

AMAL (TP 8)

Regularisation \longrightarrow

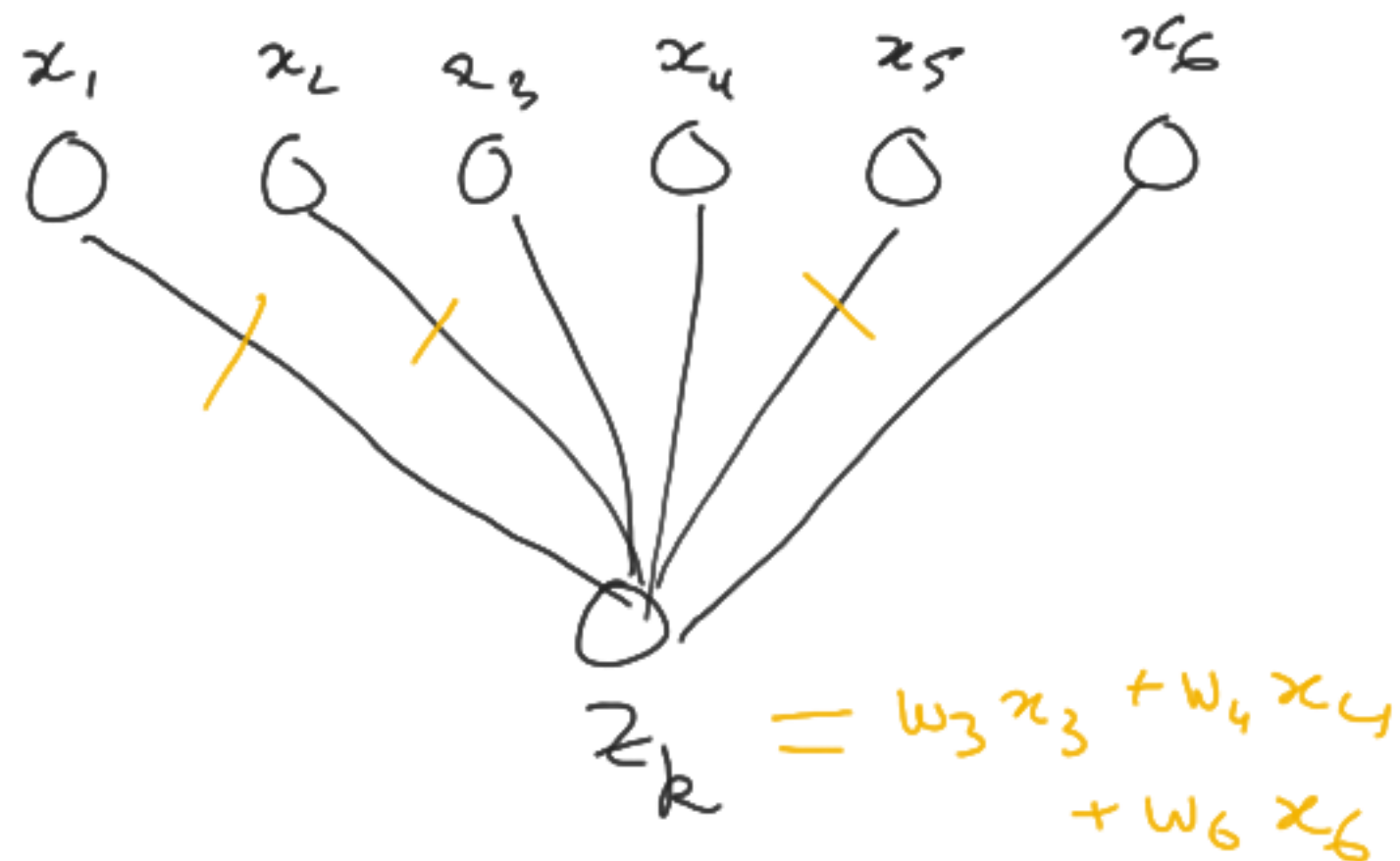
$L1/L2$ L_c
(à priori Gaussien /
Laplace sur les
 L_1
paramètres)

Dropout
"ensemble" light

- ① Y_i les prévisions
d'un modèle i
(recevant X_i)
- ② Y_i sont iid
écart type σ , $\mu_i = y_i$

$$E(\bar{Y}) = y_i$$

$$V(\bar{Y}) = \frac{V(Y_i)}{n}$$



Dropout $p = 0.5$

⚠ En inférence,

$$z_k = P\left(\sum_j w_j x_j\right)$$

En pratique, • model.train()
en app.

• model.eval()
en inférence

→ Batch Norm / Layer Norm → 1 chose mais au niveau d'1 individu

↳ idée entrées soient centrées / réduites

		caractéristiques
individus		$z_k^{(1)}$
		\vdots
		$z_k^{(b)}$
		Batch

$$z_k \rightarrow \mathbb{E}(z_k) = 0 \quad V(z_k) = 1$$

en mode app., on utilise le batch

$$\bar{z}_k = \frac{1}{b} \sum_i z_k^{(i)}$$

σ_k (estimé sur le batch)

en mode inférence, on utilise des estimations "glissantes" de l'esp. et de l'écart type

→ Regarder l'effet des différentes techniques de régularisation

→ (Q1) Surveiller les valeurs

- des paramètres
- des gradients
- entropie sortie
- coût (train / test)



Utilisez seulement 5% du jeu de train (au moins)

↳ TensorBoard

add_histogram (paramètres, gradients, etc)
(ne l'utilisez pas à chaque itération)

add_hparams pour les hyperparamètres

