimiro Falcone	12/apr/1995 (28)	NaN	NaN	5,00 mln €	Portiere	US-LECCE
1	21	Federico Brancolini	14/lug/2001 (21)	NaN	NaN	200 mila €	Portiere	US-LECCE
2	1	Marco Bleve	18/ott/1995 (27)	NaN	NaN	150 mila €	Portiere	US-LECCE
3	36	Jasper Samooja	21/lug/2003 (19)	NaN	NaN	75 mila €	Portiere	US-LECCE
4	6	Federico Baschirotto	20/set/1996 (26)	NaN	NaN	8,00 mln €	Difensore centrale	US-LECCE

fig.16 : CSV di dati di giocatori singoli

Step 2

- Somma e raggruppamento dei valori per squadra in un dataset che andrà poi a completare il dataset di base

Atalanta	338375000.0
Bologna	153625000.0
Cremonese	83775000.0
Empoli	125525000.0
Fiorentina	258800000.0
Hellas Verona	123885000.0
Inter	567550000.0
Juventus	483125000.0
Lazio	266425000.0
Lecce	105200000.0
Milan	545100000.0
Monza	135700000.0
Napoli	654075000.0
Roma	361375000.0
Salernitana	118350000.0
Sampdoria	82960000.0
Sassuolo	232200000.0
Spezia	115810000.0
Torino	197185000.0
Udinese	188860000.0

come possiamo notare dalle due immagini ogni valore è stato normalizzato per poter essere sommato e poi inserito nel modello.

fig.17 : CSV di prezzi delle squadre di Serie A

Step 3

- Unione dei dati ricercati

	Date	Time	Round	Day	Venue	Result	GF	GA	Opponent	xG	...	Cmp%	CrsPA	Team	target	venue_code	opp_code	hour	day_code	Valore di mercato_Team	Valore di mercato_Opponent
0	2022-08-15	18:30	Matchweek 1	Mon	Away	W	5	2	Hellas Verona	2.4	...	83.9	3.0	Napoli	1	0	5	18	0	654075000.0	123885000.0
1	2022-08-21	18:30	Matchweek 2	Sun	Home	W	4	0	Monza	2.0	...	88.0	2.0	Napoli	1	1	11	18	6	654075000.0	135700000.0
2	2022-08-28	20:45	Matchweek 3	Sun	Away	D	0	0	Fiorentina	1.7	...	84.3	0.0	Napoli	0	0	4	20	6	654075000.0	258800000.0
3	2022-08-31	20:45	Matchweek 4	Wed	Home	D	1	1	Lecce	1.7	...	84.9	3.0	Napoli	0	1	9	20	2	654075000.0	105200000.0
4	2022-09-03	20:45	Matchweek 5	Sat	Away	W	2	1	Lazio	2.1	...	87.1	4.0	Napoli	1	0	8	20	5	654075000.0	266425000.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
757	2023-05-08	18:30	Matchweek 34	Mon	Away	L	0	2	Udinese	0.9	...	84.8	0.0	Sampdoria	0	0	19	18	0	82960000.0	188860000.0
758	2023-05-15	20:45	Matchweek 35	Mon	Home	D	1	1	Empoli	1.3	...	71.0	4.0	Sampdoria	0	1	3	20	0	82960000.0	125525000.0
759	2023-05-20	20:45	Matchweek 36	Sat	Away	L	1	5	Milan	0.6	...	81.7	0.0	Sampdoria	0	0	10	20	5	82960000.0	545100000.0
760	2023-05-26	20:45	Matchweek 37	Fri	Home	D	2	2	Sassuolo	2.2	...	75.1	1.0	Sampdoria	0	1	16	20	4	82960000.0	232200000.0
761	2023-06-04	18:30	Matchweek 38	Sun	Away	L	0	2	Napoli	0.6	...	78.8	3.0	Sampdoria	0	0	12	18	6	82960000.0	654075000.0

fig.18 : CSV ottenuto da fornire in input al modello

Questa volta, per addestrare il modello, abbiamo optato per una divisione dei dati di train in modo diverso. Abbiamo preso in considerazione le partite disputate prima del 1° maggio 2023, totalizzando un totale di 602 partite, e le abbiamo utilizzate come set di **train**. Per il set di **test**, abbiamo selezionato le restanti 154 partite della stagione. Questa suddivisione ci ha permesso di avere una visione più bilanciata delle prestazioni del modello su dati di train e test provenienti da periodi temporali diversi della stagione calcistica. Tale approccio ci permette di valutare meglio la capacità predittiva del modello in contesti temporali diversi e di ottenere una valutazione più accurata delle sue prestazioni.

Valutazione dei risultati

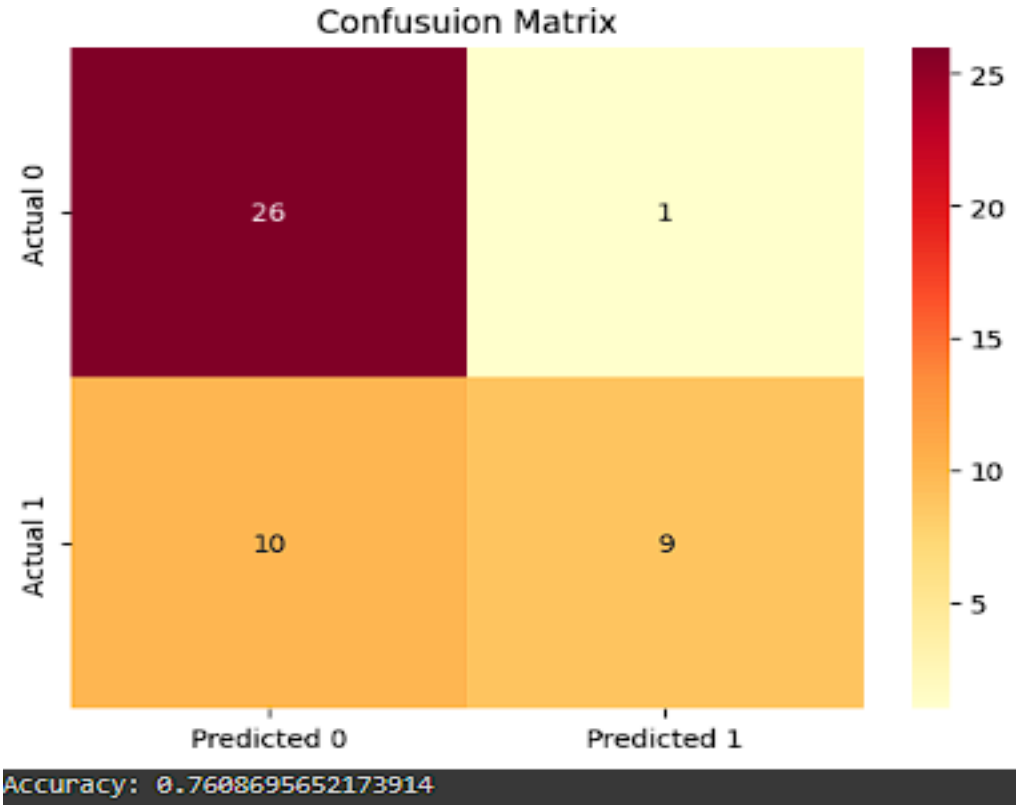

fig.19 : Matrice di confusione secondo test

È notevole notare che nonostante le dimensioni ridotte del set di dati di test, la precisione del nostro modello è eccezionalmente alta, raggiungendo il 90%. Questo significa che il 90% delle previsioni positive fatte dal modello è corretto.

Inoltre, mentre la dimensione del set di dati di test è stata ridotta, l'accuratezza complessiva del modello è aumentata significativamente, raggiungendo il 76%. Questo suggerisce che il modello è riuscito a generalizzare bene anche su un set di dati più piccolo, migliorando la sua

capacità di fare previsioni accurate su nuovi dati.

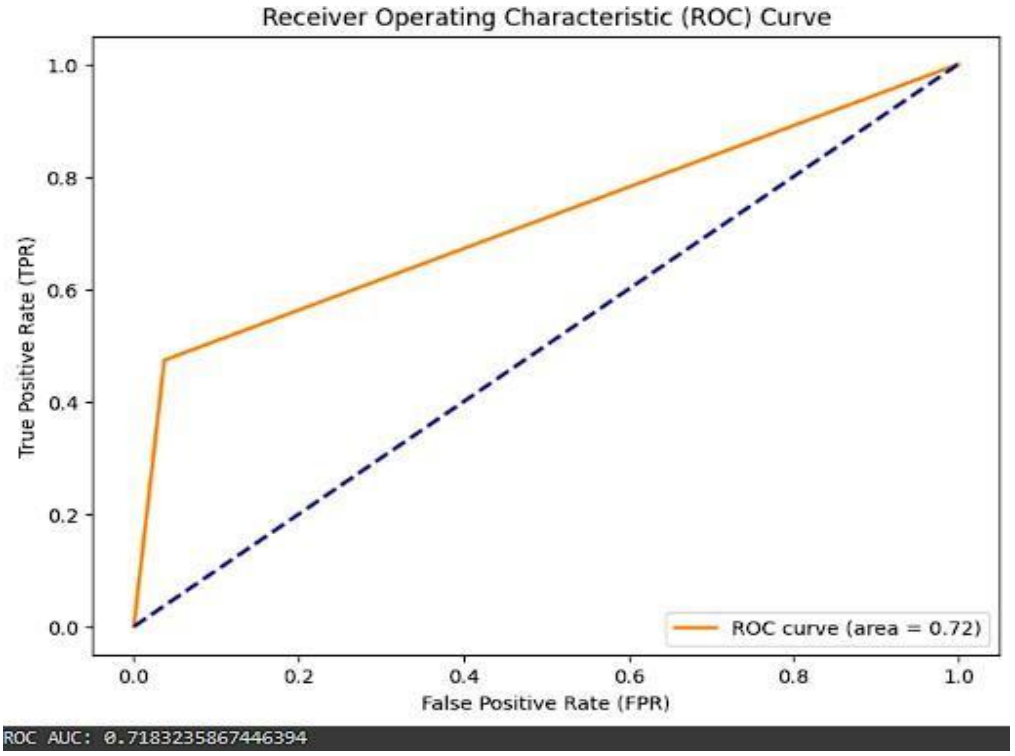

fig.20 : ROC secondo test

Il modello ha una discreta capacità discriminativa, ma c'è spazio per migliorare ulteriormente le prestazioni.

È importante sottolineare che questi risultati mettono in luce l'importanza della metrica relativa al valore della squadra nel contesto delle previsioni degli esiti delle partite di calcio. L'aggiunta di questa variabile al nostro modello ha contribuito significativamente alla sua capacità predittiva, come evidenziato dalla precisione del 90% e dall'accuratezza del 76%. Questo sottolinea quanto sia cruciale considerare fattori come il valore della squadra, che possono influenzare in modo significativo le prestazioni e gli esiti delle partite.

Funzionamento concreto del modello

actual	predicted	Date	Team	Opponent	Result
0	0	2023-05-27	Atalanta	Inter	L
1	1	2023-06-04	Atalanta	Monza	W
0	0	2023-05-28	Bologna	Napoli	D
1	0	2023-06-04	Bologna	Lecce	W
0	0	2023-05-28	Cremonese	Lazio	L
1	0	2023-06-03	Cremonese	Salernitana	W
1	0	2023-05-22	Empoli	Juventus	W
0	0	2023-05-28	Empoli	Hellas Verona	D
0	0	2023-06-03	Empoli	Lazio	L
1	0	2023-05-27	Fiorentina	Roma	W
1	0	2023-06-02	Fiorentina	Sassuolo	W
0	0	2023-05-28	Hellas Verona	Empoli	D
0	0	2023-06-04	Hellas Verona	Milan	L
1	0	2023-06-11	Hellas Verona	Spezia	W
1	1	2023-05-27	Inter	Atalanta	W
1	1	2023-06-03	Inter	Torino	W
0	0	2023-05-22	Juventus	Empoli	L
0	0	2023-05-28	Juventus	Milan	L
1	0	2023-06-04	Juventus	Udinese	W
1	1	2023-05-28	Lazio	Cremonese	W
1	1	2023-06-03	Lazio	Empoli	W
1	0	2023-05-28	Lecce	Monza	W
0	0	2023-06-04	Lecce	Bologna	L
1	1	2023-05-28	Milan	Juventus	W
1	1	2023-06-04	Milan	Hellas Verona	W
0	0	2023-05-28	Monza	Lecce	L
0	0	2023-06-04	Monza	Atalanta	L
0	1	2023-05-28	Napoli	Bologna	D

La colonna "**actual**" contiene le vere etichette di classe delle istanze nel set di dati di test. Queste etichette indicano l'esito reale delle partite di calcio, ad esempio 1 se la squadra ha vinto e 0 se ha perso o pareggiato.

La colonna "**predicted**" contiene le previsioni fatte dal modello per le stesse istanze. Queste previsioni indicano l'esito predetto dal modello per le partite di calcio, basate sulle caratteristiche delle partite nel set di dati di test

fig.21 : CSV che mostra il funzionamento concreto del modello

Capitolo 6

Conclusione

La realizzazione di questo progetto ha portato alla luce diverse considerazioni cruciali riguardanti la modellazione predittiva nel contesto calcistico ma in particolare dello sport
Tuttavia, è importante riconoscere che la raccolta e l'elaborazione di grandi quantità di dati possono presentare sfide significative, specialmente quando si lavora con limitazioni hardware. In particolare, le capacità computazionali possono limitare la quantità di dati che è possibile elaborare e analizzare, influenzando direttamente la dimensione del set di dati utilizzabile per addestrare e valutare i modelli predittivi. Queste limitazioni possono portare a compromessi nella complessità dei modelli e nella loro capacità di catturare la complessità dei dati sottostanti.

Partendo dal concetto di Moneyball applicato al baseball, è importante riconoscere che il baseball è uno sport caratterizzato da un numero limitato di variabili rispetto al calcio. Nel baseball, le azioni sono definite da singole giocate, come lanci, battute e basi rubate, il che rende più “semplice” l'analisi delle prestazioni dei giocatori e delle squadre. Questo ambiente più strutturato e misurabile ha reso il baseball una pietra miliare nell'applicazione di metodologie analitiche e predittive nello sport.

Al contrario, il calcio è notoriamente complesso e dinamico, con un'ampia gamma di variabili che influenzano il risultato di una partita.

Nonostante questa sfida, l'applicazione di metodologie analitiche nel calcio,

ispirate al concetto di Moneyball, ha dimostrato di avere un potenziale significativo nel migliorare le prestazioni delle squadre e ottimizzare le risorse. Sebbene il calcio possa presentare più variabili da considerare, l'uso di modelli predittivi avanzati e l'analisi dei dati possono ancora fornire insight preziosi e informazioni utili per le società di calcio nel prendere decisioni tattiche, strategiche e finanziarie. In questo contesto, l'applicazione di approcci simili a quelli utilizzati nel baseball può offrire un'opportunità unica per innovare e ottenere vantaggi competitivi nel mondo del calcio professionistico.

Tuttavia, esistono una vasta gamma di modelli e tecniche che possono essere utilizzati per affrontare il problema della previsione degli esiti delle partite di calcio. Dalle semplici regressioni lineari ai più complessi algoritmi di machine learning, le opzioni sono molteplici e offrono opportunità per miglioramenti significativi nelle prestazioni predittive. L'uso di algoritmi più avanzati come Random Forest ha dimostrato di poter ottenere risultati **promettenti**, consentendo di catturare relazioni complesse nei dati e migliorare le previsioni degli esiti delle partite.

Dopo aver condotto un'analisi dettagliata dei risultati ottenuti dal mio modello attuale, riconosco che ci sono diverse aree in cui potrei apportare miglioramenti significativi. Nonostante abbia ottenuto buoni risultati, sono consapevole che c'è infinito spazio per l'ottimizzazione e il raffinamento delle prestazioni del modello.

Una delle possibili vie di miglioramento potrebbe riguardare l'ingegneria delle metriche. Considerando l'ampia gamma di metriche che possono influenzare gli esiti delle partite di calcio, potrei esplorare ulteriori caratteristiche o variabili rilevanti per la previsione.

Inoltre, potrei valutare l'utilizzo di modelli ensemble, che combinano le previsioni di diversi modelli per ottenere una previsione più robusta e accurata. Sperimentare con l'ottimizzazione degli iperparametri potrebbe anche rivelarsi utile, cercando la combinazione ottimale di parametri per massimizzare le prestazioni del modello.

In conclusione, sono soddisfatto dei risultati finora ottenuti, sono motivato a continuare a esplorare nuove possibilità e approfondire le analisi al fine di ottenere risultati ancora più significativi e utili, sebbene ci siano sfide da affrontare, l'analisi e la previsione degli esiti delle partite di calcio offrono un campo ricco di opportunità per l'applicazione di tecniche di data science e machine learning. Attraverso un approccio metodico e l'adozione di modelli avanzati e tecniche di ottimizzazione, è possibile ottenere previsioni più accurate e significative, aprendo la strada a nuove prospettive nel campo dell'analisi sportiva e della modellazione predittiva.

Bibliografia

[1] *Reider B. Moneyball. The American Journal of Sports Medicine. 2014;42(3):533-535. doi:10.1177/0363546514524161*

[2] *Pisano, Joseph. "Moneyball." (2011): 53-55.*

[3] *RATHOD, MILAN RAJESHBHAI. "Characterization and calibration of sensors on a wearable device for the analysis of outdoor sports activity."*

[4] *Mahesh B. Machine learning algorithms-a review. International Journal of Science and Research (IJSR).[Internet]. 2020 Jan;9(1):381-6.*

[5] *Horvat, Tomislav, and Josip Job. "The use of machine learning in sport outcome prediction: A review." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 10, no. 5 (2020): e1380.*

[6] *Bruce G. Marcot, Trent D. Penman, Advances in Bayesian network modelling: Integration of modelling technologies, Environmental Modelling & Software, Volume 111, 2019, Pages 386-393, ISSN 1364-8152, https://doi.org/10.1016/j.envsoft.2018.09.016. (https://www.sciencedirect.com/science/article/pii/S1364815218302937)*

[7] *Rigatti, Steven J. "Random forest." Journal of Insurance Medicine 47, no. 1 (2017): 31-39.*

[8] *Floreano, Dario, and Claudio Mattiussi. Manuale sulle reti neurali. Il mulino, 2002. Accessed 1 February 2024*

[9] *Gabriel Fialho, Aline Manhães, João Paulo Teixeira, Predicting Sports Results with Artificial Intelligence – A Proposal Framework for Soccer Games, Procedia Computer Science, Volume 164, 2019, Pages 131-136, ISSN 1877-0509, https://doi.org/10.1016/j.procs.2019.12.164. (https://www.sciencedirect.com/science/article/pii/S1877050919322033)*

[10] *Hughes, Michael, Tim Caudrelier, Nic James, Athalie Redwood-Brown, Ian Donnelly, Anthony Kirkbride, and Christophe Duschesne. "Moneyball and soccer-an analysis of the key performance indicators of elite male soccer players by position." Journal of Human Sport and Exercise 7, no. 2 (2012): 402-412.*

[11]Beneventano, Philip, Paul D. Berger and Bruce D. Weinberg.
"Predicting Run Production and Run Prevention in Baseball: The Impact of

Sabermetrics.” (2012).

[12]Richardson, Leonard. "Beautiful soup documentation." (2007).

[13]Waskom, Michael L. "Seaborn: statistical data visualization." Journal

of

Open Source Software 6, no. 60 (2021): 3021.