Sperimentazione

Sperimentazione

Premessa

Abbiamo visto:

1. Come definire il problema.
2. Come utilizzare i dati.
3. Come creare un modello di rete.
4. Come addestrare un modello.
5. Come valutare le performance ottenute con quel modello.

Come si può essere certi che il modello ottenuto sia il migliore?

Sperimentazione

Premessa

La sperimentazione ha un obbiettivo principale:

*Permettere di cercare e selezionare il **modello migliore** fra una serie di modelli ottenuti.*

Sperimentare, nel Deep Learning, significa:

- Creare più modelli.
- Fare più addestramenti.
- Modificare i dati.
- Modificare parametri.
- Modificare iper-parametri.
- …

Sperimentazione

Premessa

Come accade in un laboratorio:

- Ogni combinazione che si sceglie di testare andrà a costituire un **esperimento**.
- L'esperimento partirà con dei parametri iniziali.
- Sarà valutato tramite apposite metriche: *per l'addestramento, la validazione e il test set…*
- Delle metriche si analizzeranno punteggi e trend ottenuti.

Al termine di questo processo iterativo sarà possibile dare uno sguardo a tutti i risultati ottenuti e scegliere il risultato **migliore**.

Il modello più accurato…Il più leggero da eseguire…il più rapido ad addestrarsi…

Sperimentazione

Cosa si può sperimentare

Molteplici sono i **parametri** su cui agire, di seguito alcuni:

- Dimensione del dataset.
- Data augmentation.
- Pre-processamento dei dati.

Molteplici sono gli **iper-parametri** su cui agire:

- Batch size.
- Learning rate.
- Dropout.
- Numero di layer e/o di neuroni per layer.

Sperimentazione

Parametri

Molteplici sono i **parametri** su cui agire, di seguito alcuni:

- **Dimensione del dataset:**
  Verificare il numero minimo di dati sufficienti a raggiungere determinate performance è utile nel caso in cui il dataset non sia molto ampio, e permette di risparmiare tempo di addestramento.
- **Data augmentation:**
  Provare differenti tipi di data augmentation permette di aggiungere dati diversi al dataset, migliorare le performance, aumentare la variabilità e la rappresentatività dei campioni…
- **Pre-processamento dei dati:**
  Il pre-processamento dei dati è una parte cruciale per la riuscita del modello. Valutare di utilizzare i dati non trattati, standardizzarli, normalizzarli…Decidere se usarli in ordine o applicarne uno shuffle.

Sperimentazione

Preprocessamento e seed

Lo shuffle dei dati permette di creare dataset diversi con cui addestrare la rete.

Questo è utile per verificare che il modello ottenuto sia robusto e che le buone performance non dipendano da una combinazione fortunata di dati.

Esistono molti elementi/algoritmi di PyTorch che basano la loro riuscita sull'utilizzo di generazione randomica di dati. Essendo però una generazione pseudo-casuale, è sempre possibile, imponendo un ***seed*** di generazione, essere ripetibili e costanti nella generazione dei valori.

Il seed può essere settato sia in numpy che in PyTorch:

numpy.random.seed(int) *torch.manual_seed(int)*

Sperimentazione

Iperparametri

Molteplici sono gli **iper-parametri** su cui agire:

- **Batch-size** e **Learning-rate**:
  La cui influenza è già stata descritta.
- **Dropout**:
  Influenza la capacità di apprendimento della rete scegliendo, sulla base di scelte probabiliste, se utilizzare o meno dei neuroni, se percorrere o meno dei percorsi della rete…
- **Numero di layer:**
  La profondità della rete determina quanto la rete sarà in grado di apprendere. Tuttavia una rete troppo profonda potrebbe portare overfitting e allungare notevolmente i tempi di addestramento, come quelli di inferenza.
- **Numero di neuroni**:
  Il numero di neuroni di ogni strato, deve essere abbastanza grande in modo che la rete possa apprendere ma abbastanza piccolo da non creare overfitting e non appesantire inutilmente il modello.

Proviamo?