Salve prof,

lo script inizia con la fase di caricamento e preparazione dei dati. Vengono eliminati gli identificativi degli ospedali (hospital\_id) poiché l'analisi si concentra sulla popolazione di una città intera. Questo dataset infatti si basava sui dati di diversi pazienti in diversi ospedali americani, io ho pensato che ai fini dell’analisi che avevo intenzione di fare fosse opportuno eliminare questa feature per simulare una popolazione di una città di circa 100k individui.

La mia idea era quella di simulare che 5000 contraessero il covid 19 basandosi su una distribuzione probabilistica legata alle condizioni mediche di ciascun individuo. Lo scopo dello script è quello di gestire le risorse limitate, come i letti d'ospedale, in modo efficiente, dando priorità ai casi più gravi basati su età e condizioni mediche, nello specifico il numero di letti da assegnare è di 320, perchè ho letto che in italia in media ogni 100k devono essere garantiti circa 320 posti letto negli ospedali.

Nello script ho applicato dei criteri per assegnare a ciascuno una priorità basandosi sull’età e su un coefficiente chiamato “apache score” che indica il rischio di morte in ospedale.

Il problema che affronto nello script è essenzialmente un problema di minimizzazione delle morti.

L'algoritmo di ottimizzazione optimize\_bed\_allocation\_scipy è utilizzato per assegnare i letti ospedalieri. Questo algoritmo utilizza il linear programming per minimizzare la probabilità di decesso e rispettare le proporzioni desiderate per i gruppi prioritari. Le penalità sono calcolate per correggere le deviazioni dalle proporzioni target di assegnazione dei letti (ho pensato che dovessero esserci delle proporzionalità nell’assegnazione dei letti un 40% circa per i codici rossi, 30% per i codici gialli, 20% per i codici verdi e 10% per i codici bianchi), integrando vincoli che limitano l'assegnazione dei letti solo agli individui malati, e non a tutti gli individui della popolazione.

Mediante il metodo run\_iterations, l'assegnazione dei letti viene simulata ripetutamente per testare la robustezza e l'efficacia del modello di ottimizzazione in diversi scenari. Questa fase fornisce dati preziosi sull'equità e l'efficienza dell'algoritmo di assegnazione.

Poi ho pensato di allenare un modello d machine learning che assegnasse un posto letto di ospedale anche senza runnare l’algoritmo di ottimizzazione.

Prima di procedere con il training del random forest, ho deciso di performare un encoding delle variabili categoriche attraverso l'one-hot encoding e la normalizzazione delle variabili numeriche, che avevo già fatto in precedenza ma ho rifatto, per garantire che il modello tratti tutte le caratteristiche in modo equo, indipendentemente dalla loro scala originale.

Ho deciso poi di performare una PCA (Principal Component Analysis) per ridurre il numero di dimensioni del dataset, facilitando così l'elaborazione dei dati e migliorando la performance del modello, pur mantenendo la maggior parte delle informazioni originali.

Ho allenato poi un modello di Random Forest per predire l'assegnazione dei letti e aggregato i dati al test df.