Introduction to Machine Learning - Challenge 1

Autore: Bredariol Francesco

Data di consegna: February 28, 2025

1 Introduzione

In questa challenge andiamo a studiare un dataset contenente dati estratti da immagini di banconote vere e fraudolente. In particolare i dati riguardano la trasformata Wavelet delle immagini da cui sono state estratte le seguenti features:

1. varianza della Wavelet
2. asimmetria della Wavelet
3. curtosi della Wavelet
4. entropia dell'immagine
5. categoria (0 o 1)

2 Pre-elaborazione dei dati ed esplorazione iniziale

La primissima operazione che andiamo ad eseguire è una visualizzazione della tabella dei dati. Da questa operazione scopriamo che il dataset è ordinato e dunque è necessario eseguirne un rimescolamento per poter ottenere dei successivi train e test set etereogenei. Studiamo anche la distribuzione delle classi per comprendere come esse sono bilanciate. Vediamo solo un leggero sbilanciamento verso la classe 0 ma questo non ci preoccupa eccessivamente in quanto non è troppo grande.

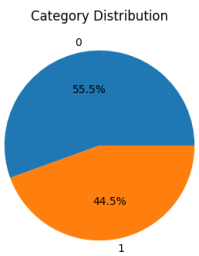


Figure 1: Visualizzo la distribuzione dei dati

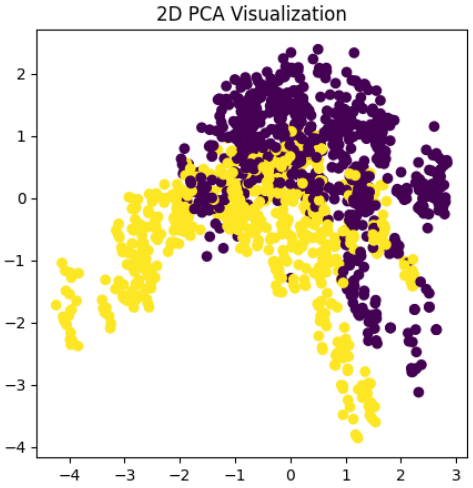
Verifichiamo poi con un test shapiro la gaussianità dei dati. Il test risulta positivo alla gaussinità motivo per cui procediamo alla standardizzazione di questi ultimi.

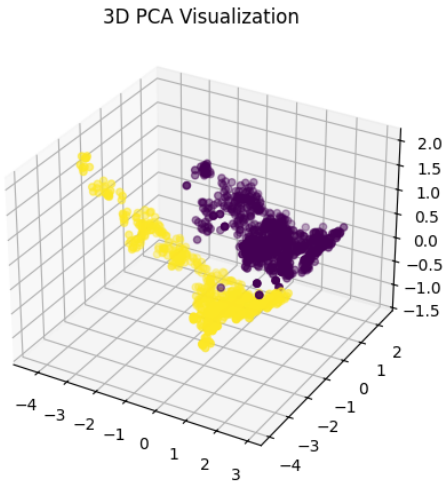
3 Unsupervised learning

Nella prima parte operativa della challenge ci ritroviamo ad esplorare varie tecniche di Unsupervised Learning con l'obbiettivo di eseguire prima una riduzione della dimensionalità del dataset e poi un clustering sugli spazi ottenuti dalle riduzioni di dimensionalità.

3.1 PCA

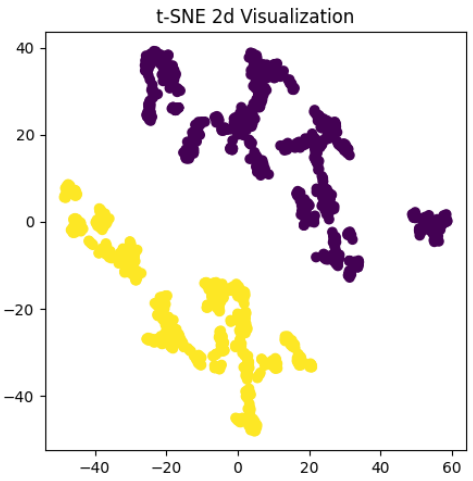
Il primo strumento utilizzato è la PCA. Lo scree test ci indica che usare 2 componenti cattura solo l'85% della varianza mentre 3 componenti più del 90%. Come ci si poteva aspettare i risultati a due dimensioni non sono soddisfacenti mentre quelli a 3 dimensioni già di più.



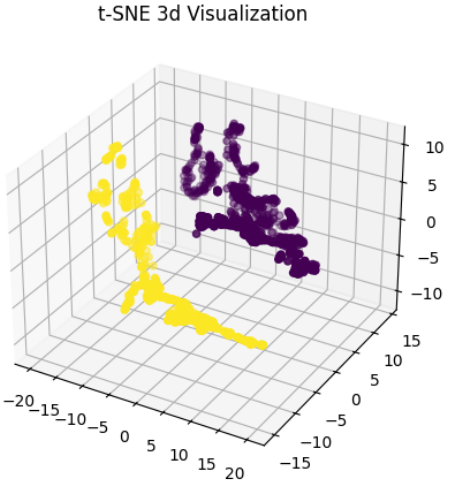


3.2 t-SNE

Eseguiamo poi anche la riduzione di dimensionalità tramite l'uso di t-SNE. Testiamo sia le due dimensioni che le tre dimensioni. Bisogna notare come già la riduzione in dimensionalità 2 è ampiamente sufficiente a creare una separazione netta.



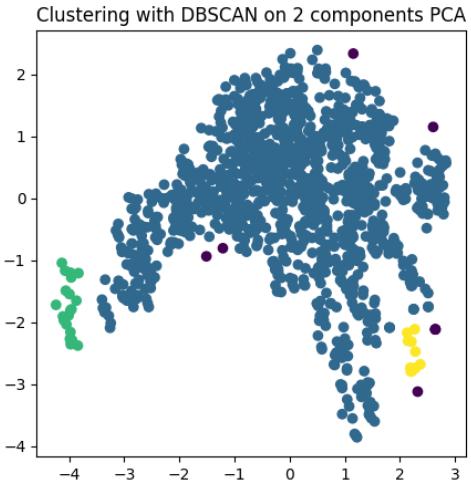
(a) perplexity=30



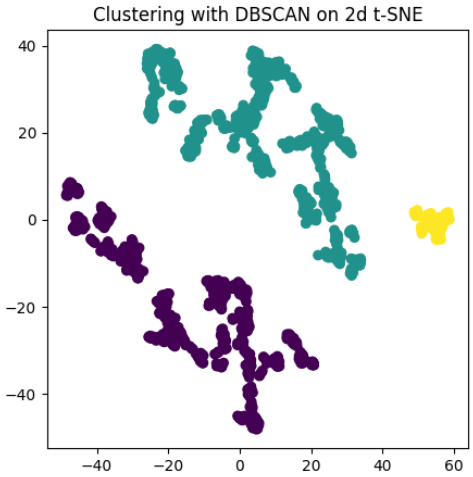
(b) perplexity=30

3.3 k-Means e DBScan

Sugli spazi ora ottenuti tramite la riduzione delle dimensionalità andiamo ad eseguire operazioni di clustering utilizzando sia K-Means che DBScan. I risultati con K-means non sono soddisfacenti in nessuno degli spazi ridotti (nemmeno cercando di aggiustare gli iper parametri), principalmente per il fatto che i clustr reali non sono sferici. Per quando riguarda DBScan otteniamo invece ottimi risultati per gli spazi ridotti da t-SNE mentre troviamo difficolta per i dati ridotti da PCA. Il motivo principale è da individuarsi nella presenza di divisioni nette per t-SNE, cosa che manca decisamente nella PCA.



(a) eps=0.3, min samples=4



(b) eps=10, min samples=3

4 Supervised Learning

Arriviamo ora all'ultima parte operativa, quella di Supervised Learning. Ci viene chiesto di eseguire predizioni tramite l'uso dei metodi studiati a lezione, i quali comprendono Regressione Logistica, Decision Tree, Naive Bayse e k-NN. I risultati ottenuti sono tutti davvero molto buoni. Precisazioni per i singoli modelli vengono presentate nelle apposite sezioni.

4.1 Logistic Regression

Dalla regressione logistica otteniamo risultati davvero ottimi. Esploriamo la regolarizzazione e scopriamo che questa addirittura peggiora i risultati. Riportiamo i risultati delle varie regressioni con relativi parametri (i risultati sono le statistiche misurate in termini di macro media).

Regolarizzazione	λ_1	λ_2	**Accuracy**	**Precision**	**Recall**	**F1-Score**
None	0	0	0.99	0.98	0.99	0.99
L1	0.1	0	0.99	0.98	0.99	0.99
None	0	0.1	0.99	0.98	0.99	0.99
L1	0.1	0.1	0.99	0.98	0.99	0.99

Table 1: Confronto delle metriche di performance delle regressioni logistiche.

4.2 Decision Tree, Naive Bayes e k-NN

I risultati del decision tree sono ottimi: già con i parametri di default otteniamo infatti errori pressoché nulli. Allo stesso modo anche k-NN. Naive Bayes è invece l'unico dei modelli che ha presentato risultati un po' meno soddisfacenti (con un'accuracy tra l'80% ed il 90%). Non abbiamo modo di sistemare gli iper parametri del modello in quanto non siamo a conoscenza di tecniche efficaci (non abbiamo visto metodi a lezione per fare hyper tuning nel naive bayes).

Modello	**Accuracy**	**Precision**	**Recall**	**F1-Score**
Decision Tree Classifier	0.98	0.98	0.98	0.98
Naive Bayes	0.85	0.86	0.85	0.85
k-NN	1	1	1	1

Table 2: Confronto delle metriche di performance tra vari metodi.

5 Considerazioni

5.1 Iper parametri

In questa challenge ci siamo ritrovati in uno scenario molto fortunato: i dati infatti riportavano ottimi risultati senza alcun tipo di fase di hyper tuning sia nell'unsupervised learning che nel supervised learning. Questo è un aspetto non banale: in molti casi prima di riuscire ad ottenere risultati soddisfacenti (e potremmo considerarci soddisfatti già con l'80% di accuracy alle volte) dobbiamo eseguire molteplici passi di validazione per sistemare gli iper parametri.

5.2 Risultati

Il modello che meglio ha performato è stato sicuramente il k-NN. Per ogni modello è stato plottato un grafico che mostrava gli errori commessi sulla proiezione 2dimensionale generata da PCA ed è stato interessante notare come le regressioni logistiche abbiano tutte sbagliato a classificare lo stesso identico punto. Bisogna poi evidenziare un collegamento molto forte tra la parte di supervised e di unsupervised learning: i risultati ottenuti infatti già ce li aspettavamo dal momento in cui abbiamo visto con i nostri occhi la divisione delle classi che si riusciva ad ottenere (approssimativamente bene) con la PCA 3dimensionale. Questo dettaglio infatti è un naturale indicatore del fatto che, con grande probabilità, nello spazio originale dei dati, esiste un iper piano separatore delle classi. Ed è questo il motivo per cui le regressioni logistiche funzionano tanto bene (poiché in realtà una regressione logistica trova un iperpiano separatore). E di conseguenza se una regressione logistica funziona bene ci aspettiamo risultati buoni anche dagli altri metodi (e ne abbiamo avuto conferma).