Batch Normalization

- Introdotta da S. Ioffe and C. Szegedy nel 2015
- Consiste nella **normalizzazione** (fatta per mini-batch durante la fase di train) delle attivazioni di livelli di rete neurale (media zero e varianza uno)
- Basata sull'idea che così come la normalizzazione dell'input rende più rapido l'apprendimento di una rete neurale, perché dovremmo avere problemi normalizzando anche gli input dei livelli più deep?
- Rende l'apprendimento di un layer *indipendente* dagli altri
- Se riusciamo a normalizzare in qualche modo gli output di un livello precedente, la discesa del gradiente **convergerà meglio** in fase di training

Batch Normalization

- La BN è semplicemente un altro livello inserito fra due livelli hidden adiacenti
- Il suo lavoro è prendere l'output del livello [l], normalizzarlo e passarlo al livello [l+1]
- In alcune implementazioni non viene preso l'output $a^{[l]}$, ma $z^{[l]}$

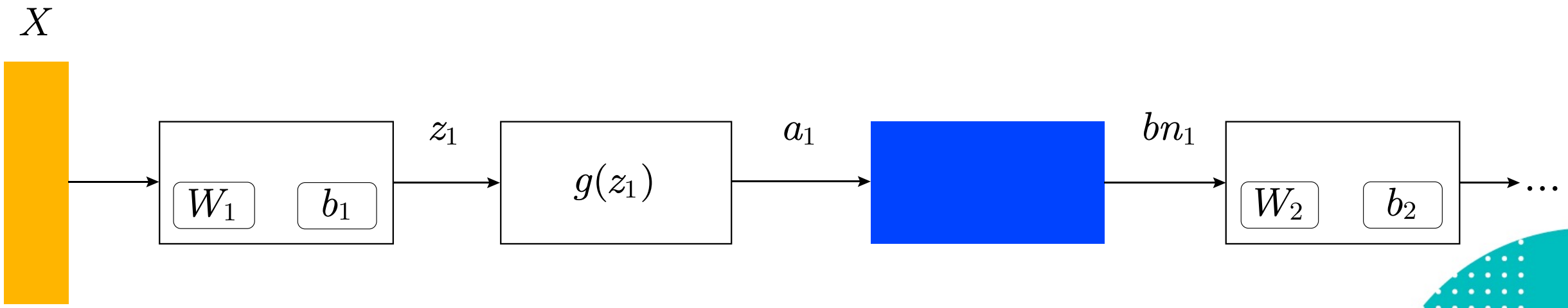
Ricordiamo che:

$$z^{[i]} = W^{[i]} * a^{[i-1]} + b^{[i]}$$

L'output di un livello in una rete neurale è dato dall'applicazione della funzione di attivazione a z

$$a^{[i]} = g(z^{[i]})$$

Batch Normalization



Batch Normalization

- Calcolo **media** e **varianza** del mini-batch
- **Normalizzazione** del batch
- **Scale and Shift** (apprendimento)

Fase di training

- La batch normalization fa cose diverse in fase di training e in fase di test
- Il livello BN calcola Media e Deviazione Standard dei valori di attivazioni sul batch:

$$\mu = \frac{1}{n}\sum_i A^i \qquad \sigma = \frac{1}{n}\sum_i (A^i - \mu)$$

- Normalizza il vettore

$$A^{(i)}_{norm} = \frac{A^i - \mu}{\sqrt{\sigma^2 - \epsilon}}$$

N.B. ε è una costante che serve unicamente ad evitare divisioni per 0

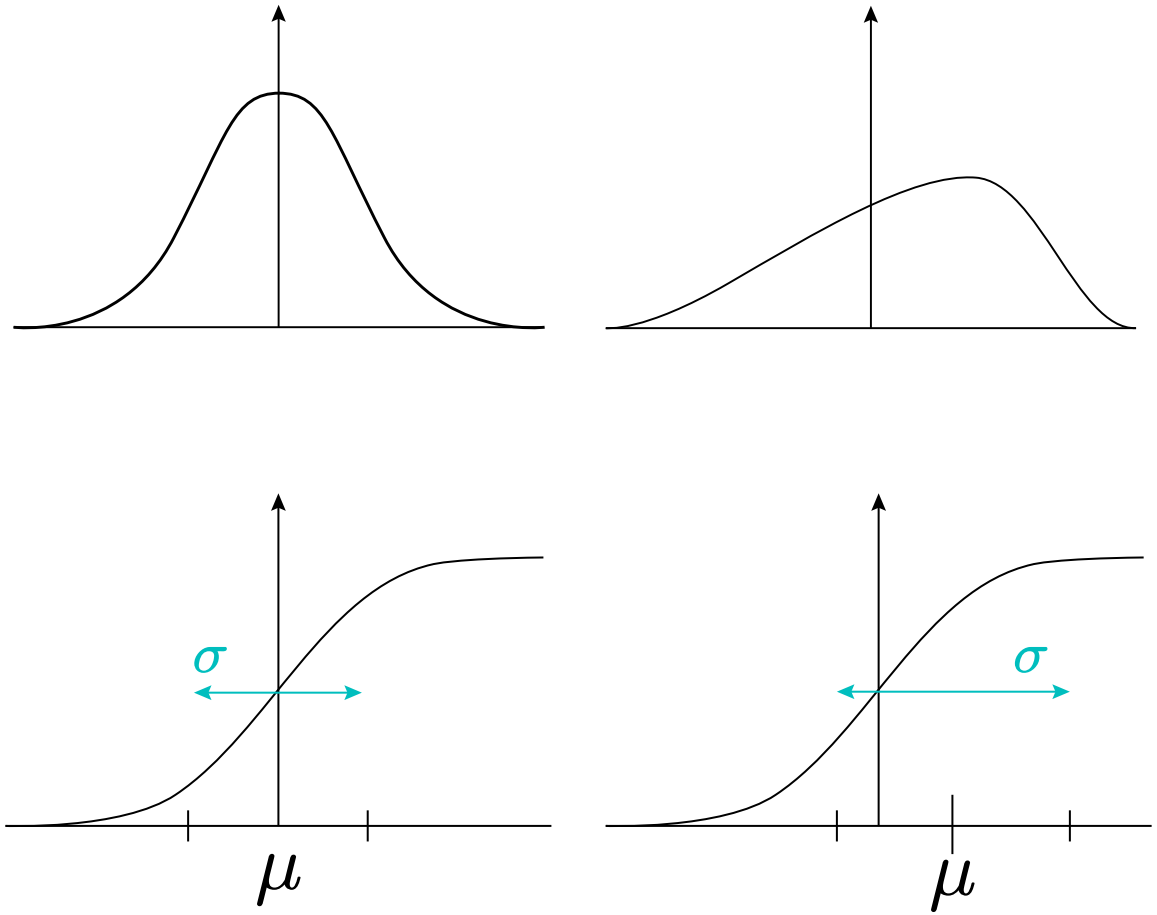
Fase di training

- Calcola il valore del layer di output applicando una trasformazione lineare con due parametri addestrabili:

$$\hat{A} = \gamma * A_{norm}^{(i)} + \beta$$

- Questa operazione è detta **scale & shift** e ci permette di modificare la distribuzione in uscita dal livello agendo su media (beta) e deviazione standard (gamma)

Scale & Shift



- Tramite l'operazione di **shift** (apprendimento del valore di beta) possiamo shiftare i valori in uscita su un valore medio **diverso** da quello calcolato
- Tramite l'operazione di **scale** (apprendimento del valore di gamma) possiamo scalarlo ad un valore di varianza diverso
- Gamma e Beta non sono iperparametri, sono parametri **addestrabili**

Fase di evaluation

- Durante la fase di training vengono anche calcolate le **media mobile** della media e della variazione standard:

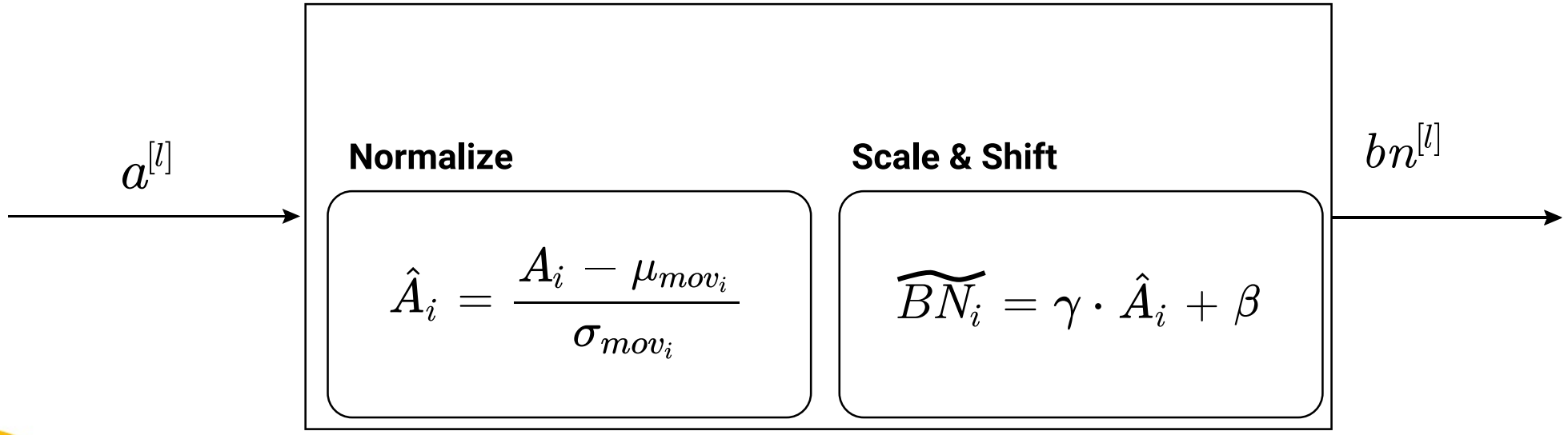
$$\mu_{mov_i} = \alpha * \mu_{mov_i} + (1 - \alpha)\mu_i$$

$$\sigma_{mov_i} = \alpha * \sigma_{mov_i} + (1 - \alpha)\sigma_i$$

- Questi valori verranno usati nella fase di **evaluation**:
  - Fase di test
  - Fase di deploy

Fase di test

- In fase di **deploy** non avremo dei batch ma dei singoli sample su cui effettuare le predizioni, non dei batch
- Vengono usate come media e dev. standard quelle mobili calcolate in fase di addestramento
- Le medie mobili sono più efficienti da calcolare rispetto alle medie su tutta la popolazione



Batch Normalization

- Intuition: mitigare fenomeni tipici dell’addestramento delle reti neurali come
  - Vanishing ed Exploding Gradient
  - Internal Covariate Shift
- Effetti:
  - Accelerazione dell’addestramento
  - Regolarizzazione