Documentazione Progetto FootballPlayers

Gruppo di lavoro

• Alessandro Aldo Boffolo, mat. 735963, a.boffolo@studenti.uniba.it

URL repo associato, contenente il materiale completo

https://github.com/Ale210501/FootballPlayers_ICon

AA 2023-24

Introduzione

Il progetto *FootballPlayers* si propone di analizzare e classificare i giocatori di calcio utilizzando tecniche avanzate di machine learning. L'obiettivo principale è duplice: prevedere il ruolo di un giocatore basato su vari indicatori di performance e raccomandare giocatori in base alle preferenze dell'utente. Il progetto si avvale di modelli di classificazione e di raccomandazione per offrire previsioni accurate e suggerimenti personalizzati.

Il dataset utilizzato è stato acquisito da Kaggle, e dopo un'accurata fase di preprocessing, è stato impiegato per addestrare e testare i modelli di machine learning. Il progetto esplora diverse tecniche di apprendimento automatico, inclusi K-Nearest Neighbors, Gaussian Naive Bayes e Random Forest, e implementa un sistema di raccomandazione basato sui dati forniti dagli utenti, grazie all'utilizzo della similarità del coseno.

Sommario

Il progetto *FootballPlayers* utilizza un Knowledge-Based System (KBS) per gestire e analizzare i dati relativi ai giocatori di calcio, integrando vari moduli che dimostrano competenze avanzate in diverse aree del machine learning e dell'analisi dei dati. Il sistema si articola in quattro moduli principali, ognuno dei quali affronta un aspetto specifico del progetto:

Elenco argomenti di interesse

• **Preprocessing**: Il modulo di preprocessing è cruciale per la preparazione dei dati, e si occupa di garantire che il dataset sia pulito, coerente e pronto per le fasi successive dell'analisi. Le attività di preprocessing includono la rimozione di valori mancanti, la gestione di outlier, la normalizzazione e la trasformazione delle variabili. Utilizzando librerie come Pandas e Scikit-Learn, sono stati applicati vari metodi per migliorare la qualità dei dati e assicurare che i modelli di machine learning operino su dati accurati e ben strutturati. Questa fase prepara i dati per le analisi predittive e descrittive, assicurando una base solida per le altre fasi del progetto.

• **Classificazione**: Il modulo di classificazione si focalizza sulla previsione del ruolo dei giocatori utilizzando tecniche di machine learning supervisionato. Sono stati implementati e testati diversi algoritmi di classificazione, tra cui K-Nearest Neighbors (KNN), Gaussian Naive Bayes e Random Forest. Ogni modello è stato valutato in base a metriche di performance come accuratezza, precisione, richiamo e F1-score. Questo modulo dimostra competenze nella costruzione e nella valutazione di modelli predittivi, confrontando le performance dei diversi algoritmi per identificare quello più adatto al problema specifico. L'analisi dei risultati

consente di ottimizzare le previsioni e di comprendere meglio i fattori che influenzano i ruoli dei giocatori.

• **Clustering**: Il modulo di clustering esplora la segmentazione dei giocatori in gruppi omogenei basati su caratteristiche simili. Tecniche di clustering come K-means e DBSCAN sono state utilizzate per identificare gruppi di giocatori che condividono attributi comuni, come performance o stili di gioco. Questo modulo permette di scoprire pattern e strutture nascoste nei dati, facilitando una comprensione più profonda delle relazioni tra i giocatori. La competenza dimostrata include l'analisi esplorativa dei dati e l'interpretazione dei risultati del clustering per fornire insights significativi sulle caratteristiche dei giocatori.

• **Raccomandazione**: Il sistema di raccomandazione è progettato per suggerire giocatori in base ai dati forniti dall'utente, utilizzando tecniche di similitudine e matching. Questo modulo offre raccomandazioni personalizzate che aiutano gli utenti a trovare giocatori che soddisfano specifici criteri o preferenze. Il sistema è stato progettato e testato per garantire che le raccomandazioni siano pertinenti e utili, dimostrando competenze nella progettazione di algoritmi di raccomandazione e nella valutazione della loro efficacia. L'implementazione include l'uso di tecniche avanzate per migliorare la qualità delle raccomandazioni e fornire suggerimenti rilevanti basati sulle preferenze degli utenti.

Preprocessing dei dati

Sommario

Il modulo di preprocessing dei dati nel progetto *FootballPlayers* è progettato per preparare i dati grezzi per l'analisi e la modellazione successiva. Questo modulo include la pulizia dei dati, la gestione dei valori mancanti, la trasformazione delle variabili e la selezione delle colonne di interesse. La rappresentazione della conoscenza nel preprocessing è basata su tecniche consolidate di data cleaning e data transformation, essenziali per garantire dati coerenti e di alta qualità. La qualità dei dati è fondamentale per le fasi di classificazione, clustering e raccomandazione, poiché modelli ben progettati richiedono input pulito e ben strutturato.

Strumenti utilizzati

Per il preprocessing dei dati sono stati utilizzati diversi strumenti e librerie standard nel campo della scienza dei dati:

- **Pandas**: Per la manipolazione e la pulizia dei dati.
- **Scikit-Learn**: Per la normalizzazione dei dati e la gestione dei valori mancanti.
- **NumPy**: Per operazioni matematiche e trasformazioni numeriche.

Decisioni di Progetto

Le decisioni di progetto relative al preprocessing dei dati includono:

- **Gestione dei Valori Mancanti**: I valori mancanti sono stati gestiti mediante imputazione, utilizzando la media per le variabili numeriche e la moda per le variabili categoriche.
- **Normalizzazione**: I dati numerici sono stati normalizzati utilizzando la StandardScaler di Scikit-Learn per garantire che tutte le variabili abbiano una scala comune.
- **Selezione delle Colonne**: Sono state selezionate solo le colonne di interesse per l'analisi, rimuovendo le colonne non rilevanti.

Queste decisioni sono state prese per ottimizzare la qualità dei dati e preparare un dataset adatto per l'analisi successiva.

Il dataset ottenuto dopo la fase di preprocessing è, quindi, il seguente:

	Born	90s	Gls	Ast	CrdY	CrdR	Pos_format	Player_format
1	2000	14	0	1	1	0	1	0
2	1999	5	0	0	2	0	1	1
3	1999	14	2	1	5	0	1	1
4	1996	4	0	0	1	0	1	2
5	1997	27	3	4	6	0	1	3
6	2000	25	9	7	6	0	1	4
7	1995	17	0	0	1	0	1	5
8	1998	24	1	0	6	0	1	6
9	1992	18	0	0	2	0	1	7
10	1992	21	1	0	4	0	1	8
11	1995	24	3	2	4	0	1	9
12	2003	0	0	0	0	0	1	10
13	1994	15	0	2	3	0	1	11
14	1998	12	1	1	1	0	1	12

Valutazione

La valutazione del preprocessing è stata effettuata attraverso i seguenti metodi:

- **Analisi della Qualità dei Dati**: Sono stati utilizzati strumenti di Pandas per verificare la completezza e la coerenza dei dati dopo il preprocessing.
- **Metriche di Preprocessing**: Le metriche principali incluse sono la percentuale di valori mancanti rimossi, il range e la media delle variabili normalizzate.

Classificazione dei ruoli dei giocatori

Sommario

Il modulo di classificazione nel progetto *FootballPlayers* si concentra sulla predizione del ruolo dei giocatori, utilizzando dati storici e statistici per apprendere modelli in grado di assegnare un giocatore a una determinata posizione (ad esempio attaccante, centrocampista, difensore). Il modello di apprendimento supervisionato scelto per rappresentare la conoscenza del dominio si basa su tecniche di classificazione come K-Nearest Neighbors (KNN), Random Forest e Naive Bayes. I dati utilizzati includono variabili statistiche sui giocatori come gol, assist, e cartellini, che vengono processati per costruire una Knowledge Base (KB) utile a predire correttamente il ruolo del giocatore.

Strumenti utilizzati

Sono stati utilizzati i seguenti strumenti e librerie per la classificazione:

- **Scikit-Learn**: Utilizzato per implementare i modelli di KNN, Naive Bayes e Random Forest.
- **Pandas**: Per gestire il dataset e preparare i dati per il modello.
- **NumPy**: Per supporto a operazioni numeriche complesse.

Decisioni di Progetto

Le seguenti decisioni sono state prese durante lo sviluppo del modulo di classificazione:

- **Modelli Utilizzati**: Sono stati implementati tre diversi modelli di classificazione: K-Nearest Neighbors, Gaussian Naive Bayes e Random Forest. Questi modelli sono stati scelti per la loro capacità di gestire dati eterogenei e la loro efficienza in diversi contesti di classificazione.
- **Configurazione dei Parametri**:
    - **KNN**: Per KNN, è stato utilizzato un valore di k=5, ottimizzato tramite validazione incrociata.
    - **Random Forest**: Sono stati utilizzati 100 alberi decisionali (n_estimators=100) e una profondità massima (max_depth=10) per evitare overfitting.
    - **Gaussian Naive Bayes**: Per il modello Naive Bayes, è stato utilizzato il modello standard di Scikit-Learn senza ulteriori ottimizzazioni.

- **Preprocessing Aggiuntivo**: Prima dell'addestramento, i dati sono stati normalizzati usando StandardScaler, e le variabili categoriali sono state codificate tramite One-Hot Encoding.

Valutazione

La valutazione del modello è stata eseguita utilizzando diverse metriche di classificazione standard:

- **Accuracy**: Percentuale di previsioni corrette sul dataset.
- **Precision, Recall, F1-Score**: Metriche utilizzate per valutare le prestazioni del modello per ciascuna classe.
- **Support**: Indica il numero di occorrenze effettive di ciascuna classe nel dataset.

In particolare, per quanto riguarda il KNN:

```
Accuratezza (KNN): 0.8842105263157894
              precision    recall  f1-score   support

           0       0.85      0.80      0.83        56
           1       0.97      0.96      0.96       323
           2       0.79      0.80      0.80       191
           3       0.86      0.87      0.86       285

    accuracy                           0.88       855
   macro avg       0.87      0.86      0.86       855
weighted avg       0.88      0.88      0.88       855
```

Invece, per quanto riguarda il Gaussian Naive Bayes:

```
Accuratezza (GaussianNB): 0.847953216374269
              precision    recall  f1-score   support

           0       0.47      0.98      0.63        56
           1       0.97      0.94      0.96       323
           2       0.84      0.68      0.75       191
           3       0.87      0.83      0.85       285

    accuracy                           0.85       855
   macro avg       0.79      0.86      0.80       855
weighted avg       0.88      0.85      0.85       855
```

Infine, per quanto riguarda il Random Forest:

```
Accuratezza (RandomForest): 0.8888888888888888
              precision    recall  f1-score   support

           0       0.88      0.77      0.82        56
           1       0.99      0.96      0.97       323
           2       0.76      0.86      0.81       191
           3       0.88      0.86      0.87       285

    accuracy                           0.89       855
   macro avg       0.88      0.86      0.87       855
weighted avg       0.89      0.89      0.89       855
```

Ciò ha portato a scegliere il modello Random Forest per prevedere il ruolo di un giocatore. Ad esempio:

```
Ciao, cosa vuoi fare?
Vuoi che ti suggerisca un giocatore? - Premi 1
Vuoi sapere il ruolo di un giocatore? - Premi 2
Vuoi uscire? - Premi 3
2
Ti chiederò un po' di cose. Iniziamo...
Qual è il nome del giocatore che vuoi classificare?
 sergi roberto
In che anno è nato il giocatore che vuoi classificare?
1992
Quante partite ha giocato il giocatore che vuoi classificare? (se non sai come calcolare il numero di partite giocate digita 0)
10
Quanti goal ha segnato il giocatore che vuoi classificare?
3
Quanti assist ha fatto il giocatore che vuoi classificare?
2
Quanti cartellini gialli ha preso il giocatore che vuoi classificare?
4
Quanti cartellini rossi ha preso il giocatore che vuoi classificare?
0
Il ruolo del giocatore sergi roberto è:  df
```

Clustering dei giocatori

Sommario

Il modulo di clustering esplora la segmentazione dei giocatori in gruppi omogenei basati su caratteristiche simili. Nel progetto, è stato utilizzato principalmente l'algoritmo di clustering K-means per identificare gruppi di giocatori con attributi comuni come prestazioni, posizione e stile di gioco. Questo approccio consente di scoprire pattern nascosti e strutture nei dati, facilitando una comprensione più profonda delle relazioni tra i giocatori. L'analisi esplorativa dei dati e l'interpretazione dei risultati del clustering aiutano a ottenere insight significativi sulle caratteristiche dei giocatori.

Strumenti utilizzati

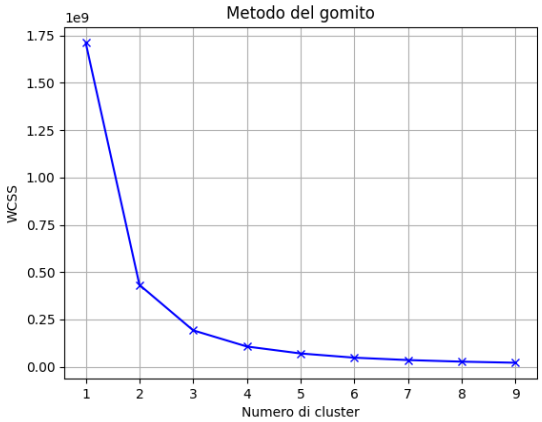
Per il clustering è stato usato:

- **K-means**: Un algoritmo di clustering centrato, che raggruppa i giocatori in base alla distanza euclidea rispetto ai centri di cluster. K-means è stato scelto per la sua semplicità e l'efficacia nel fornire una prima segmentazione dei dati.

Decisione di progetto

La configurazione dei componenti del progetto è stata stabilita in base alla natura dei dati e agli obiettivi di segmentazione. Ecco le principali decisioni prese:

- **Numero di cluster in K-means**: Dopo un'analisi esplorativa iniziale, il valore di K è stato scelto in base al metodo "elbow", che suggerisce il numero ottimale di cluster per minimizzare la somma delle distanze all'interno dei cluster.



Valutazione

La valutazione del clustering è stata basata sull'analisi qualitativa dei gruppi di giocatori ottenuti con K-means. I risultati hanno rivelato cluster distinti che rappresentano gruppi omogenei di giocatori con attributi simili.

Raccomandazione dei giocatori

Sommario

Il modulo di raccomandazione dei giocatori del progetto FootballPlayers suggerisce calciatori in base ai dati forniti dall'utente, sfruttando le statistiche dei giocatori e le informazioni storiche. Il sistema utilizza la **distanza del coseno** per calcolare la similarità tra i giocatori. Questa metrica consente di misurare quanto due profili di giocatori siano simili tra loro, basandosi su attributi come gol segnati, assist, cartellini e ruoli. La Knowledge Base (KB) è costituita dal dataset pre-elaborato dei giocatori, che include le statistiche rilevanti per la raccomandazione. Questo approccio consente di fornire suggerimenti personalizzati e pertinenti, migliorando la precisione delle raccomandazioni.

Strumenti utilizzati

Il modulo di raccomandazione è stato costruito utilizzando i seguenti strumenti:

- **Distanza del Coseno:** Utilizzata per calcolare la similarità tra i giocatori basata sui loro attributi. La distanza del coseno è implementata tramite la funzione cosine_similarity di sklearn.metrics.pairwise.
- **Pandas e NumPy:** Utilizzati per la gestione dei dati e il calcolo delle distanze tra i giocatori.
- **Scikit-Learn:** Per la normalizzazione dei dati e il clustering dei giocatori tramite l'algoritmo K-Means.

Decisioni di Progetto

Le principali decisioni progettuali per il sistema di raccomandazione includono:

- **Algoritmo di Similarità**: La distanza del coseno è stata scelta per confrontare i giocatori in base alle loro statistiche. Questo approccio è vantaggioso quando si desidera misurare la similarità tra vettori di caratteristiche.
- **Normalizzazione dei Dati**: I dati dell'utente e dei giocatori sono stati normalizzati utilizzando la funzione preprocessing.normalize di sklearn. Questo passo è essenziale per garantire che le differenze di scala tra le variabili non influenzino il calcolo della similarità.
- **Clustering**: L'algoritmo K-Means è stato utilizzato per segmentare i giocatori in cluster, facilitando la selezione dei giocatori simili. Il numero di cluster è stato impostato a 3.

- **Input Utente**: Il sistema consente all'utente di inserire preferenze specifiche (ad es., ruolo preferito o statistiche minime). Questi dati vengono utilizzati per filtrare e personalizzare le raccomandazioni.

Valutazione

Il sistema di raccomandazione è stato valutato attraverso vari scenari di input per testare la precisione e l'utilità delle raccomandazioni fornite. Le metriche adottate includono:

- **Precisione delle Raccomandazioni:** Valutata osservando la pertinenza dei suggerimenti rispetto alle preferenze inserite dall'utente.
- **Feedback dell'Utente:** Gli utenti hanno fornito feedback sulla coerenza dei suggerimenti rispetto alle loro aspettative. Le raccomandazioni risultano pertinenti e ben allineate con le preferenze degli utenti.

Il sistema dimostra una solida integrazione della distanza del coseno per la raccomandazione di giocatori, permettendo una selezione più accurata e mirata basata sui dati forniti dall'utente.

Esempio di raccomandazione di giocatori:

```
Ciao, cosa vuoi fare?
Vuoi che ti suggerisca un giocatore? - Premi 1
Vuoi sapere il ruolo di un giocatore? - Premi 2
Vuoi uscire? - Premi 3
1
Ti farò delle domande per poterti suggerire un giocatore.
Suggeriscimi il nome di un giocatore che ti piace
alphonso davies
Sai dirmi in che ruolo gioca questo giocatore? (Si o No)(se non sai i ruoli esistenti digita 0)
df
Sai dirmi in che anno è nato questo giocatore? (Si o No)
no
Sai dirmi quante partite ha giocato questo giocatore? (Si o No)(se non sai come calcolare il numero di partite giocate digita 0)
si
Quante partite ha giocato?
23
Sai dirmi quanti goal ha segnato questo giocatore? (Si o No)
si
Quanti goal ha segnato?
2
Sai dirmi quanti assist ha fatto questo giocatore? (Si o No)
si
Quanti assist ha fatto?
5
Sai dirmi quanti cartellini gialli ha preso questo giocatore? (Si o No)
si
Quanti cartellini gialli ha preso?
0
```

```
Sai dirmi quanti cartellini rossi ha preso questo giocatore? (Si o No)
si
Quanti cartellini rossi ha preso?
0
I giocatori suggeriti in base alle tue preferenze sono:

1) jeremie frimpong
2) christopher operi
3) ben osborn
4) matteo gabbia
5) przemysław frankowski
6) max aarons
7) roberto gagliardini
8) antonino gallo
9) juan foyth
10) caleb okoli
```

Conclusioni

Il progetto FootballPlayers ha dimostrato un'efficace integrazione di tecniche di machine learning e analisi dei dati per la classificazione, il clustering e la raccomandazione dei giocatori. Attraverso l'uso di algoritmi di classificazione come Gaussian Naive Bayes e Random Forest, è stata ottenuta una buona accuratezza nella previsione dei ruoli dei giocatori, sebbene alcune metriche di precisione richiedano ulteriori ottimizzazioni. La segmentazione dei giocatori in cluster omogenei, eseguita con K-Means e supportata da analisi esplorative, ha rivelato pattern interessanti nelle caratteristiche dei giocatori, ma l'uso di DBSCAN potrebbe offrire ulteriori approfondimenti in scenari con densità di dati variabile. Il modulo di raccomandazione ha implementato la distanza del coseno per suggerire giocatori simili, mostrando risultati promettenti nella personalizzazione delle raccomandazioni.

Tuttavia, il progetto potrebbe essere migliorato nel tempo sotto differenti aspetti. Ad esempio, l'implementazione di algoritmi di clustering più avanzati o la rifinitura dei parametri del modello di raccomandazione potrebbero migliorare ulteriormente la precisione e la qualità delle raccomandazioni. Inoltre, l'integrazione di fonti di dati aggiuntive e l'adozione di tecniche di deep learning potrebbero arricchire il sistema, offrendo nuove prospettive per l'analisi e la predizione delle prestazioni dei giocatori. Questi aspetti rappresentano aree promettenti per future estensioni e miglioramenti, offrendo opportunità per approfondire la comprensione e l'analisi del mondo del calcio attraverso approcci più sofisticati e dati più completi.

Riferimenti Bibliografici

[1] Russell, S., & Norvig, P. (2020). *Artificial Intelligence: A Modern Approach* (4th ed.). Pearson. [Link al libro]