05/10/23

Deep learning

- Recap: Batch normalization
- Inplant
- ResNet
- $y_k^{(i)} = \gamma_k \cdot \frac{x_k^{(i)} - \mu_k}{\sqrt{\sigma_k^2 + \varepsilon}} + \beta_k$

  i → sample index; k → dim index

$$B = \begin{bmatrix} | & | & & | \\ \underline{x}^{(1)} & \underline{x}^{(2)} & \cdots & \underline{x}^{(m)} \\ | & | & & | \end{bmatrix} = \begin{bmatrix} x_1^{(1)} & \cdots & x_1^{(m)} \\ x_2^{(1)} & & \\ \vdots & & \\ x_d^{(1)} & & x_d^{(m)} \end{bmatrix}$$

(μ_1, σ_1^2 over row 1)

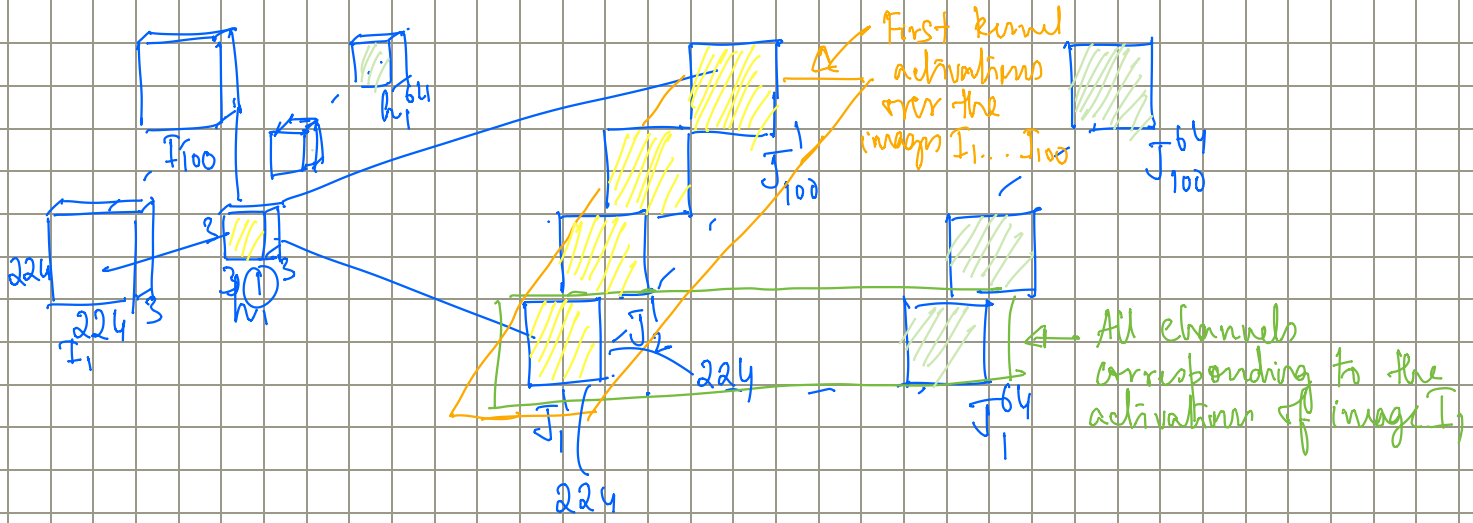
$$B_{BN} = \begin{bmatrix} y_1^{(1)} = \gamma_1 \frac{(x_1^{(1)} - \mu_1)}{\sqrt{\sigma_1^2 + \varepsilon}} + \beta_1 \\ \vdots \\ y_d^{(m)} = \gamma_d \hat{x}_d^{(m)} + \beta_d \end{bmatrix}$$

- BN in conv layer:

  I is of size $224 \times 224 \times 3$

  First conv layer: conv 3 - 64

  The output of the first conv layer: $224 \times 224 \times 64$ (assuming "same" size as input)



- BN applied channel wise.

$$\mu_B^k = \frac{1}{m \times W \times H} \sum_w \sum_h \sum_m I_m^k(w,h)$$

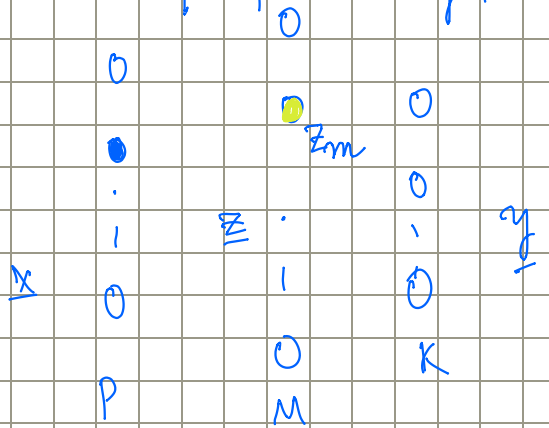
Similar definition for σ_k^2.

- Rest as before.

- Nrtk with BN $\quad \theta_{BN} = \{\theta \cup \gamma_k, \alpha_k\}$ $\quad k$: no of nodes where BN is applied.

———— x ————

- A regularization technique for training DL models: Dropout



Vanilla: $z_m = \sigma\left(\langle \underline{x}, \underline{\alpha}_m \rangle\right)$

Dropout: $z_m = \sigma\left(\langle \underline{r} \odot \underline{x}, \underline{\alpha}_m \rangle\right)$

$r = [r_1, r_2, \ldots, r_p]^T$

$r_i \sim \text{Bern}(p)$

$\Downarrow$

$P(r_i = 1) = p$

$P(r_i = 0) = 1 - p$

Recommendation: $p = 0.8$ for input layer

$p = 0.5$ for hidden layer

Aside: ($p = 1$ for output layer) We do not drop output nodes.