ADVANCED MACHINE LEARNING

REPORT ASSIGNEMENT #1

Giugliano Mirko

matricola: 800226

INTRODUZIONE

Il primo assignement per il corso di "Advanced Machine Learning" consiste nella previsione delle inadempienze di pagamento, sviluppando una rete neurale, a partire da un dataset contenente informazioni sulle suddette inadempienze di pagamento a Taiwan da aprile 2005 a settembre 2005. Le altre informazioni riguardano fattori demografici, dati di credito, cronologia dei pagamenti ed estratti conto dei clienti di carte di credito.

ANALISI ESPLORATIVA e PREPROCESSING

Durante l'esplorazione del dataset si nota subito che la dimensione è di 27000 righe e di 24 colonne, di cui, oltre alla variabile da prevedere, 3 categoriche. Quindi si è proceduto con la binarizzazione delle variabili categoriche tramite one hot encoding. In esguito si sono standardizzate tutte le altre variabili numeriche, fatta eccezione per la variabile da predire. Continuando l'analisi esplorativa si comprende che le classi da predire sono fortemente sbilanciate, così si è deciso di procedere splittando train set e test set, per poi andare a bilanciare solamente il train set, eliminando delle righe dalla classe in esubero fino ad arrivare ad ottenere un numero pari a quello della classe minoritaria. Bilanciato il train set si sono creati 4 numpy array, 2 per il train set e 2 per il test set, uno contenente la variabile da predire, l'altro tutte le altre classi.

MODELLI

PRIMO MODELLO:

- 700 epoche
- 2 hidden layers in ordine da 512 e 256 neuroni.
- activation function hidden layers: ReLU
- activation function output layer: Sigmoid
- optimizer: sgd
- loss function: binary_crossentropy
- regularization: 2 dropout layers e regolarizzatori L1 e L2

Il numero di epoche è stato scelto in base a quando l'accuracy del train set e del test set arrivavano a convergenza, evitando overfitting e/o underfitting. Il numero di layer e di neuroni è stato scelto in seguito a svariati tentativi, cercando un buon compromesso fra complessità e prestazioni. Per l'activation function degli hidden layers, la ReLU, quella comunamente più usata, risultava la più performante mentre per l'activation function degli output layer la Sigmoid, trattandosi di una classificazione binaria, è stata una scelta obbligata. L'optimizer SGD è anche qua è una scelta abbastanza standard e per evitatare l'overfitting si sono introdotti anche dei regolarizzatori L1 e L2 e due dropout layer, anche se con valori decisamente bassi, poiché il problema non sembrava presentarsi.La loss functioN binary_crossentropy è praticamente obbligatoria trattandosi di stimare valori [0,1]

SECONDO MODELLO

- 400 epoche
- 2 hidden layers in ordine da 32 e 16 neuroni.
- activation function hidden layers: Leaky ReLU
- activation function output layer: Sigmoid
- optimizer: Adam
- loss function: binary_crossentropy
- regularization: 2 dropout layers

Il numero di epoche è stato scelto, come precedentemente, in base a quando l'accuracy del train set e del test set arrivavano a convergenza, evitando l'overfitting. Anche il numero di layer e di neuroni è stato scelto seguento il medesimo ragionamento del precedente modello. Per l'activation function degli hidden layers, invece, la Leaky ReLU, più modificabile rispetto alla comune ReLU, risultava la più performante mentre per l'activation function degli output layer anche qua si è utilizzata la Sigmoid. L'optimizer utilizzato per questo modello invece è Adam, algoritmo adattivo e molto utilizzato viste le sue prestazioni. Per evitatare l'overfitting questa volta si sono implementati solo due dropout layer, con valori decisamente bassi. Loss function sempre binary_crossentropy.

VALUTAZIONE MODELLI E CONCLUSIONI

PRIMO MODELLO:
confusion matrix e classification report

```
confusion matrix
 [[4051 1162]
 [ 581  956]]
precision
 [0.87456822 0.45136922]
recall
 [0.77709572 0.62199089]
f-score
 [0.82295582 0.52311902]
```

	precision	recall	f1-score	support
0	0.87	0.78	0.82	5213
1	0.45	0.62	0.52	1537
accuracy			0.74	6750
macro avg	0.66	0.70	0.67	6750
weighted avg	0.78	0.74	0.75	6750

SECONDO MODELLO:

confusion matrix e classification report

```
confusion matrix
 [[3972 1241]
 [ 558  979]]
precision
 [0.87682119 0.44099099]
recall
 [0.7619413  0.63695511]
f-score
 [0.81535461 0.5211605 ]
```

	precision	recall	f1-score	support
0	0.88	0.77	0.82	5213
1	0.45	0.63	0.52	1537
accuracy			0.74	6750
macro avg	0.66	0.70	0.67	6750
weighted avg	0.78	0.74	0.75	6750

I risultati sono molto simili ed in entrambi i casi non eccellenti, questo potrebbe essere proprio per il dataset, difficilmente esplicabile o che richiede reti di complessità maggiori. È da aggiungere anche che, soprattutto dopo il bilanciamento, nel train set rimangono meno di 9000 righe, che non sono moltissime per una rete neurale. Si può concludere tuttavia che, a parità di prestazioni, il secondo è preferibile, poiché richiede meno una minor complessità in termini di neuroni e di epoche.