Normalization

Zenglin Shi

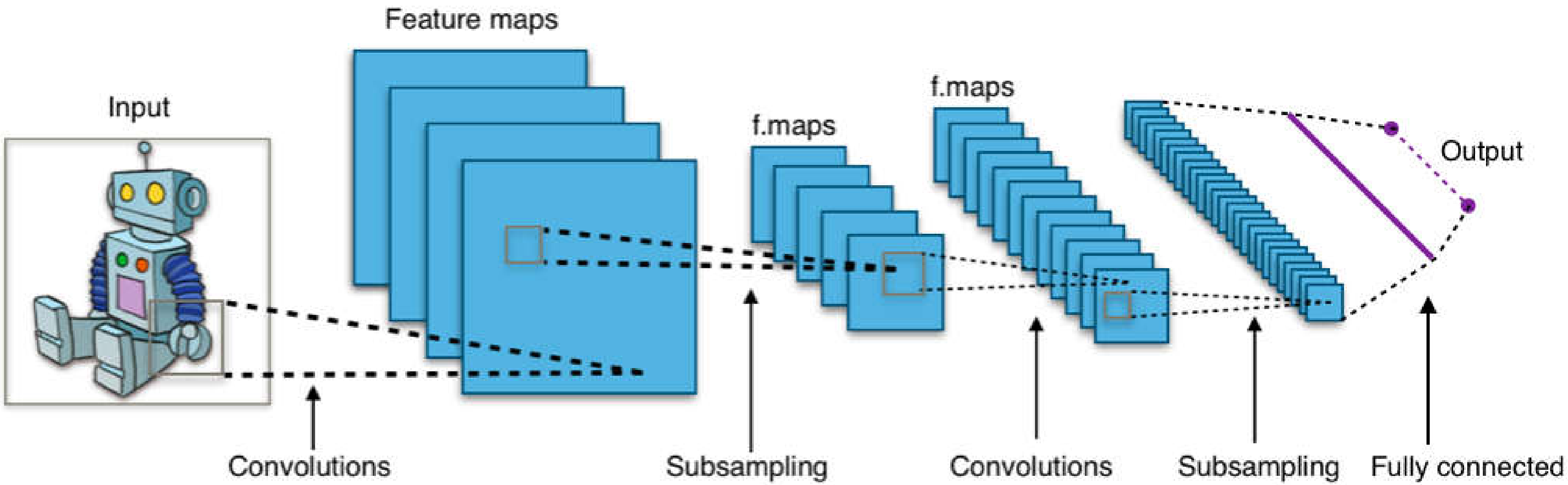
Data Normalization

- Why Data Normalization?
- **Increased consistency.** Information is stored in one place and one place only, reducing the possibility of inconsistent data.
- **Easier object-to-data mapping**. Highly-normalized data schemas in general are closer conceptually to object-oriented schemas because the object-oriented goals of promoting high cohesion and loose coupling between classes results in similar solutions (at least from a data point of view).

Data Normalization

- Two methods are very common :
- **Min-Max scaling:** Subtract the minimum value and divide by the range (i.e maximum value - minimum value) of each column. Each new column has 0 as its minimum value and 1 as its maximum.
- **Standardization scaling**: Subtract the mean[1] and divide by the standard deviation[2] of each column. Each new column has mean 0 and standard deviation of 1.

ConvNet



Feature maps
Input
f.maps
f.maps
Output
Convolutions
Subsampling
Convolutions
Subsampling
Fully connected

Batch Normalization

Input: Values of x over a mini-batch: $\mathcal{B} = \{x_{1...m}\}$;
Parameters to be learned: γ, β

Output: $\{y_i = \text{BN}_{\gamma,\beta}(x_i)\}$

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i \qquad \text{// mini-batch mean}$$

$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu_{\mathcal{B}})^2 \qquad \text{// mini-batch variance}$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} \qquad \text{// normalize}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \text{BN}_{\gamma,\beta}(x_i) \qquad \text{// scale and shift}$$

$$\mu := \mu + \alpha(\mu_{\mathcal{B}} - \mu) \qquad \text{// Update moving averages}$$

$$\sigma := \sigma + \alpha(\sigma_{\mathcal{B}} - \sigma)$$

Inference: $$y \leftarrow \gamma \cdot \frac{x - \mu}{\sigma} + \beta$$

Batch ReNormalization

$$\frac{x_i - \mu}{\sigma} = \frac{x_i - \mu_\mathcal{B}}{\sigma_\mathcal{B}} \cdot r + d, \quad \text{where } r = \frac{\sigma_\mathcal{B}}{\sigma}, \quad d = \frac{\mu_\mathcal{B} - \mu}{\sigma}$$

Input: Values of x over a training mini-batch $\mathcal{B} = \{x_{1\ldots m}\}$; parameters γ, β; current moving mean μ and standard deviation σ; moving average update rate α; maximum allowed correction $r_{\max}$, $d_{\max}$.

Output: $\{y_i = \mathrm{BatchRenorm}(x_i)\}$; updated μ, σ.

$$\mu_\mathcal{B} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i$$

$$\sigma_\mathcal{B} \leftarrow \sqrt{\epsilon + \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_\mathcal{B})^2}$$

$$r \leftarrow \texttt{stop\_gradient}\left(\mathrm{clip}_{[1/r_{\max}, r_{\max}]}\left(\frac{\sigma_\mathcal{B}}{\sigma}\right)\right)$$

$$d \leftarrow \texttt{stop\_gradient}\left(\mathrm{clip}_{[-d_{\max}, d_{\max}]}\left(\frac{\mu_\mathcal{B} - \mu}{\sigma}\right)\right)$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_\mathcal{B}}{\sigma_\mathcal{B}} \cdot r + d$$

$$y_i \leftarrow \gamma\,\widehat{x}_i + \beta$$

$$\mu := \mu + \alpha(\mu_\mathcal{B} - \mu) \qquad \text{// Update moving averages}$$

$$\sigma := \sigma + \alpha(\sigma_\mathcal{B} - \sigma)$$

Inference: $y \leftarrow \gamma \cdot \dfrac{x - \mu}{\sigma} + \beta$

Group Normalization

```
def GroupNorm(x, gamma, beta, G, eps=1e-5):
    # x: input features with shape [N,C,H,W]
    # gamma, beta: scale and offset, with shape [1,C,1,1]
    # G: number of groups for GN

    N, C, H, W = x.shape
    x = tf.reshape(x, [N, G, C // G, H, W])

    mean, var = tf.nn.moments(x, [2, 3, 4], keep_dims=True)
    x = (x - mean) / tf.sqrt(var + eps)

    x = tf.reshape(x, [N, C, H, W])

    return x * gamma + beta
```

Instance Normalization

$$y_{tijk} = \frac{x_{tijk} - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}}, \quad \mu_i = \frac{1}{HWT} \sum_{t=1}^{T} \sum_{l=1}^{W} \sum_{m=1}^{H} x_{tilm}, \quad \sigma_i^2 = \frac{1}{HWT} \sum_{t=1}^{T} \sum_{l=1}^{W} \sum_{m=1}^{H} (x_{tilm} - mu_i)^2. \tag{2}$$

$$y_{tijk} = \frac{x_{tijk} - \mu_{ti}}{\sqrt{\sigma_{ti}^2 + \epsilon}}, \quad \mu_{ti} = \frac{1}{HW} \sum_{l=1}^{W} \sum_{m=1}^{H} x_{tilm}, \quad \sigma_{ti}^2 = \frac{1}{HW} \sum_{l=1}^{W} \sum_{m=1}^{H} (x_{tilm} - mu_{ti})^2.$$

Adaptive Normalization

$$\Psi^s(x) = \lambda_s x + \mu_s BN(x),$$