Progetto data e web mining

Corò Elia

Matricola: 892507

Modelli scelti

Per questo progetto sono stati scelti i seguenti modelli:

- Random forest
- Artificial Neural Network
- LightGBM

Pulizia e manipolazione del dataset

Prima di passare i dati ai modelli, sono stati eseguiti i seguenti passaggi di pre-elaborazione:

1. Lettura e pulizia del dataset

Sono state **rimosse** le **colonne assenti nel set di test**, così come **tutte le colonne** relative alla variabile ***season***, in quanto proprio per la loro natura non forniscono un contributo al modello predittivo.

2. Eliminazione di righe con un numero eccessivo di valori mancanti

Per evitare che istanze scarsamente rappresentate introducano rumore nel modello, sono state **eliminate le istanze** con una **percentuale di valori mancanti superiore al 90%,** escludendo la colonna *sii*.

3. Eliminazione di valori anomali

È stato generato il box plot delle features e, **le istanze con valori anomali**, sono state o **rimosse** o **sostituiti** i singoli valori **con NaN**. I valori minimi e massimi da rispettare sono stati scelti manualmente facendo ricerche o osservandone la distribuzione.

4. Standardizzazione delle variabili numeriche

Tutte le features numeriche sono state **standardizzate**. Sebbene questa operazione non sia strettamente necessaria per modelli come *Random Forest* e *LightGBM*, è utile per la *Artificial Neural Network*.

5. One-Hot Encoding delle variabili categoriali

Le variabili categoriali, ad eccezione di sii e id, sono state convertite utilizzando **One-Hot Encoding** con l'opzione ***drop='first'***. Operazione in realtà non necessaria in quanto non vi erano più colonne categoriali.

6. Suddivisione del dataset in training e test

Il dataset è stato **suddiviso** in **set di training** e **di test**, garantendo che **nel test set non fossero incluse istanze con *sii != NaN***. Per limitare l'impatto della suddivisione sulla distribuzione complessiva dei dati, è stata adottata una percentuale di separazione relativamente bassa, pari a 0,2.

7. Gestione dei valori mancanti tramite imputazione

I valori mancanti nei dataset di training e test sono stati imputati tramite **Iterative Imputation** basata su **Random Forest**.

8. Applicazione della Label Propagation

Per preservare il maggior numero possibile di istanze, è stata applicata la ***Label Propagation*** **nel training set**. Tuttavia, questa operazione ha introdotto un forte **sbilanciamento** del dataset con un aumento significativo delle istanze appartenenti alle classi 0 e 1.
(Sono state fatte delle prove senza utilizzare la label propagation ma rimuovendo le istanze con sii = NaN, per limitare il forte sbilanciamento, ma l'accuracy non è variata significativamente)

9. Analisi della correlazione tra le diverse features

È stato analizzato il plot delle correlazioni tra le feature, filtrando quelle con una correlazione superiore all'85% (Vedere il notebook per il grafico).

In seguito, sono state applicate le seguenti operazioni ai dataset:

1. **Colonne BIA-BIA**:

   Tra le colonne fortemente correlate figurano: BIA-BIA_BMR, BIA-BIA_DEE, BIA-BIA_ECW, BIA-BIA_ICW, BIA-BIA_FFM, BIA-BIA_LDM, BIA-BIA_LST, BIA-BIA_SMM, BIA-BIA_TBW, BIA-BIA_BMI.

   Tra queste è stata mantenuta solo **BIA-BIA_BMI**, poiché rappresenta una misura più generica e completa della composizione corporea. (Successivamente, però, è stata rimossa e sostituita da Physical-BMI)

2. **Colonne fisiche**

   Tra Physical-Waist_Circumference, Physical-Weight, BIA-BIA_BMI, BIA-BIA_Fat e Physical-BMI, è stata mantenuta **Physical-BMI**, poiché ottenuta tramite misurazione diretta, mentre BIA-BIA_BMI è una stima calcolata.

3. **Unione delle colonne**:
   - **FGC-FGC_SRL** e **FGC-FGC_SRR** (Sit & Reach, flessibilità lato sinistro e destro)
   - **FGC-FGC_GSD** e **FGC-FGC_GSND** (Grip Strength, forza della mano dominante e non)

Ciascuna coppia è stata combinata in una nuova colonna **FGC-FGC_SRM** e **FGC-FGC_GSM**, che rappresenta la media delle due misure.

4. **Rimozione delle colonne _Zone**:
Tutte le colonne con il suffisso **'_Zone'** sono state rimosse poiché rappresentano solo categorizzazioni di altre colonne numeriche senza apportare valore aggiunto.

5. **Selezione della colonna PAQ_Total**:
   - **PAQ_A-PAQ_A_Total**: Livello di attività fisica per adulti
   - **PAQ_C-PAQ_C_Total**: Livello di attività fisica per bambini/adolescenti

Poiché un individuo può appartenere solo a una delle due categorie (adulto o bambino/adolescente), allora sono state sostituite con **PAQ_Total** che contiene il valore massimo tra i due valori.

6. **Rimozione di Fitness_Endurance-Time_Mins**:
   - **Fitness_Endurance-Time_Mins**: Tempo totale in minuti del test di resistenza
   - **Fitness_Endurance-Time_Sec**: Tempo totale in secondi del test di resistenza

Poiché **Fitness_Endurance-Time_Sec** è una misura più dettagliata e precisa, è stata rimossa Fitness_Endurance-Time_Mins.

7. **SDS-SDS_Total_T e SDS-SDS_Total_Raw**:
   Tra:
   - **SDS-SDS_Total_T** (punteggio T standardizzato del disturbo del sonno)
   - **SDS-SDS_Total_Raw** (punteggio grezzo del disturbo del sonno)

È stata mantenuta **SDS-SDS_Total_T.**

Alla fine, le uniche correlazioni rimaste sono: **Physical-BMI** e **BIA-BIA_FMI**

Queste due variabili, pur essendo leggermente correlate, rappresentano due misurazioni distinte, perciò le manteniamo separate.

10. Analisi della distribuzione delle etichette e calcolo della baseline accuracy

Al termine di queste operazioni il dataset è stato analizzato e si è riscontrato un **forte sbilanciamento**, soprattutto a seguito della label propagation, con prevalenza delle classi 0 e 1.
È stata quindi definita una **funzione di scoring** basata sull'**inverso della proporzionalità delle classi nel dataset** in modo che le classi 2 e 3 avessero un impatto maggiore.

Inoltre, è stata calcolata la **baseline accuracy**, con un valore di 0.6208, sebbene questa misura debba essere interpretata con cautela a causa dello sbilanciamento.

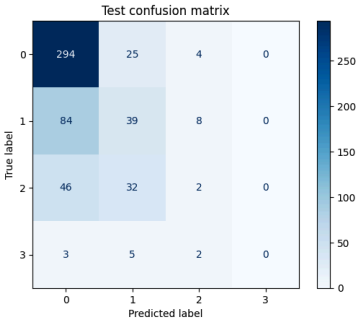
Modelli

Random Forest

È stato testato un primo modello con Random Forest, preceduto da una ***RFECV*** per selezionare le features più importanti e da una **grid search** per trovare i parametri ottimali. Nel modello è stato impostato ***class_weight***="***balanced***" in modo da tentare di **bilanciare il modello**.

I risultati ottenuti sono i seguenti:

	Train	Test
Normale	0.9997	0.6158
Bilanciata	0.9997	0.3082



Si osserva che il modello **non è riuscito a predire** alcuna istanza della **classe 3**, mostrando una netta tendenza a **focalizzarsi** sulla **classe 0**. Questo probabilmente è dovuto al forte sbilanciamento del dataset. Non è riuscito a superare la baseline accuracy.

Artificial Neural Network

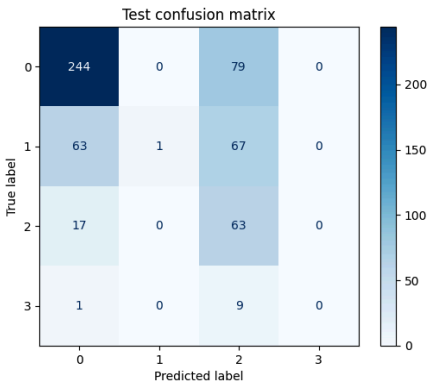
Successivamente, è stato utilizzato un modello **ANN**, con una **grid search** per ottimizzare i seguenti parametri:

- pesi da applicare a ciascuna classe
- numero di neuroni
- numero di layer nascosti
- dropout (attivato al 25% prima dell'ultimo layer)

Come ultima funzione, prima dell'ultimo layer, è stata utilizzata la **softmax** per ottenere il vettore delle probabilità, da cui estrarre l'argmax. Dopo aver trovato la combinazione ottimale di parametri, è stato effettuato un **training** con criteri di **early stopping** e **riduzione del learning rate**.

I risultati ottenuti sono i seguenti:

	Train	Test
Normale	0.6305	0.5662
Bilanciata	0.3980	0.3876



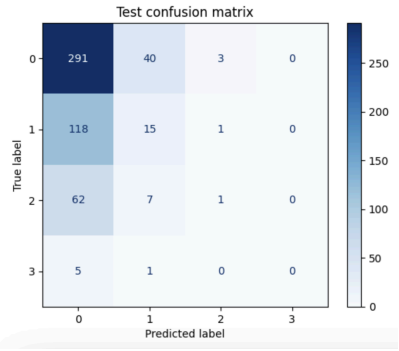
Si osserva che nonostante **commetta più errori** rispetto alla random forest, **l'accuracy bilanciata risulta migliorata**. Ha una tendenza maggiore a classificare in classe 0 e 2 ignorando quasi totalmente le classi 1 e 3. In questo caso non ha superato la baseline accuracy.

LightGBM

Infine è stato realizzato questo modello, preceduto dal **tuning dei parametri** (nei limiti delle capacità di calcolo del computer). Prima di passare i dati al modello, dato che **è in grado di gestire i valori NaN**, è stato rifatto il processo di pulizia iniziale del dataset **senza iterative imputation**.

I risultati ottenuti sono i seguenti:

	Train	Test
Normale	0.9163	0.5129
Bilanciata	0.8890	0.2252



A livello di **accuracy** risulta il **peggiore** tra i tre, probabilmente anche a causa dello scarso tuning dei parametri. Anche in questo caso, come nel modello *Random Forest*, il modello tende a **concentrarsi** principalmente sulle **classi 0**, trascurando quasi totalmente le altre, nonostante la **confusion matrix del train** risulti essere molto **buona**. Non è riuscito neanche in questo caso a superare la baseline accuracy.

Analisi dei modelli

In conclusione, il modello che ha ottenuto la **miglior accuracy** è la **Random Forest**, probabilmente a causa di una focalizzazione sulle classi 0 e 1. Tuttavia, il modello che ha mostrato la miglior **accuracy distribuita** è la **Artificial Neural Network** (ANN).

L'aumento dell'accuracy bilanciata è dovuto all'assegnazione di pesi alle classi, che ha ridotto la tendenza del modello a concentrarsi esclusivamente sulla classe 0, distribuendo meglio le predizioni sulle altre classi, come si può vedere dalla matrice di confusione.

Durante la fase di tuning dei parametri, a causa dei limiti di capacità di calcolo e di tempo, il modello ANN non è stato testato con un numero considerevole di configurazioni e pesi. Pertanto, prolungando il processo di ottimizzazione, potrebbe essere possibile individuare configurazioni migliori e affinare ulteriormente la distribuzione dei pesi, riducendo l'errore complessivo.

Analisi degli errori

È stato analizzato il modello **ANN**, che ha ottenuto la migliore accuracy distribuita.

Sono state quindi **esaminate le istanze** classificate **correttamente** ed **erroneamente**, tracciando i **plot delle loro distribuzioni** e **delle medie** per vedere se vi erano distribuzioni anomale o importanti differenze nelle medie.

Dall'analisi delle distribuzioni, non emergono differenze significative tra le istanze classificate correttamente e quelle erroneamente. In effetti, nella maggior parte dei grafici, le distribuzioni si sovrappongono.

Le uniche caratteristiche che mostrano distribuzioni leggermente differenti tra le istanze classificate correttamente ed erroneamente sono: **Basic Demos-Age**, **Physical-Height** e **Preint_EduHx-computerinternet_hoursday.**

Per queste feature, si osserva che valori più alti tendono ad essere associati a un numero maggiore di errori di classificazione, mentre valori più bassi risultano in una classificazione corretta.

Queste osservazioni sono confermate anche dal secondo grafico.