

# Progetto Icon

Giuseppe Sancesario 706408  
Pierpaolo Ventrella 643917  
A.A. 2020/2021

[Github](#)  
[Repository](#)



# PROGETTO

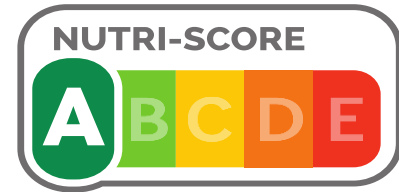
Il sistema da noi progettato è in grado di analizzare i valori nutrizionali basilari (kCal, Proteine, Grassi e Carboidrati) di un alimento in input e stimare un valore di nutriscore (A-E), predire se l'alimento sia salutare o meno e consigliare degli alimenti simili nutrizionalmente a quello fornito dall'utente.



# NUTRIScore

Il nutriscore è un sistema di etichettatura dei prodotti alimentari per semplificare l'identificazione dei valori nutrizionali di un alimento, utilizzando due scale correlate: una cromatica divisa in 5 gradazioni ed una alfabetica.

Il calcolo del punteggio tiene conto di sette diversi parametri di informazioni nutritive per 100 g di cibo e 100ml di bevande. Un alto contenuto di frutta e verdura, fibre e proteine promuove un punteggio più alto, mentre un alto contenuto di energia, zucchero, acidi grassi saturi e sodio si traduce in un punteggio negativo.



# **STRUMENTI SOFTWARE**

# Strumenti Software

**IDE:**PyCharm

**Linguaggio utilizzato:**Python

**Librerie utilizzate:**

- **Sklearn:** costruzione del KNN-classifier, Random Forest per il task di classificazione, K-Means per l'individuazione degli alimenti simili
- **Pgmpy:** creazione di una Rete Bayesiana per il calcolo probabilistico sulla bontà in termini nutrizionali dell'alimento

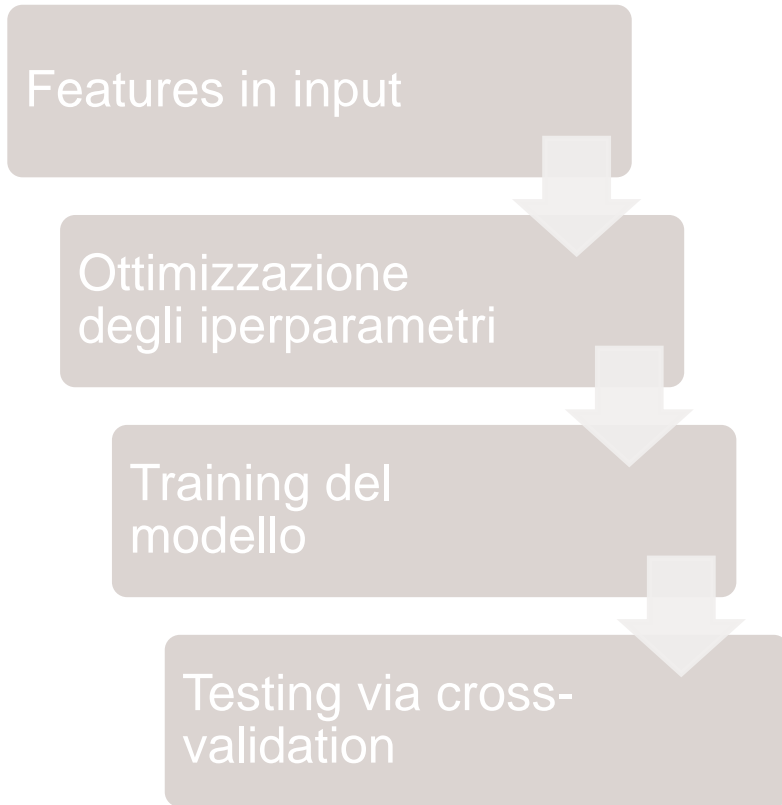
# Dataset utilizzato

Il dataset di partenza da noi utilizzato è stato scaricato dal seguente sito: [Open Food Facts](#)

È stato poi necessario un editing per la rimozione di duplicati e per cercare di ottenere una copertura su ogni categoria di alimento, ottenendo un totale di 648.

Dataset:   
File con valori  
separati da virgola (CSV)

# ALGORITMI DI PREDIZIONE SUPERVISIONATI



- **Input:** energy\_100g, fat\_100g, saturated-fat\_100g, monounsaturated-fat\_100g, polyunsaturated-fat\_100g, omega-3-fat\_100g, carbohydrates\_100g, sugars\_100g, fiber\_100g, proteins\_100g, salt\_100g, sodium\_100g, vitamin-a\_100g, vitamin-d\_100g, vitamin-e\_100g, vitamin-c\_100g, vitamin-b1\_100g, vitamin-b2\_100g, vitamin-pp\_100g, vitamin-b6\_100g, vitamin-b9\_100g, vitamin-b12\_100g, pantothenic-acid\_100g, potassium\_100g, calcium\_100g, phosphorus\_100g, iron\_100g, magnesium\_100g, zinc\_100g, iodine\_100g
- **Target:** nutriscore\_grade



# KNN-CLASSIFIER



Dopo una fase di ottimizzazione degli iperparametri, è stato effettuato il training ed il del modello, tramite k-fold cross-validation (26 folds).

Sono state utilizzate 25 folds per il training e la restante per il testing, per un totale di 26 risultati di accuracy. I valori così ottenuti sono stati mediati.



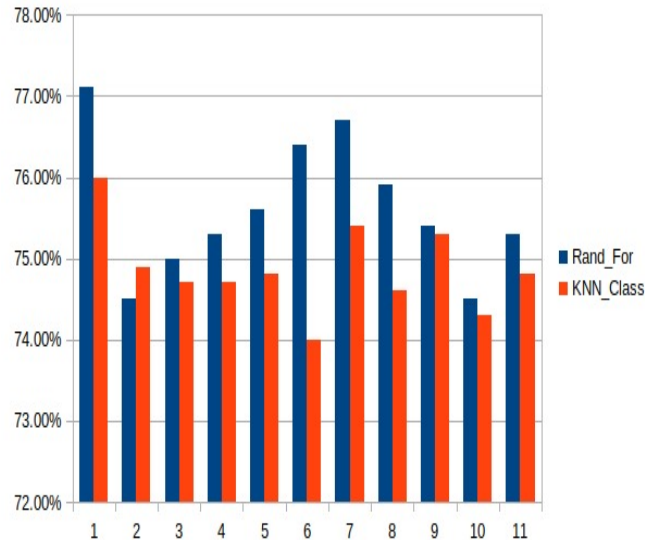
# RANDOM FOREST

Anche in questo caso abbiamo ottimizzato i parametri, con la stessa tecnica vista precedentemente, ed abbiamo effettuato la k-fold cross validation, per mediare i vari risultati di accuracy.

Le fold utilizzate sono le stesse utilizzate per il KNN.

Abbiamo poi variato il contenuto delle folds per 11 volte, ricalcolando i risultati ottenuti da entrambi gli algoritmi, e mediato ulteriormente i risultati ottenuti.

# DECISIONE



**KNN**

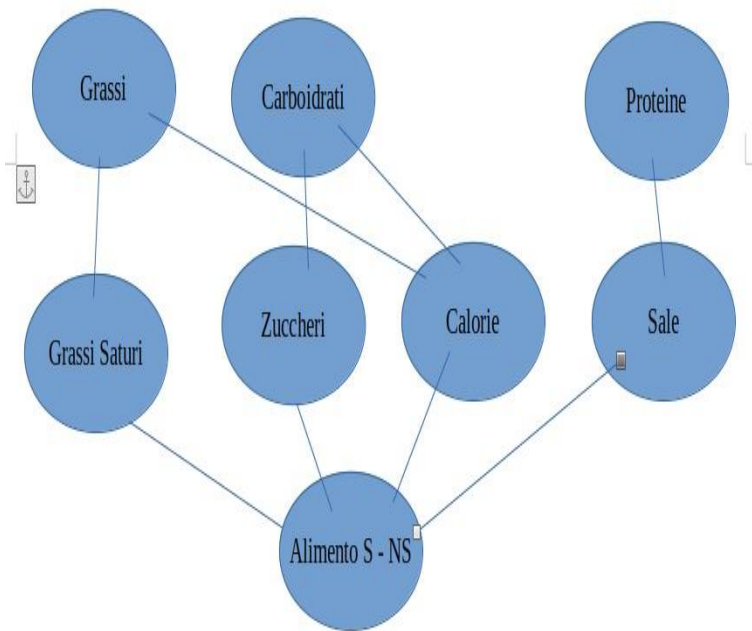
**74.86%**

**RF**

**75.61%**

Dopo aver analizzato i risultati così ottenuti, si è deciso di utilizzare il classificatore KNN.

# RETE BAYESIANA



**PREPROCESSING:** i valori delle feature sono processati modificandoli in valori compresi tra 0 e 4 per permettere il funzionamento del modello probabilistico.

## IL MODELLO

Ha lo scopo di predire se l'alimento inserito in input dall'utente sia salutare o meno, con una certa probabilità. Le dipendenze tra le features sono state individuate utilizzando la matrice di correlazione.

Successivamente è stata utilizzata la k-fold cross-validation per calcolare l'accuracy del modello.

**L'algoritmo ha un accuracy media del 78 %**

# ALGORITMO DI CLUSTERIZZAZIONE

**Input:** energy\_100g, fat\_100g,  
carbohydrates\_100g, proteins\_100g

## K-Means

L'algoritmo tiene in considerazione le seguenti features: calorie, carboidrati, proteine, grassi, sale e suddivide il dataset in cluster. Dopo aver acquisito il nuovo alimento dall'utente, il sistema dovrà restituire una serie di alimenti simili a quello dato in base al cluster.

