Pulizia e Preprocessing dei dati

1. Introduzione

Questo documento ha l'intento di spiegare i passaggi dell'attività di pulizia e preprocessing dei dati.

Verranno descriti nei capitoli successivi quindi tutti i procedimenti e i flussi di pensiero durante l'attività.

La pulizia dei dati si occupa principalmente di eliminare eventuali dati mancanti o duplicati. Per evitare di eliminare grandi quantità di dati è possibile intramprendere una strategia di correzione dei dati, in linea con i parametri necessari oppure contrassegnati da marker specifici.

2. Gestione dei valori nulli

Ci sono diversi approcci per gestire i valori nulli, eliminare le righe corrispondenti oppure inizzializzarli con valori marker o standardizzati. La scelta di uno dei due metodi ricade in base al tipo di dato che si vuole analizzare, dati importanti non possono essere inizzializzati con valori medi o standard conviene eliminarli.

Un altro parametro per la scelta di come procedere e la quantità di dati disponibili e la quantità di valori nulli, grandi dataset che perdono una piccola percentuale dei loro dati posso tranquillamente adottare un approccio di eliminazione dei valori nulli.

Nel mio caso per un totale di 541909 righe sono risultati esserci 1454 valori nulli per quanto riguarda la colonna "Description" e 135080 valori per "CustomerID".

Nonostante i valori nulli di CustomerID siano molti, sono dati fondamentali per analizzare la clientela, per questo motivo sono stati eliminati tutti i valori nulli.

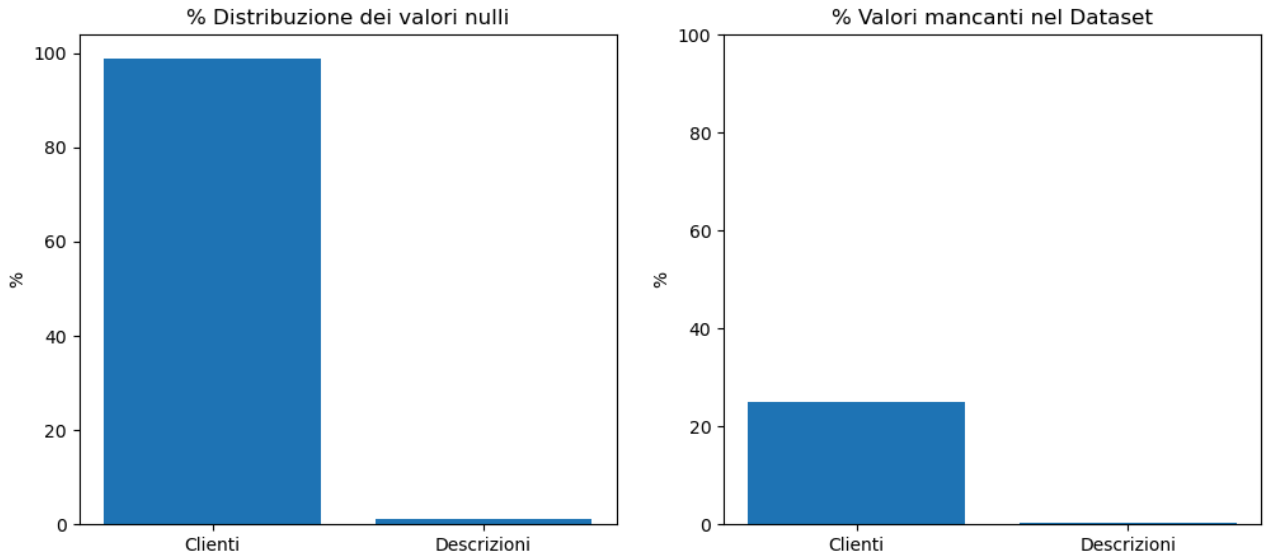

Figure 1: A sinistra la percentuale di valori nulli tra Clienti e Descrizioni, a destra la percentuale di valori nulli nei dati totali.

3. Gestione dei valori duplicati

Analogamente ai valori nulli, è stato eseguito un controllo sulla presenza di eventuali valori duplicati tramite il codice `df.duplicated().sum()`. Sono risultate esserci 5225 valori doppi i quali sono stati eliminati.

Di seguito riporto un grafico con la percentuale di valori duplicati rispetto al dataset (dataset gia pulito dai valori nulli).

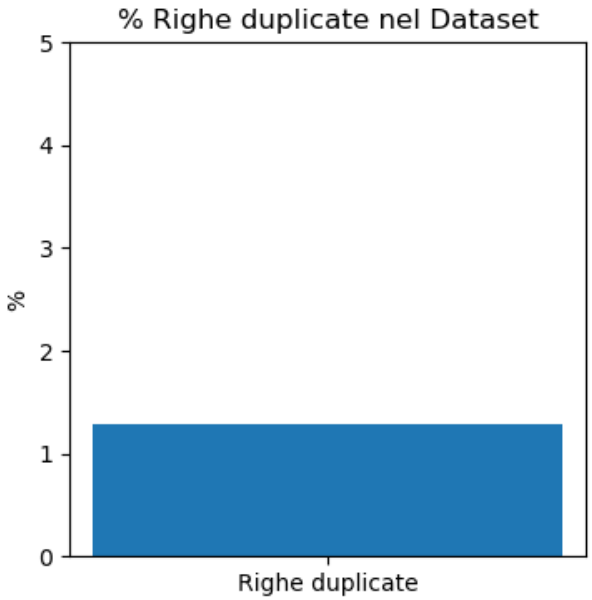

Figure 2: Percentuale righe duplicate sul totale.

4. Ulteriore pulizia

Come ulteriori controlli per la pulizia e integrità di dati, ho voluto cercare eventuali valori negativi di prezzo, di quantità o di cancellazione ordini.

Analizzando le quantità negative è risultato che ogni riga con valore negativo era associata ad un InvoiceNo con iniziale "C". Probabilmente si riferiscono a valori di rimborso o cancellati, per questo motivo le quantità negative sono state eliminate.

Analogamente è risultato esserci molte righe con prezzo unitario di 0, anch'essi sono stati eliminati.

Come ultimo controllo ho voluto riguardare le statistiche base del DataFrame attaulmente pulito tramite `df.describe().T` di seguito un'immagine con i valori restituiti.

	count	mean	std	min	25%	50%	75%	max
Quantity	392692.0	13.119702	180.492832	1.000	2.00	6.00	12.00	80995.00
UnitPrice	392692.0	3.125914	22.241836	0.001	1.25	1.95	3.75	8142.75
CustomerID	392692.0	15287.843865	1713.539549	12346.000	13955.00	15150.00	16791.00	18287.00

Figure 3: Statistiche del dataset.

Ne risultano due valori particolari sia in "Quantity" che in "UnitPrice" nella colonna "max". Dopo un approfondimento si è evinto che i valori di "Quantity" per quanto enormi non erano aberrazioni, bensì "UnitPrice" tutti i valori associati a numeri anomali veniva associato un "StockCode" a "POST". Sono state quindi eliminate tutte le righe associate, probabilmente con riferimento a spedizioni, in ogni caso non era di nostro interesse in quanto vogliamo analizzare i clienti in base ai prodotti.

5. Conclusione

Una volta conclusa la pulizia ho voluto confrontare la dimensione attuale del dataset in confronto con il dataset originale.

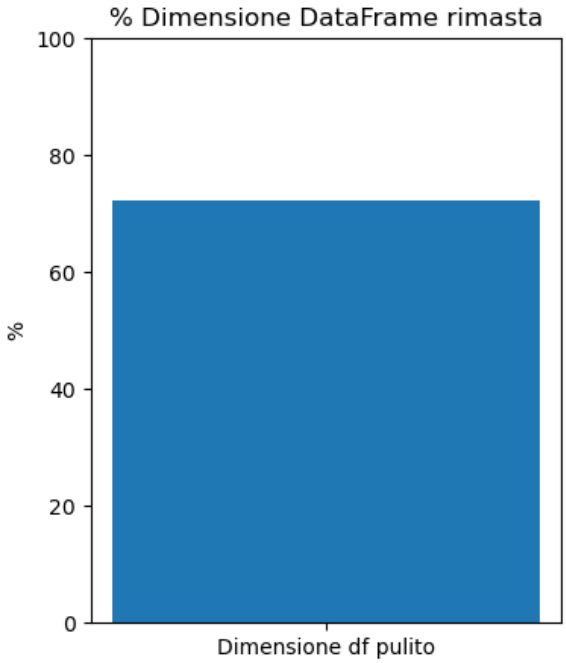

Figure 4: Dimensione del dataset pulito.

La dimensione è del 72,26% rispetto a quello originale, per quanto ci sia stata una diminuzione drastica delle dimensioni presenta ancora quasi 400mila records.