Laboratorio di Big Data

Progetto – Alberto Mastromarino – 11/82/00302

Airline Delay

1. Obiettivo del progetto

L'obiettivo del progetto è quello di prevedere se un determinato volo aereo farà ritardo o meno, date le informazioni sulla partenza programmata, creando così un modello che possa prevedere il ritardo. Si tratta di un problema di classificazione binaria, difatti all'interno del dataset è presente una variabile target che presenta valori booleani: il valore 1 indica il ritardo di un aereo, il valore 0 indica che un aereo non ha fatto ritardo.

2. Descrizione Dataset

Il dataset su cui è incentrato il progetto, è stato scaricato in formato csv, dal sito Kaggle.com (https://www.kaggle.com/datasets/ulrikthygepedersen/airlines-delay).
Si tratta di un dataset contenente più di 500 mila osservazioni, riguardanti informazioni sulla partenza programmata di voli aerei.
Una volta caricato il file, è stato creato un dataframe formato da 539382 righe e 8 colonne. Nel dettaglio le 13 colonne sono:

1. *'Flight'* codice ID identificativo che si utilizza per identificare ogni singolo.
2. *'Time'* variabile che indica il tempo della partenza di ogni volo; questa variabile si trova espressa in minuti in rapporto ad una giornata di 24 ore, 60 nel dataset corrisponde infatti ad 1, 120 a 2, 180 a 3, fino ad arrivare a 1440 che corrisponde alle 24, ovvero mezzanotte; successivamente si procederà ad una conversione di questi valori numerici per ottenere una variabile contenente 6 categorie ben distinte: Late night, Early morning, Morning, Afternoon, Evening e Night.
3. *'Length'* variabile quantitativa che indica la durata di ogni volo, anch'essa espressa in minuti.
4. *'Airline'* variabile categoriale che indica l'ID degli aeroporti.
5. *'AirportFrom'* variabile categoriale che indica l'ID degli aeroporti di partenza.
6. *'AirportTo'* variabile categoriale che indica l'ID degli aeroporti di arrivo.

7. '*DayOfWeek'* variabile qualitativa ordinale che indica i giorni della settimana in cui partono i voli; lunedì viene indicato con 1, fino ad arrivare a domenica indicato con 7.
8. '*Class'* determina se il volo ha fatto ritardo o meno e identifica la variabile target.

Successivamente si è proceduto ad analizzare il bilanciamento del dataset e si è constatato che fosse abbastanza bilanciato.

3. Data Cleaning

La fase di data cleaning ha lo scopo di assicurarsi che tutte le variabili del dataset siano del tipo corretto, che siano utili ai fini dell'analisi successiva e che non contengano dei valori nulli.

Dopo aver utilizzato la funzione show() per dare una prima visione del dataset, si è proceduto con l'eliminazione della colonna '*Flight'* poiché non utile ai fini dell'analisi.

Dopo aver constatato che il dataset non contenesse valori nulli, è stata analizzata la natura delle variabili presenti nel dataset.

Si è deciso di convertire la variabile *'Length'* da Double Type a Integer Type.

4. Analisi esplorativa

La fase di analisi esplorativa è volta alla rappresentazione grafica delle variabili.
Sono stati realizzati dei grafici che permettono di vedere il rapporto tra la variabile di risposta *"Class"* e le altre variabili presenti all'interno del dataset.

È stato creato un primo count plot che mette in relazione le compagnie aeree con la variabile target; questo grafico ci permette di vedere quali sono le compagnie che fanno più ritardi.

Successivamente un secondo count plot mette in relazione la variabile *'DayOfWeek'* con la variabile target, e questo ci permette di vedere quali sono i giorni della settimana in cui ci sono più ritardi. Questo grafico è stato poi creato in misura percentuale, per vedere le percentuali di ritardo (e non) nei giorni della settimana.

Come quarto grafico si è deciso di creare un barplot che mette in relazione la variabile target con la variabile *'Airline'* in misura percentuale; questo permette di vedere quali sono le percentuali di ritardi (e non) per quanto riguarda le compagnie aeree.

Si è scelto di utilizzare anche un boxplot che mette in relazione la variabile target *'Class'* con la variabile *'Length'*. Da questo grafico è possibile vedere qual è la lunghezza mediana, il valore minimo e il valore massimo sia dei voli che fanno ritardo che dei voli che di quelli che non fanno ritardo.

Infine come ultimi due grafici, sono stati creati due count plot (uno in percentuale) che hanno messo in relazione la variabile *'Time'(*che indica i momenti della giornata in cui partono i voli) e la variabile *'Class'*.

5. Pre-processing

La fase di pre-processing prevede la trasformazione dei dati usando lo Scaling per le variabili quantitative e l'Encoding per le variabili qualitative.

Nello specifico, per la variabile '*Length'* si è scelto di utilizzato il Robust Scaler, uno scaler robusto, che gestisce la presenza degli outlier nella variabile che si sta scalando, e si va ad utilizzare quando si hanno outlier nei dati.

Per le variabili *'Time', 'DayOfWeek'* dato che non erano presenti outliers, si è scelto di utilizzare il MinMax Scaler, lo scaler più comunemente utilizzato, che esegue lo scaling in un range compreso tra un valore minimo ed un valore massimo, che di default sono 0 e 1.

Per le variabili *'Airline', 'AirportFrom', 'AirportTo'* è stato utilizzato inizialmente l'algoritmo String Indexing e poi quello di One Hot Encoding che ha permesso ottenere dei vettori.

Ottenuti i valori trasformati delle variabili, è stata poi applicata un'operazione di Vector Assembler delle sole variabili indipendenti per creare una variabile vettoriale che le rappresenta ed è stata nominata *"Features".*

6. Generazione e test dei modelli

L'ultima parte dell'analisi prevede l'applicazione di modelli di Machine Learning. Prima di fare questo però è stato diviso il dataset in due parti: 80% delle osservazioni è stato casualmente assegnato al training set e il restante al 20% al test set. Questo avviene per fare in modo che il modello venga addestrato su determinati dati di training e la sua performance venga valutata su dati sconosciuti, di test. Sono stati utilizzati tre classificatori per la variabile target.

Regressione Logistica

Il primo modello utilizzato è la regressione logistica, con la quale si è ottenuta un'accuratezza di 62,93%. Il risultato non è pienamente soddisfacente, per cui si è proceduto ad utilizzare altri classificatori. Il modello prende la colonna *'Features'* come variabile indipendente, mentre come variabile dipendente la variabile binaria *'Class'.*

Random Forest Classifier

Inizialmente sono stati utilizzati parametri scelti arbitrariamente maxDepth: 14 e numTrees: 25. A seguito dell'addestramento si è proceduto con la valutazione del modello sui dati di test attraverso un Multiclass Classification Evaluator e si è ottenuta un'accuratezza pari a 63,40%. Tale risultato è leggermente migliore rispetto a quello ottenuto con la regressione logistica.

Naive Bayes

L'ultimo modello che è stato addestrato è un Naive Bayes. Il modello è stato addestrato sui dati del Training set e le performance sono state valutate sul Test set attraverso un Binary Classification Evaluator, ottenendo un'accuratezza inferiore di due punti percentuale rispetto alla regressione logistica, pari a 60,15%.

7. Conclusioni

I modelli di Machine Learning hanno ottenuto dei risultati molto simili, non pienamente soddisfacenti. Questi risultati non ottimali si ottengono probabilmente a causa di un basso numero di variabili all'interno del dataset, 8 variabili ma se ne utilizzeranno 6 (ai fini dell'analisi), probabilmente insufficienti per predire la possibilità che un aereo faccia ritardo o non lo faccia. Dati i risultati ottenuti, potremmo ipotizzare che le variabili non contengano informazioni rilevanti e non veicolino informazioni adeguate ai fini dell'analisi.

8. Algoritmi

Regressione logistica

Con questo modello ho ottenuto una un'accuratezza di**: 0,63**

Random Forest

Con questo modello ho ottenuto un'accuratezza pari a**: 0,63**

Naive Bayes

Con questo modello ho ottenuto un’accuratezza pari a**: 0,60**