<u>Batch- Norm</u>

Problems(Internal Covariate shift)

- Distribution of each layers input changes during training, a parameters of previous layer change.
- Resulting in slower training due to lower learning rates, careful paramater initialisation, dealing with saturating non-linearities.

Batch Norm allows use of **higher learning rates**(thus can converge faster), and be less careful on parameter initialisation.

The change in the distribution of layer's input presents a problem, as they have learn a new distribution every time.(covariate shift - when the distribution of input changes of a learning system)

It is good for distribution of xi to remain fixed, as then weights do not have to always change to compensate for change in xi. (1st Reason)

Due to batch-norm, we can use saturated non-linearities(resulting to vanishing gradient in deep network), as now due to normalising , they won't get stuck in saturation, hence training accelerates. (2nd Reason)

Covariate shift - Input distribution change
Internal covariate shift - Internal node in deep network distribution change.

Applying batch norm we can easily use large values of learning rate , without worrying as all Ws are scaled, (prevents small changes in W in translating deep into the network=>no problem of vanishing and exploding gradient)(gradient becomes independent of the scale of the parameters)(3rd Reason)

This is fact - A network training converges faster if input data is whitened(covariance matrix is an identity matrix)

Since the training of a particular example doesn't just depend on the example,but on the complete mini-batch, thus it has a regularisation effect.(4th Reason)