

# Modulo di IA

## Progetto combinato con IS: NewDM

## Sezione: Apprendimento

## Autori: Cirillo Franco, Cirillo Luigi, Fusco Ciro, Aiello Vincenzo

# Introduzione

## **Ambito del modulo**

Il nostro modulo è utilizzato nel prodotto software NewDM, una Piattaforma Desktop per la gestione di un punto vendita di un supermercato. Il modulo è stato progettato per fornire un supporto all’attività del magazziniere, che tra le tante mansioni deve anche gestire le richieste di rifornimento per il supermercato.

## **Scopo del modulo**

Il suddetto modulo di IA nasce allo scopo di fornire delle previsione sulle quantità di prodotti realmente utili per il magazzino, al fine di fornire dei consigli per l’acquisto di nuove forniture. In particolare il magazziniere andrà ad inserire i dati richiesti e il sistema restituirà la quantità da acquistare per quel dato prodotto.

# Soluzione proposta

La soluzione è stata proposta inizialmente attraverso un notebook jupiter scritto in python, al fine di valutarne tutti gli aspetti e l’ accuratezza. Al termine è stato tradotto in java attraverso l’utilizzo di un Porter.

Per effettuare la predizione è stato utilizzato il classificatore Random Forest della libreria sklearn. In assenza di dati su cui effettuare l’ addestramento di questo classificatore abbiamo provveduto alla scrittura di un dataset facendolo con la massima accuratezza in modo da renderlo il più realistico possibile.

Il dataset è formato da i seguenti campi:

* tipologia(FruttaVerdura, Pesce, Carne, Casa, Elettronica)
* stagione(Estate, Inverno, Primavera, Autunno)
* zona supermercato(Periferia, Residenziale)
* festività(Feriale, Lavorativo)
* scadenza(Breve, Media, Lunga)
* dimensione confezione(Piccola,Media,Grande)
* costo(prezzo)
* spedizione(prezzo)
* fornitura(20-50-100-150-200-300-500)

Queste rappresentano tutte le variabili su cui deve basarsi il classificatore e che quindi devono essere immesse da input, l’ultima è l’ etichetta che deve restituire il classificatore.

Raggiungibile al link:

# Notebook

In allegato il pdf del notebook, dove è presente il codice commentato e tutti i grafici risultanti dai dati per una migliore chiarezza.

In breve

1. caricamento dataset
2. divisione in training set e test set
3. addestramento della Random Forest
4. calcolo dell’accuratezza sui dati di test
5. calcolo dell’importanza di ogni feature e visualizzazione con grafico
6. estrapolazione di un albero dalla Random Forest
7. visione dettagliata di una parte dell’albero
8. traduzione in java

# Motivazioni delle scelte

La scelta è ricaduta sulla Random Forest perché essa risponde bene alle esigenze della situazione, cioè gestire le risposte relative ad una serie di domande al fine di restituire l’ etichetta giusta per i dati in input.

Infatti questo classificatore si presta molto bene ai problemi decisionali ed è un classificatore n-ario completamente automatizzabile. Lavora bene con valori numerici e categorici e non è influenzato da valori anomali, quindi risponde alle esigenze del nostro dataset.

Le foreste casuali generalmente forniscono un'elevata precisione poiché il principio del modello è quello di calcolare la media dei risultati tra i molteplici alberi decisionali che costruisce.