

Assignment 3 Ferrario Gabriele 817518

L'architettura utilizzata è sequenziale ed è composta da 6 layers:

1. **2D convolutional layer:** ha come input immagini di dimensione $28 \times 28 \times 1$, è composto da 8 filtri di dimensione $5 \times 5 \times 1$, fornisce un output di dimensione $24 \times 24 \times 8$, come funzione di attivazione usa la Relu e in totale ha 208 parametri;
2. **AveragePooling2D layer:** ha input di dimensione $24 \times 24 \times 8$, `pool_size=(2,2)`, fornisce un output di dimensione $12 \times 12 \times 8$ e in totale ha 0 parametri;
3. **2D convolutional layer:** ha input di dimensione $12 \times 12 \times 8$, è composto da 16 filtri di dimensione $5 \times 5 \times 8$, fornisce un output di dimensione $8 \times 8 \times 16$, come funzione di attivazione usa la Relu e in totale ha 3216 parametri;
4. **AveragePooling2D layer:** ha input di dimensione $8 \times 8 \times 16$, `pool_size=(2,2)`, fornisce un output di dimensione $4 \times 4 \times 16$ e in totale ha 0 parametri;
5. **Flatten layer:** ha input di dimensione $4 \times 4 \times 16$, restituisce un output di dimensione 1×256 e in totale ha 0 parametri;
6. **Dense layer:** ha input di dimensione 1×256 , restituisce un output di dimensione 1×10 , in totale ha 2570 parametri e come funzione di attivazione usa la softmax poiché si tratta di un task di classificazione multi-classe;

Quindi in totale il modello è composto da 5.994 parametri.

Ho usato questa configurazione della rete poiché nelle prove che ho fatto è stata quella che si è comportata meglio ottenendo le performance migliori.

Come optimizer ho utilizzato adam mentre, come loss function la categorical_crossentropy (funzione di perdita usata nei tasks di classificazione multi-classe) e come metrica ho usato l'accuracy.

Per l'allenamento e la valutazione del modello ho deciso di suddividere i dati a disposizione per il training in 90% training e 10% validation.

Il modello è stato allenato tramite 15 epoche e con la dimensione dei batch pari a 16. Con il passare delle epoche il valore dell'accuracy si alza fino a raggiungere cifre vicine a 0.995 nel training e di 0.99 nel validation. Infine, valutando il modello sul test set si ottiene un valore di accuracy pari a 0.99

e di loss pari a 0.03.

Di seguito vengono riportati i plot dell'accuracy (figura 1) e della loss (figura 2) ottenuti utilizzando il training e il validation.

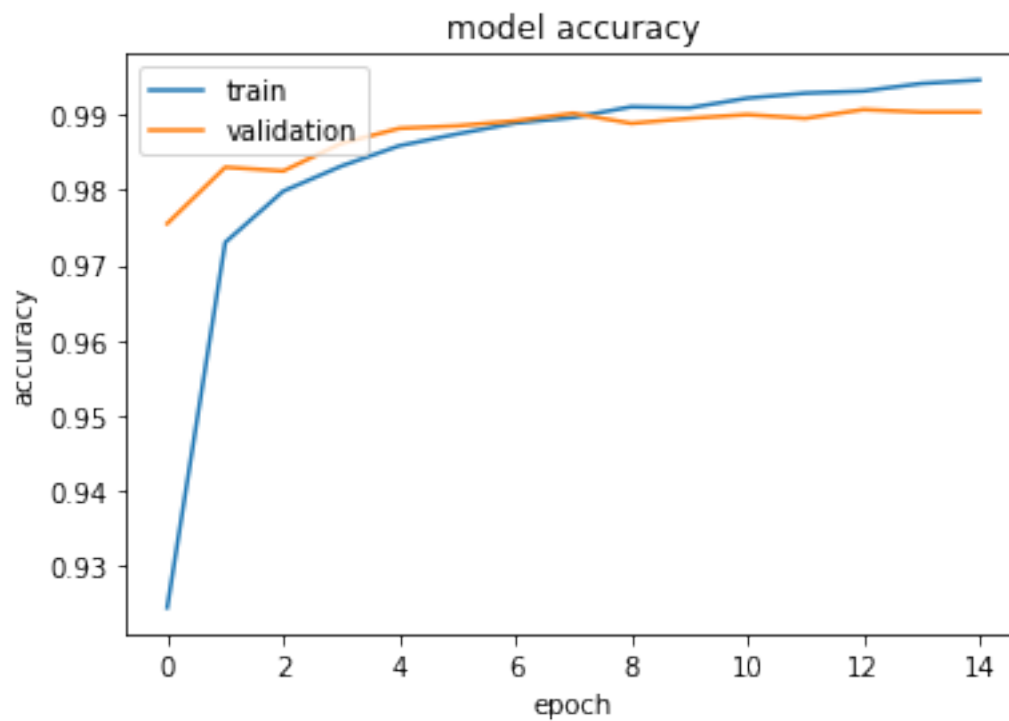


Figura 1: Plot accuracy

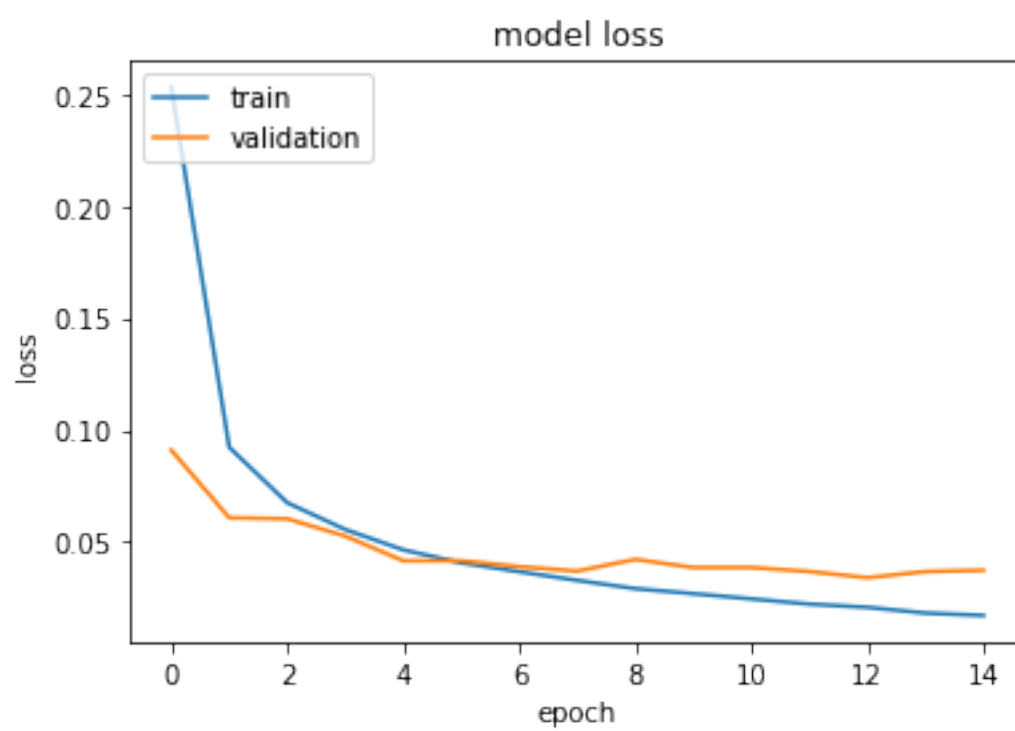


Figura 2: Plot loss