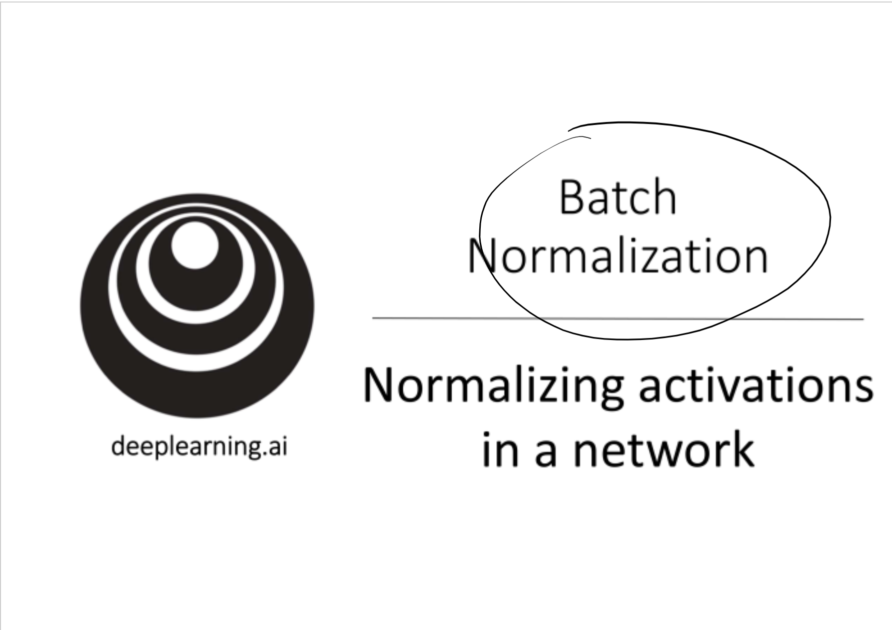
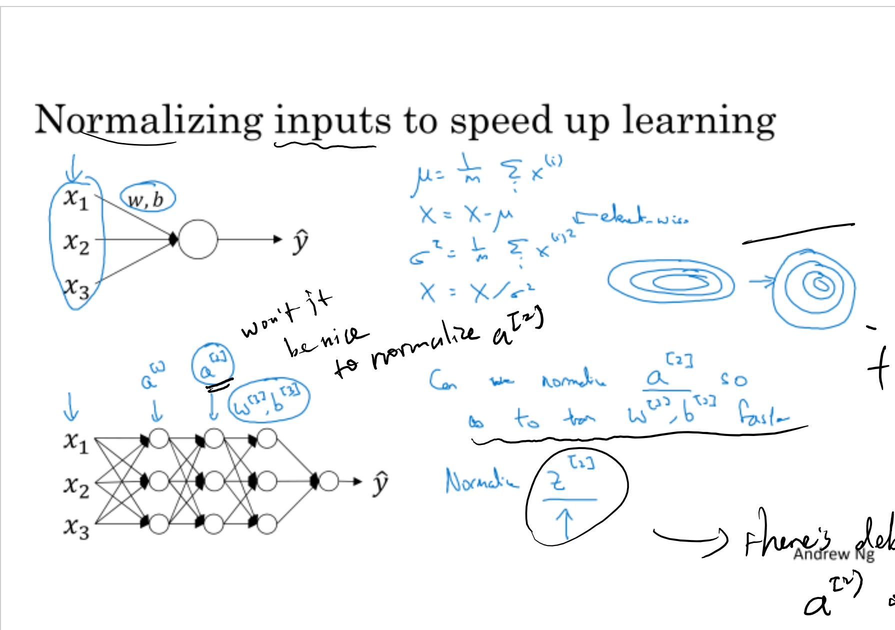
Batch normalization

Monday, June 4, 2018 9:32 PM

batchnorm1



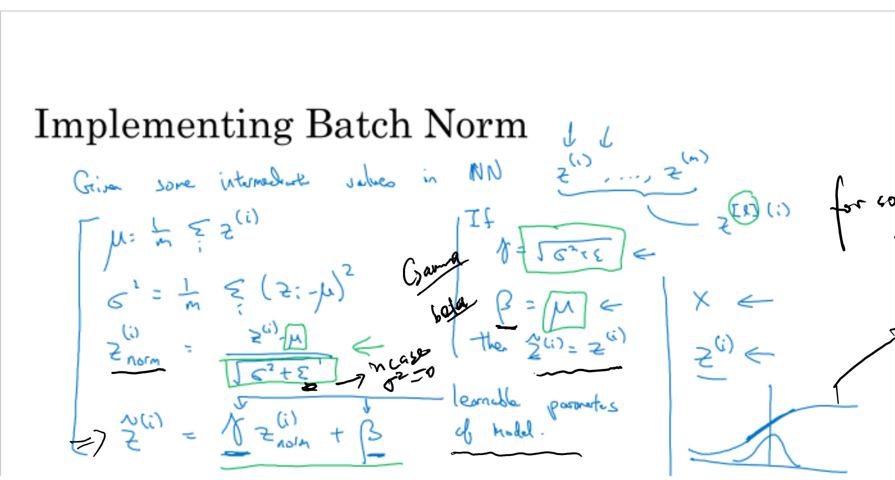


from elongated to ..

How about a deeper model

This is what batch norm like

→ there's debate about whether you should normalize $a^{[2]}$ or $z^{[2]}$ Andrew Ng prefers $z^{[2]}$



for some hidden layer
[l] omitted in this slide.

→ You don't always want to want your hidden layer value forced to mean 0.

how to rescale?

e.g. if you have ... activation

$\hookrightarrow \tilde{z}^{(i)} = \gamma z^{(i)}_{norm} + \beta$ learnable parameters of model.

Use $\tilde{z}^{[l](i)}$ instead of $z^{[l](i)}$.

Andrew Ng

how to rescale?

forced ...

eg. if you have sigmoid activation

adjust $\gamma . \beta$

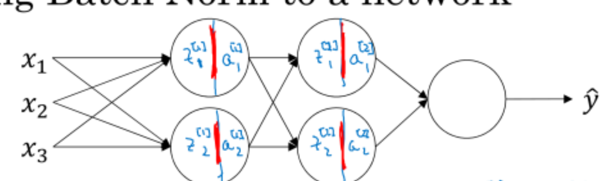
batchnorm2

Batch Normalization

Fitting Batch Norm into a neural network

deeplearning.ai

Adding Batch Norm to a network



$X \xrightarrow{w^{[1]}, b^{[1]}} z^{[1]} \xrightarrow[\text{Batch Norm (BN)}]{\beta^{[1]}, \gamma^{[1]}} \tilde{z}^{[1]} \rightarrow a^{[1]} = g^{[1]}(\tilde{z}^{[1]}) \xrightarrow{w^{[2]}, b^{[2]}} z^{[2]} \xrightarrow[\text{BN}]{\beta^{[2]}, \gamma^{[2]}} \tilde{z}^{[2]} \rightarrow a^{[2]} \rightarrow \dots$

Parameters: $w^{[1]}, b^{[1]}, w^{[2]}, b^{[2]}, \dots, w^{[L]}, b^{[L]}$ $\quad d\beta^{[l]} \quad \beta^{[l]} = \beta^{[l]} - \alpha \, d\beta^{[l]}$ ← these can be updated ..

$\rightarrow \beta^{[1]}, \gamma^{[1]}, \beta^{[2]}, \gamma^{[2]}, \dots, \beta^{[L]}, \gamma^{[L]}$ $\quad$ tf.nn.batch-normalization ← tensorflow

$\rightarrow \beta$

$\alpha \rightarrow$ distinguish from the hyperparam of momentum etc.

Andrew Ng

Working with mini-batches

$X^{\{1\}} \xrightarrow{W^{[1]}, b^{[1]}} z^{[1]} \xrightarrow[BN]{\beta^{[1]}, \gamma^{[1]}} \tilde{z}^{[1]} \rightarrow g^{[1]}(\tilde{z}^{[1]}) = a^{[1]} \xrightarrow{W^{[2]}, b^{[2]}} z^{[2]} \rightarrow \dots$

$\boxed{X^{\{2\}}} \rightarrow z^{[1]} \xrightarrow[BN]{\beta^{[1]}, \gamma^{[1]}} \tilde{z}^{[1]} \rightarrow \dots$

Just using minibatch

$X^{\{3\}} \rightarrow \dots$

Parameters: $W^{[l]}$, $\cancel{b^{[l]}}$, $\beta^{[l]}$, $\gamma^{[l]}$.

$(n^{[l]}, 1)$ $(n^{[l]}, 1)$ $(n^{[l]}, 1)$

$z^{[l]}$ $(n^{[l]}, 1)$

$\rightarrow z^{[l]} = W^{[l]} a^{[l-1]} + \boxed{\cancel{b^{[l]}}}$

$z^{[l]} = W^{[l]} a^{[l-1]}$

$z^{[l]}_{norm}$

$\rightarrow \tilde{z}^{[l]} = \gamma^{[l]} z^{[l]}_{norm} + \boxed{\beta^{[l]}}$

# of layers

第 l 层节点 y normalization

$a^{[l](1)}$ $a^{[l](2)}$ $a^{[l](3)}$

Andrew Ng

important

within batch normalization

$b^{[l]}$ not necessary

=> will be canceled out by normalization

or set $b^{[l]} = 0$

batch normal zero out the mean of $z^{[l]}$

=> so set $b^{[l]} = 0$.

Implementing gradient descent

BN = batch normalization

for $t = 1 \dots$ num MiniBatches

Compute forward prop on $X^{\{t\}}$.

In each hidden layer, use BN to replace $z^{[l]}$ with $\tilde{z}^{[l]}$.

Use backprop to compute $dW^{[l]}$, $\cancel{db^{[l]}}$, $d\beta^{[l]}$, $d\gamma^{[l]}$

Update params
$W^{[l]} := W^{[l]} - \alpha\, dW^{[l]}$
$\beta^{[l]} := \beta^{[l]} - \alpha\, d\beta^{[l]}$
$\gamma^{[l]} := \dots$

Works w/ momentum, RMSprop, Adam.

Andrew Ng

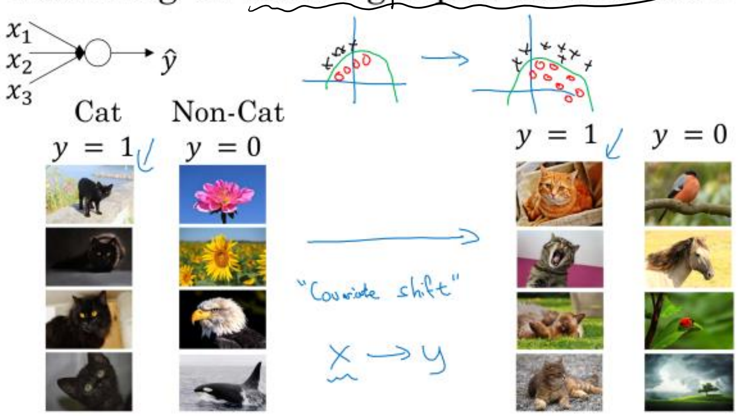
batchnorm3

deeplearning.ai

Batch Normalization

Why does Batch Norm work?

Learning on shifting input distribution



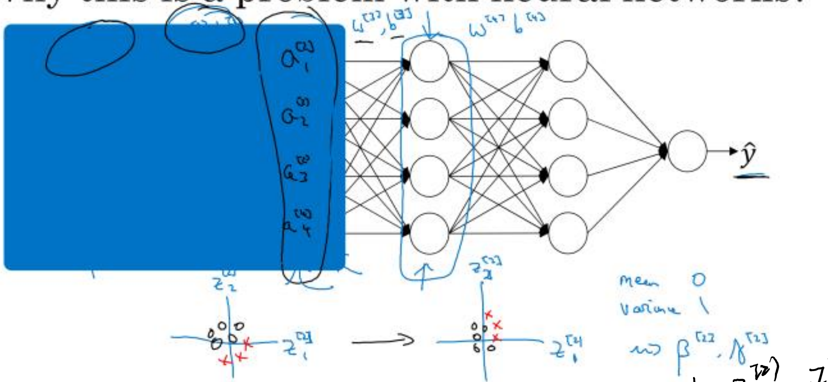
Andrew Ng

make weights robust to changes of change of weights in previous layers

model of cat detection trained on black cat picture

*-> test on colored cats. this is an example of a changed data distribution

Why this is a problem with neural networks?



mean & variance of $z_1^{[2]}$ $z_1^{[2]}$ won't change.

Andrew Ng

Batch norm reduces the amount that the hidden-layer values change.

Batch Norm 的意义

听懂一部分吧.

Stabilize reduce the effect of input value on later layers

⇒ speed up learning

△ So does it help reusing pre-trained model

Batch Norm as regularization

- Each mini-batch is scaled by the mean/variance computed on just that mini-batch.
- This adds some noise to the values $z^{[l]}$ within that minibatch. So similar to dropout, it adds some noise to each hidden layer's activations.
- This has a slight regularization effect.

X $X^{\{t\}}$ $\tilde{z}^{[l]}$ 64, 128 $z^{[l]}$ μ, σ^2

mini-batch : 64 ⟶ 512

有点像 data augmentation 么?

Andrew Ng

作用 ② regularization.

Add noise

Next: what to do at test time?

batchnorm4

8. Which of the following statements about γ and β in Batch Norm are true?

- [x] They can be learned using Adam, Gradient descent with momentum, or RMSprop, not just with gradient descent.

Correct

- [ ] β and γ are hyperparameters of the algorithm, which we tune via random sampling.

Un-selected is correct

- [x] The optimal values are $\gamma = \sqrt{\sigma^2 + \varepsilon}$, and $\beta = \mu$.

This should not be selected

- [x] They set the mean and variance of the linear variable $z^{[l]}$ of a given layer.

Correct

- [ ] There is one global value of $\gamma \in \Re$ and one global value of $\beta \in \Re$ for each layer, and applies to all the hidden units in that layer.

Un-selected is correct

computed at mini batches

Batch Norm at test time

$$\mu = \frac{1}{m}\sum_i z^{(i)}$$

$$\sigma^2 = \frac{1}{m}\sum_i (z^{(i)} - \mu)^2$$

$$z^{(i)}_{\text{norm}} = \frac{z^{(i)} - \mu}{\sqrt{\sigma^2 + \varepsilon}}$$

estimated

$$\tilde{z}^{(i)} = \gamma z^{(i)}_{\text{norm}} + \beta$$

μ, σ^2: estimate using exponentially weighted average (across mini-batches).

$X^{\{1\}}, X^{\{2\}}, X^{\{3\}}, \dots$

$\mu^{\{1\}[l]}$, $\mu^{\{2\}[l]}$, $\mu^{\{3\}[l]}$ → μ

θ_1, θ_2, θ_3

$\sigma^{2\{1\}[l]}$, $\sigma^{2\{2\}[l]}$, ... σ^2

每层一个 μ, σ^2

$z_{norm} = \frac{z - \mu}{\sqrt{\sigma^2 + \varepsilon}}$ $\tilde{z} = \gamma z_{norm} + \beta$

Andrew Ng

estimate μ, σ^2

Use running average to estimate μ, σ^2

keep a running average

use the exponentiated average