MMetaAnalysis

Progetto di Machine Learning
Prof. Giuseppe POLESE, Prof.ssa Loredana CARUCCIO
Sviluppato da: Antonio Ceruso MAT:[0512116285]
Università degli Studi di Salerno | A.A. 2024-2025

Scenario preso in esame

Lo scenario analizzato per lo sviluppo di *MMetaAnalysis* riguarda il **meta** (*metagioco*) di *Magic: The Gathering* e l'oscillazione del valore monetario delle carte in base al loro utilizzo nei mazzi competitivi con maggiore presenza in tornei regionali, internazionali e mondiali.

In particolare, l'analisi si concentra su dati provenienti da tornei ufficiali del formato **Pioneer**, una delle modalità di gioco più seguite all'interno di *Magic: The Gathering* (*MTG*), uno dei TCG (*Trading Card Game*) più longevi e influenti della storia.

MTG è un gioco di carte strategico uno contro uno, creato nel 1993 da Richard Garfield e pubblicato da Wizards of the Coast. Attualmente conta oltre **20.000 carte uniche**, suddivise in numerose espansioni. Il **meta**, ovvero l'insieme delle strategie più efficaci in un determinato

periodo, ha un impatto diretto sul valore delle singole carte, influenzato da diversi fattori chiave:

- **L’uscita di nuove espansioni** → Ogni espansione introduce nuove carte che possono modificare l’equilibrio del meta, rafforzando archetipi esistenti o creando nuove strategie.
- **Le Ban List** → Liste periodiche che vietano o limitano l’utilizzo di determinate carte nei tornei competitivi, influenzando così la costruzione dei mazzi.
- **Tornei ed eventi ufficiali** → La presenza di giocatori di alto livello che competono in eventi prestigiosi può determinare l'affermazione di determinati mazzi e archetipi, aumentando la richiesta di specifiche carte.

Il progetto *MMetaAnalysis* si concentra proprio su quest'ultimo aspetto, analizzando come l’andamento dei tornei influenzi la domanda e, di conseguenza, il prezzo delle carte.

Il valore di una carta in un TCG è fortemente influenzato dalla **domanda** che c'è per essa, e la domanda, a sua volta, dipende dalla sua rilevanza nel meta.

- **Se una carta diventa cruciale** per un mazzo di successo nel meta (ad esempio, una carta che è particolarmente potente o che interagisce bene con altre carte in un mazzo vincente), **la domanda per quella carta aumenterà**, portando a un aumento del suo prezzo.
- **Se una carta perde rilevanza** nel meta, per esempio perché viene sostituita da una carta più potente o perché il mazzo in cui veniva utilizzata non è più competitivo, **la domanda diminuirà**, causando una discesa nel suo prezzo.

I mazzi di *Magic: The Gathering* si basano su **archetipi**, ovvero insiemi di carte accomunati da una strategia specifica. L’**archetipo** definisce il piano di gioco di un mazzo, influenzando le sue combinazioni di carte e il modo in cui cerca di ottenere la vittoria.

Dataset

Il Dataset utilizzato per la realizzazione del task è ***Magic: The Gathering - Winning Pioneer Decks*** pubblicato da George reperibile tramite seguente link: https://www.kaggle.com/datasets/scarfsman/magic-the-gathering-winning-pioneer-decks

Il dataset era composto di:

- 2306 Entry
- 16 Colonne

Ogni colonna rappresentava sia informazioni circa la carta in se e sia le circostanze di utilizzo. Di seguito sono elencate tutte le colonne.

Nome colonna	Descrizione colonne
Card	Il nome della carta
Quantity	Il quantitativo di quella carta utilizzato in un deck
Pilot	Identificativo anonimo del giocatore
Archetype	Archetipo specifico di quella carta
Evento	Evento in cui il deck ha fatto una buona posizione in classifica
Date Posted:	Data in cui la deck list è stata caricata nel dataset

Main/Sideboard	Identifica se la carta faceva parte della sideboard o della mainboard
Mana Value	Il costo totale della carta
Mana Cost	Il costo effettivo esplicitato sulla carta
Colours	Il colore della carta
Most Recent Printing	Stampa più recente di quella carta
Card Text	Effetto della carta
Type line	Tipi e sottotipi della carta
Price EUR	Prezzo in euro della carta
Price USD	Prezzo in dollari della carta
Rarity	Rarità della carta

Esempio oggetto preso in esame

Problemi rilevati e soluzioni

- **Rimozione Features**: Il dataset includeva diverse feature considerate non pertinenti ai fini delle previsioni, tra cui l'effetto della carta, la data di aggiunta al dataset e l'informazione relativa all'ultima stampa. Inoltre, la colonna relativa al colore della carta è stata eliminata poiché conteneva informazioni ridondanti già esplicitate dalla colonna mana cost.

  Per migliorare l'accuratezza delle previsioni e semplificare il modello, tutte queste feature irrilevanti sono state rimosse.

- **Presenza di valori nulli**: Per le colonne Price USD, Price EUR e Mana Cost, è stata applicata un'imputazione dei valori mancanti utilizzando la media. In particolare, la colonna Mana Cost presentava il 27% di valori nulli, rendendo necessaria questa

operazione per garantire coerenza e completezza nel dataset.

- **Sbilanciamento delle classi**: il dataset presentava una forte disomogeneità nella distribuzione delle classi, con alcune rappresentate da appena 25 entry, mentre altre arrivavano fino a 350. Questo squilibrio avrebbe potuto influenzare negativamente le prestazioni del modello, portandolo a favorire le classi più rappresentate a discapito di quelle meno frequenti. Per risolvere il problema, è stato applicato l'algoritmo ADASYN (Adaptive Synthetic Sampling), che genera dati sintetici aumentando il numero di entry nelle classi meno rappresentate. In particolare, sono state incrementate le classi con meno di 50 entry, ponendo un limite massimo di 50 elementi generati per ciascuna di esse. L'imposizione di questo vincolo è stata fondamentale per evitare un'eccessiva sovrapposizione di dati sintetici, che avrebbe potuto portare a un overfitting del modello, compromettendone le prestazioni del modello.
- **Preparazione classi per il training:**Dopo aver caricato il dataset pulito, le variabili categoriche vengono convertite in formato numerico utilizzando il Label Encoder, che assegna un valore univoco a ciascuna categoria. In particolare, la colonna *Archetype* viene codificata separatamente e il suo encoder viene salvato per un utilizzo futuro.

  Per garantire coerenza nei dati, viene applicata anche la One-Hot Encoding dove necessario, trasformando le categorie in vettori binari per evitare che il modello interpreti erroneamente le etichette come valori ordinali.

  Successivamente, le variabili numeriche vengono normalizzate con StandardScaler per uniformare le scale dei dati e migliorare le prestazioni del modello. Infine, il dataset trasformato viene salvato in un nuovo file, pronto per essere utilizzato nell'addestramento del modello di Machine Learning.

Progettazione del modello

Per lo sviluppo del progetto è stato utilizzato il linguaggio di programmazione Python, sfruttando L'IDE PyCharm messo a disposizione da JetBrains.

Le librerie adoperate nello specifico sono:

- **Matplotlib**: per la generazione di tutti i grafici (confusion matrix, ROC Curve, ecc.)
- **pandas**: per la creazione di strutture DataFrame
- **Scikit-Learn:** per il training del modello
- **Pickel**: per la serializzazione e deserializzazione di oggetti
- **Seaborn**: per la creazione di grafici più complessi ed esplicativi
- **Joblib**: utilizzata per il salvataggio e caricamento dei file pkl (utilizzati per il LabelEncoder, One-Hot-Encoder)
- **Imblearn**: utilizzata per la gestione dell'oversampling con ADASYN
- **Numpy**: utilizzata per calcolo scientifico e l'elaborazione di dati

Il primo modello di Machine Learning sviluppato in **MMetaAnalysis** affronta un **problema di classificazione multiclasse**. La variabile target, **"Archetype"**, può assumere diversi valori, ciascuno corrispondente a una specifica classe. Per risolvere questo problema, viene utilizzato un **Random Forest Classifier**, un algoritmo di apprendimento supervisionato basato su una foresta di alberi decisionali. Nel contesto del progetto, il modello sfrutta **100 alberi decisionali** per prevedere l'archetipo di ogni singola carta con maggiore accuratezza e robustezza.

Il secondo modello, invece, è progettato per un **problema di regressione**, in cui l'obiettivo è prevedere il **prezzo** delle carte in **USD o EUR**. Per questa task viene impiegato un **Random Forest Regressor**, un regressore ad alberi decisionali che, anche in questo caso, utilizza una foresta di **100 alberi** per stimare il valore della carta. Grazie alla sua capacità di cogliere relazioni non lineari tra le variabili, questo modello fornisce previsioni più affidabili rispetto ai metodi di regressione tradizionali.

Configurazione del modello

Per il modello di classificazione multiclasse è stata utilizzata la **k-fold cross validation a k = 10**. Lo stratified K-Fold garantisce che ogni fold abbia una distribuzione bilanciata delle classi, evitando problemi di sbilanciamento nei sottoinsiemi di training e test.

Per il modello di regressione per la predizione dei prezzi è stata testata sia la tecnica GridSearch e sia la tecnica k-fold cross validation con k=10. Entrambe le tecniche hanno mostrato un drastico calo di performance con un sostanzioso aumento di costo in termini computazionali.

Analisi delle performance

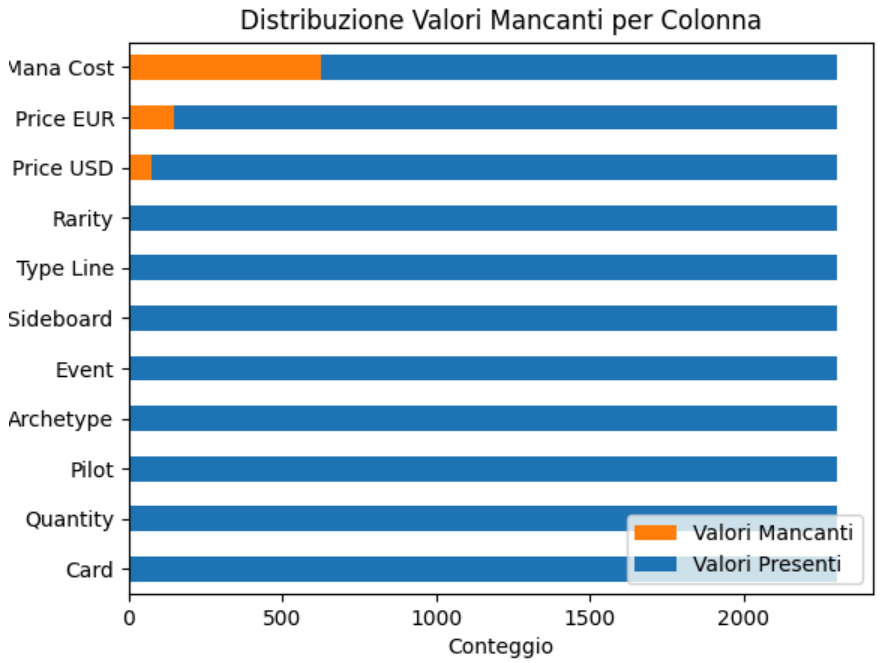
Di seguito vengono riportate le principali metriche risultanti dall'esecuzione del modello, anche mediante l'utilizzo di grafici autogenerati tramite librerie python.

N.B. Per il modello sul problema di classificazione multiclasse è stata posta maggior attenzione sul valore della **Balanced Accuracy.** Tale metrica tiene conto dell'equilibrio tra le diverse classi il che la rende particolarmente utile quando si ha a che fare con dataset sbilanciati.

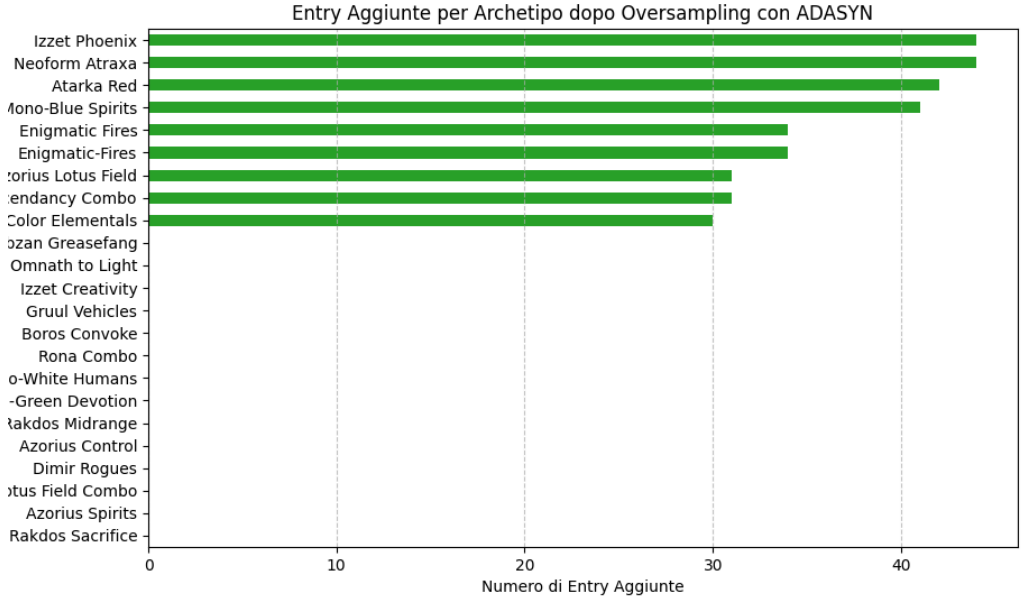
$$\text{Balanced Accuracy} = \frac{1}{N} \sum_{i=1}^{N} \text{Recall}_i$$

Tale metrica nel caso specifico del progetto ha un valore di: 0.8540, il quale è considerabile accettabile conoscendo la natura intrinseca del dataset.

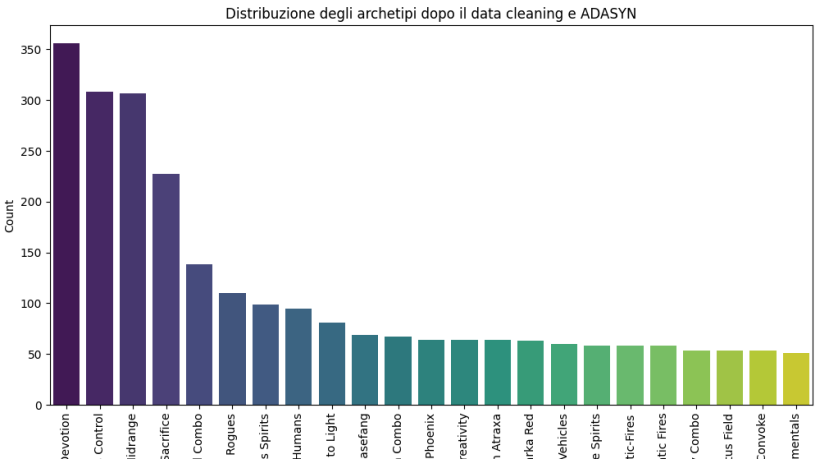
Distribuzione valori mancanti



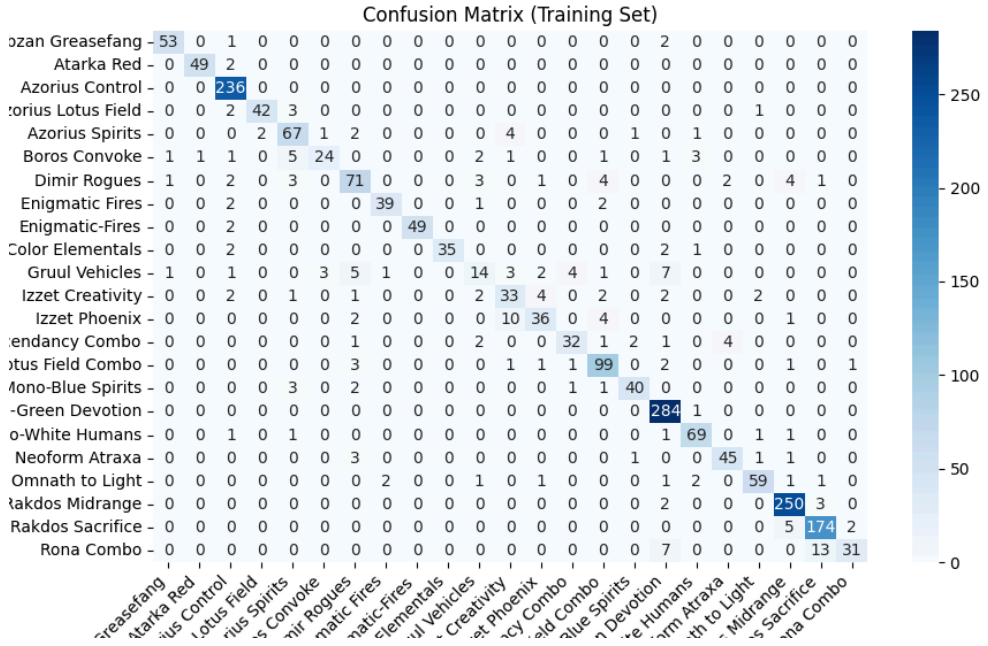
Numero di entry aggiunte dall'oversampling



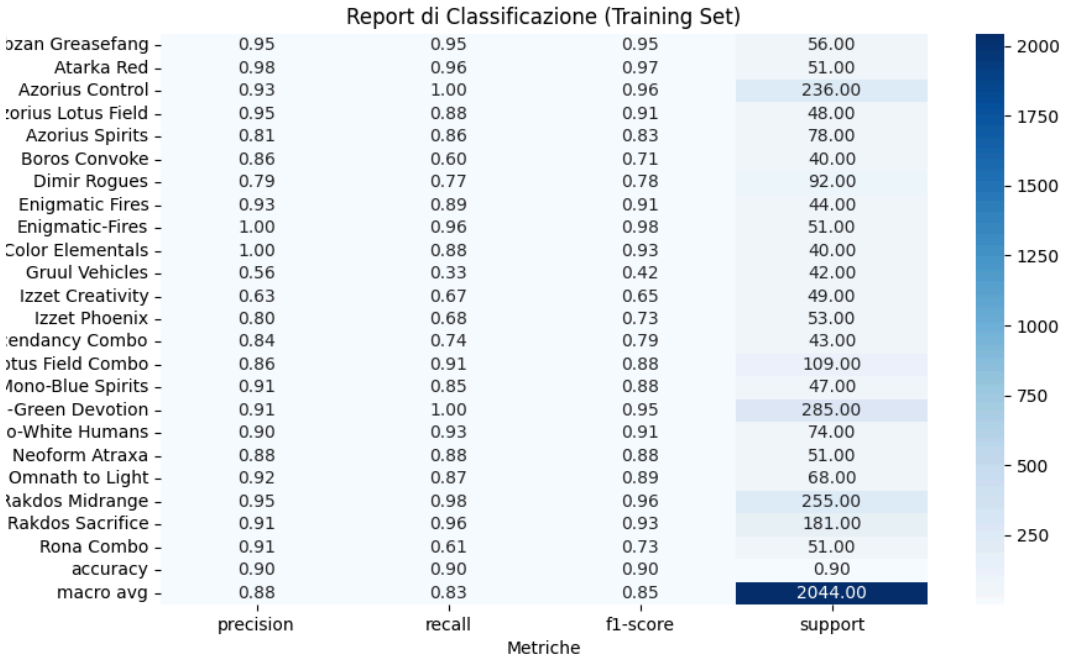
Distribuzione post Oversampling ADASYN



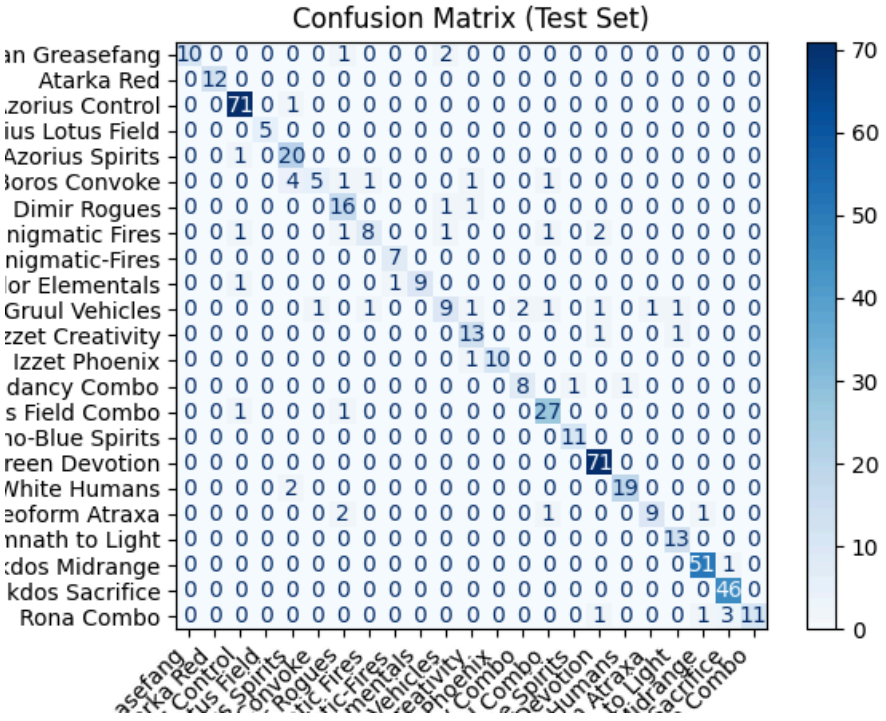
Confusion Matrix Training



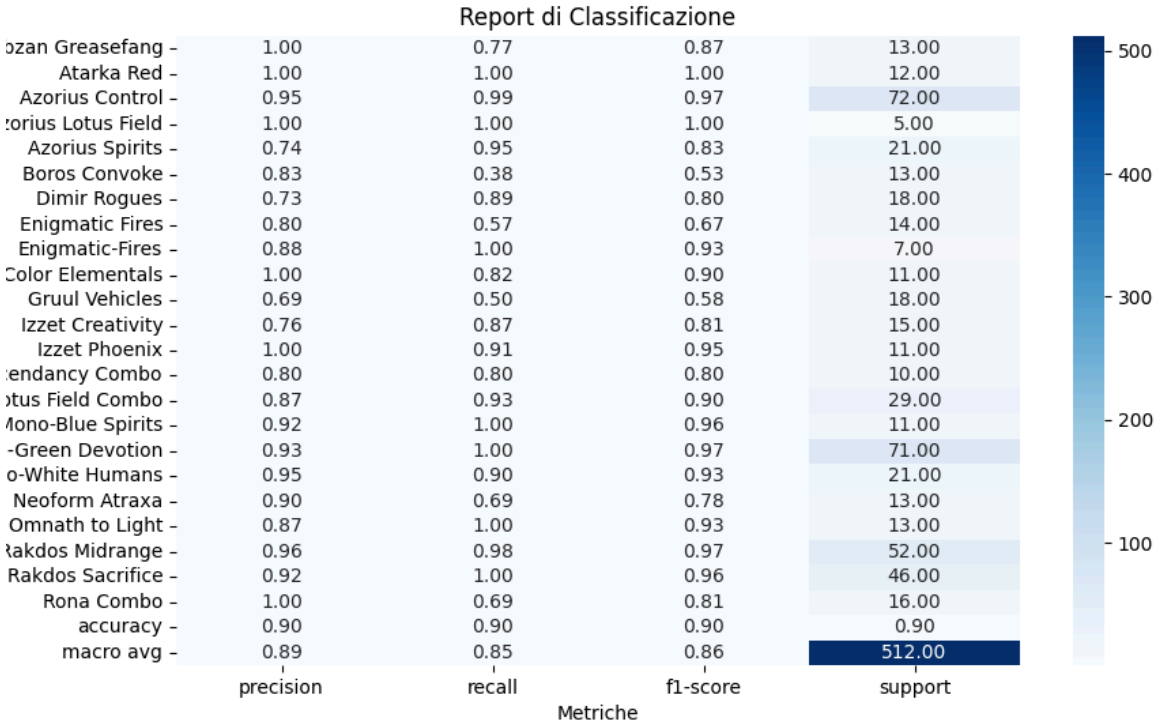
Classification Report Training



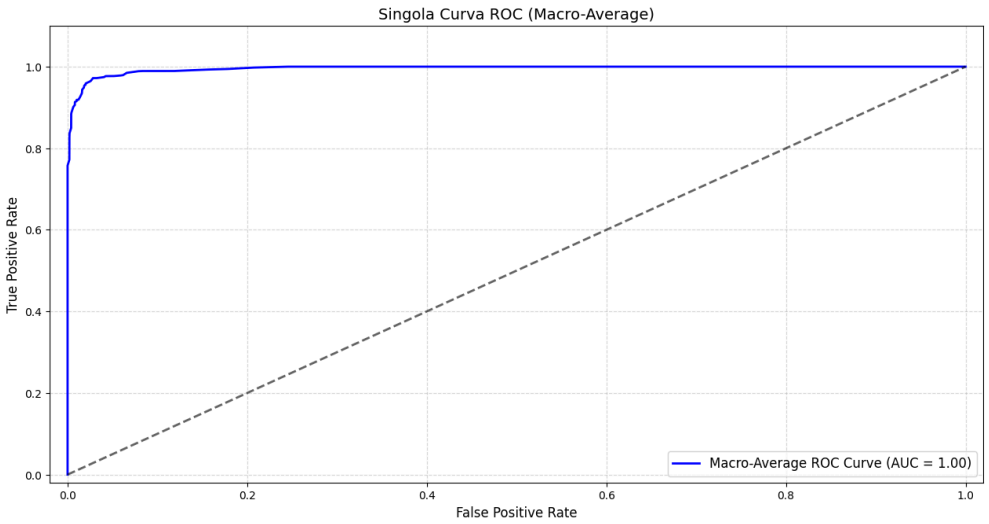
Confusion Matrix Testing



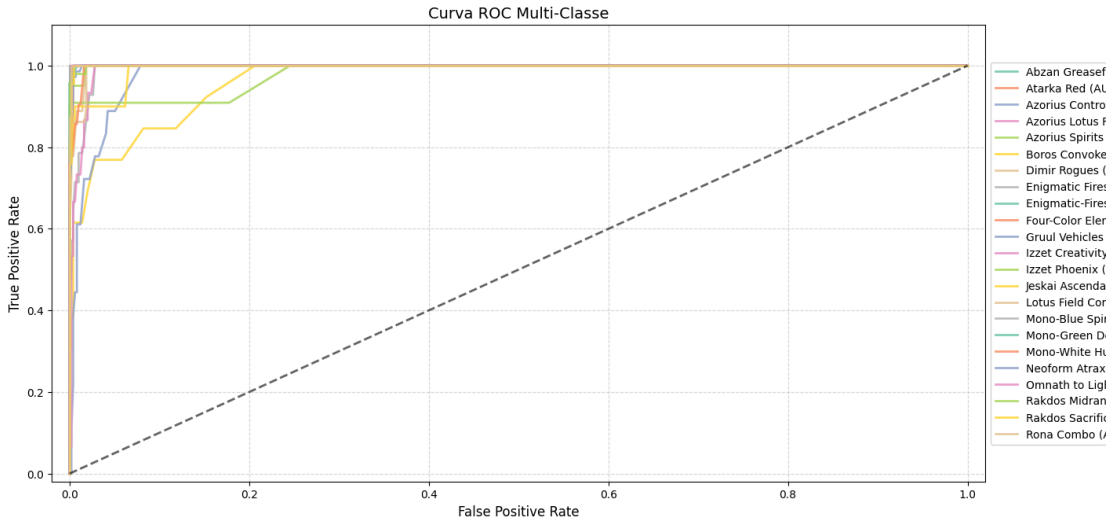
Classification Report Testing



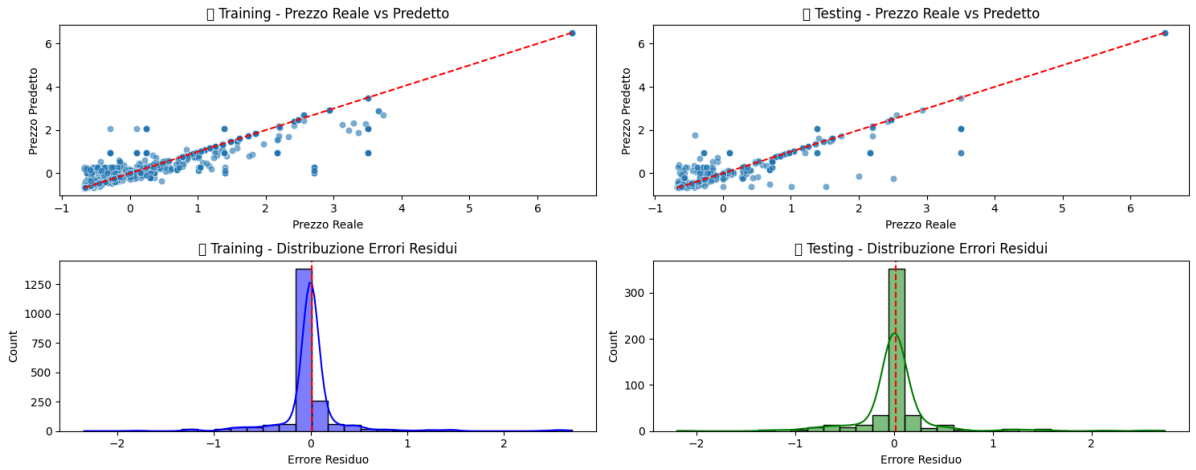
Curva ROC Singola



Curva ROC Multiclasse



Analisi performance sotto progetto per la predizione dei prezzi



Conclusioni

In conclusione analizzando le performance dei due modelli sono stati rilevati degli aspetti positivi ma sicuramente anche delle aree sulla quale

è possibile applicare un sostanziale miglioramento.

Aspetti positivi

Per il problema di classificazione, il modello basato su Random Forest Classifier ha riscontrato una Balanced Accuracy dello 0.8540 nella classificazione degli archetipi. Considerando che il dataset presentava degli sbilanciamenti abbastanza evidenti, questo risultato mette in luce delle scelte di progettazione valide. Un esempio è sicuramente l'oversampling mediante ADASYN, il quale ha migliorato la rappresentatività delle categorie meno frequenti.

Per sotto-problema di regressione, il modello basato su Random Forest Classifier ha dimostrato di essere efficace nella predizione del prezzo delle carte in EUR/USD, cogliendo relazioni non lineari tra le variabili. L'uso di One-Hot Encoding e StandardScaler ha permesso di migliorare la qualità dei dati in input, normalizzando le feature numeriche e codificando correttamente quelle categoriche.

Essenziale è stato l'utilizzo della k-10-fold cross validation, la quale ha consentito una maggiore suddivisione delle classi e quindi della loro rappresentatività ai fine del testing e del training.

Aree di miglioramento

L'utilizzo di un Dataset più corposo in termini di entry, oppure fare affidamento a dati sulle tendenze di mercato, come vendite su piattaforme come Cardmarket o TCGPlayer, potrebbe sicuramente migliorare la capacità predittiva del modello.

L'utilizzo di modelli più avanzati come quelli basati sul Deep Learning o Gradient Boost nel caso della regressione avrebbe portato allo sviluppo di un modello più solido e consistenze.

Il conclusione il modello ha ottenuto risultati promettenti, ma con ulteriori ottimizzazioni e un dataset più ampio, potrebbe diventare ancora più accurato e performante.