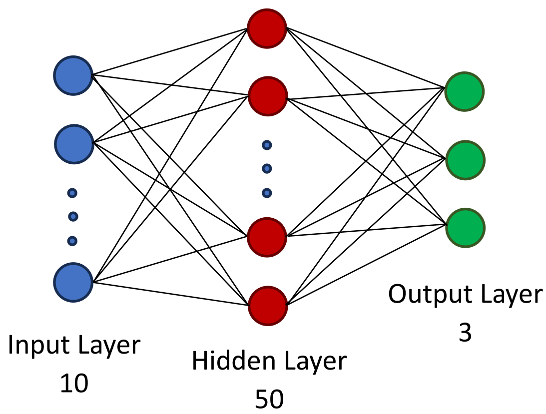
<u>EX 1 – DL basics</u>

<u>Liav Eliyau 308167675</u>

<u>Inbal Cohen 211388491</u>

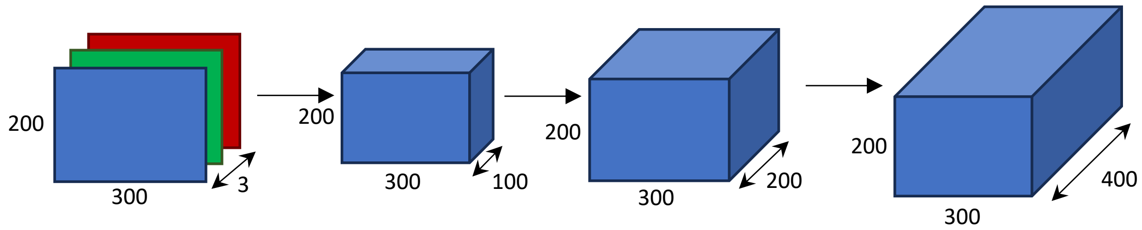
<u>Theory</u>

1.



a. The shape of the input X is $m \times 10$ – m vectors of the input size 10.
b. The shape of the weigh vector W_h is 10×50 and the shape of its bias vector b_h is 50×1.
c. The shape of W_0 is 50×3 , and b_0 is 3×1.
d. The shape of the output Y is $m \times 3$.
e. $Y = \psi(Z_0 W_0 + b_0) = \psi(\psi(W_h X + b_h)W_0 + b_0) =$
$= \max(0, \max(0, W_h X + b_h)W_0 + b_0)$

2.



\# of parameters Conv Layer $= ((filter\_width \times filter\_height \times n\_filters\_prev\_layer + 1) \times n\_filters)$

Input Layer: $200 \times 300 \times 3$
\# of parameters Conv Layer1: $(3 * 3 * 3 + 1) * 100 = 2800$
\# of parameters Conv Layer2: $(3 * 3 * 100 + 1) * 200 = 180200$
\# of parameters Conv Layer3: $(3 * 3 * 200 + 1) * 400 = 720400$

Total # of parameters $= 2800 + 180200 + 720400 = 903400$

3.

a. $\frac{\partial f}{\partial \gamma} = \frac{\partial f(y)}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial f}{\partial y_i}\frac{\partial y_i}{\partial \gamma} = \sum_{i=1}^{m} \frac{\partial f}{\partial y_i}\hat{x}_i$

b. $\frac{\partial f}{\partial \beta} = \frac{\partial f(y)}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial f}{\partial y_i}\frac{\partial y_i}{\partial \beta} = \sum_{i=1}^{m} \frac{\partial f}{\partial y_i} \cdot 1 = \sum_{i=1}^{m} \frac{\partial f}{\partial y_i}$

c. $\frac{\partial f}{\partial \hat{x}_i} = \frac{\partial f}{\partial y_i}\frac{\partial y_i}{\partial \hat{x}_i} = \frac{\partial f}{\partial y_i}\gamma$

d. $\frac{\partial f}{\partial \sigma^2} = \sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i}\frac{\partial \hat{x}_i}{\partial \sigma^2} = \sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i}\frac{\partial}{\partial \sigma^2}\left(\frac{x_i-\mu}{\sqrt{\sigma^2+\varepsilon}}\right) = \sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i}\left(-\frac{1}{2}\left(\frac{x_i-\mu}{(\sigma^2+\varepsilon)^{\frac{3}{2}}}\right)\right) =$

$$= -\frac{1}{2}\sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i}\frac{x_i-\mu}{(\sigma^2+\varepsilon)^{\frac{3}{2}}}$$

e. $\frac{\partial f}{\partial \mu} = \sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i}\frac{\partial \hat{x}_i}{\partial \mu} + \frac{\partial f}{\partial \sigma^2}\frac{\partial \sigma^2}{\partial \mu} = \sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i}\frac{-1}{\sqrt{\sigma^2+\varepsilon}} + \frac{\partial f}{\partial \sigma^2}\frac{-2}{m}\sum_{i=1}^{m}(x_i-\mu) =$

$$= -\frac{1}{\sqrt{\sigma^2+\varepsilon}}\sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i} - 2\frac{\partial f}{\partial \sigma^2}\frac{1}{m}\sum_{i=1}^{m}(x_i-\mu) =$$

$$= -\frac{1}{\sqrt{\sigma^2+\varepsilon}}\sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i} - 2\frac{\partial f}{\partial \sigma^2}\left(\frac{1}{m}\sum_{i=1}^{m} x_i - \mu\right) =$$

$$= -\frac{1}{\sqrt{\sigma^2+\varepsilon}}\sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i} - 2\frac{\partial f}{\partial \sigma^2}(\mu-\mu) =$$

$$= -\frac{1}{\sqrt{\sigma^2+\varepsilon}}\sum_{i=1}^{m} \frac{\partial f}{\partial \hat{x}_i}$$

f. $\frac{\partial f}{\partial x_i} = \frac{\partial f}{\partial \hat{x}_i}\frac{\partial \hat{x}_i}{\partial x_i} + \frac{\partial f}{\partial \mu}\frac{\partial \mu}{\partial x_i} + \frac{\partial f}{\partial \sigma^2}\frac{\partial \sigma^2}{\partial x_i} =$

$$= \frac{\partial f}{\partial \hat{x}_i}\frac{1}{\sqrt{\sigma^2+\varepsilon}} - \frac{1}{\sqrt{\sigma^2+\varepsilon}}\sum_{j=1}^{m} \frac{\partial f}{\partial \hat{x}_j}\frac{1}{m} - \frac{1}{2}\sum_{j=1}^{m} \frac{\partial f}{\partial \hat{x}_j}\frac{x_j-\mu}{(\sigma^2+\varepsilon)^{\frac{3}{2}}}\frac{2(x_i-\mu)}{m} =$$

$$= \frac{1}{\sqrt{\sigma^2+\varepsilon}}\frac{\partial f}{\partial \hat{x}_i} - \frac{1}{m\sqrt{\sigma^2+\varepsilon}}\sum_{j=1}^{m} \frac{\partial f}{\partial \hat{x}_j} - \frac{1}{m\sqrt{\sigma^2+\varepsilon}}\frac{x_i-\mu}{\sqrt{\sigma^2+\varepsilon}}\sum_{j=1}^{m} \frac{\partial f}{\partial \hat{x}_j}\frac{x_j-\mu}{\sqrt{\sigma^2+\varepsilon}} =$$

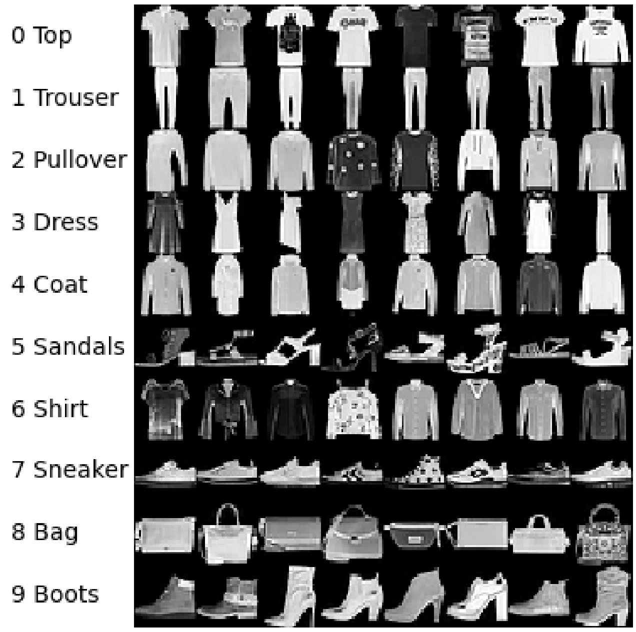
$$= \frac{1}{m\sqrt{\sigma^2+\varepsilon}}\left(m\frac{\partial f}{\partial \hat{x}_i} - \sum_{j=1}^{m} \frac{\partial f}{\partial \hat{x}_j} - \frac{x_i-\mu}{\sqrt{\sigma^2+\varepsilon}}\sum_{j=1}^{m} \frac{\partial f}{\partial \hat{x}_j}\frac{x_j-\mu}{\sqrt{\sigma^2+\varepsilon}}\right) =$$

$$= \frac{1}{m\sqrt{\sigma^2+\varepsilon}}\left(m\frac{\partial f}{\partial \hat{x}_i} - \sum_{j=1}^{m} \frac{\partial f}{\partial \hat{x}_j} - \hat{x}_i\sum_{j=1}^{m} \frac{\partial f}{\partial \hat{x}_j}\hat{x}_j\right)$$

Practical

The Data:

The data we use for this exercise is Fashion-MNIST, a dataset of Zalando's article images—consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image, associated with a label from 10 classes.



The Model:

LeNet is a convolutional neural network structure proposed by LeCun et al. in 1998.[1] In general, LeNet refers to LeNet-5 and is a simple convolutional neural network. (Wikipedia)

Baseline model – LeNet5:

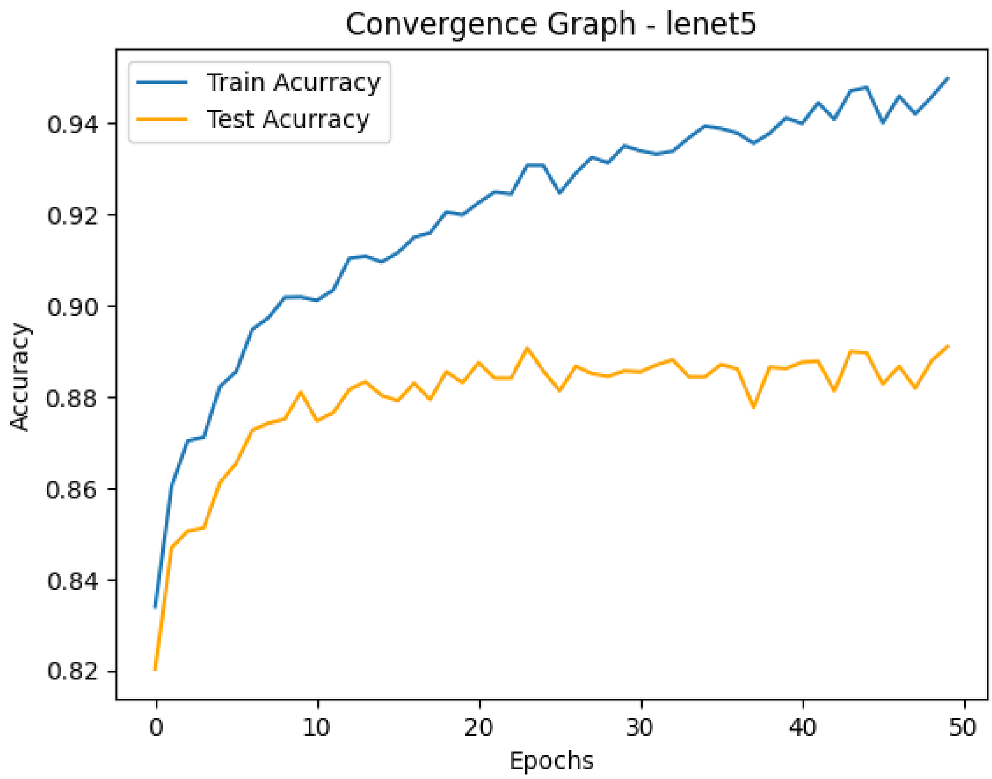
Layer (type)	**Output Shape**	**Param #**
Conv2d	[-1, 6, 28, 28]	156
Tanh	[-1, 6, 28, 28]	0
AvgPool2d	[-1, 6, 14, 14]	0
Conv2d	[-1, 16, 10, 10]	2,416
Tanh	[-1, 16, 10, 10]	0
AvgPool2d	[-1, 16, 5, 5]	0
Conv2d	[-1, 120, 1, 1]	48,120
Flatten	[-1, 120]	0
Linear	[-1, 84]	10,164
Tanh	[-1, 84]	0
Linear	[-1, 10]	850
		Total params: 61,706

Techniques to compare:

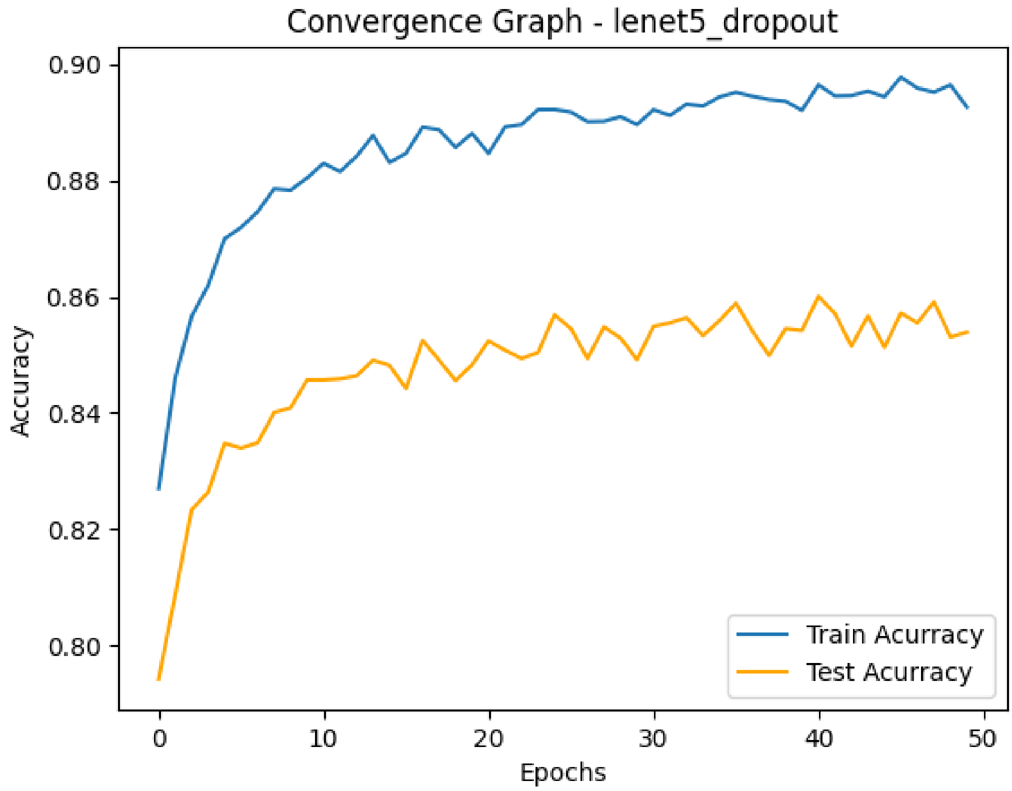
- Dropout
- Weight Decay
- Batch Normalization

Results:

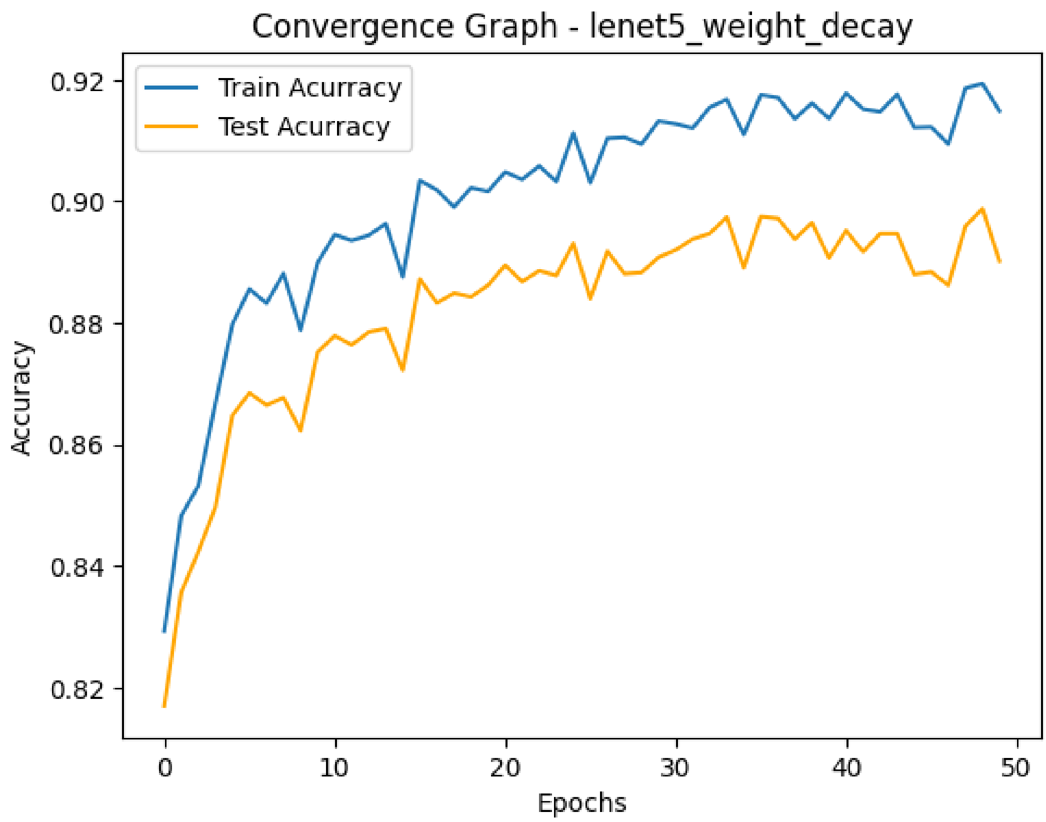
LeNet5 (Baseline):



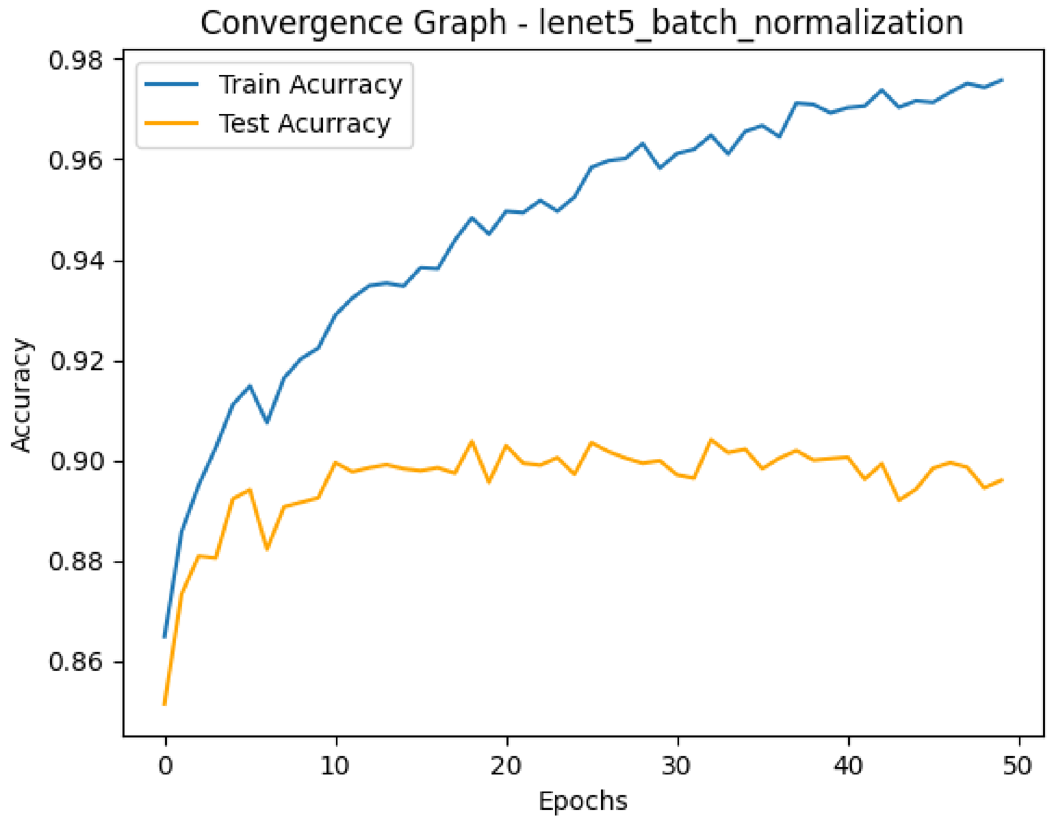
LeNet5 with Dropout:



LeNet5 with Weight Decay:



LeNet5 with Batch Normalization:



Summary of all accuracies:

Model	Train Accuracy [%]	Test Accuracy [%]
lenet5	94.98	89.1
lenet5_dropout	89.26	85.39
lenet5_weight_decay	91.49	89.02
lenet5_batch_normalization	97.58	89.6

Conclusion:

- We got the best accuracy with the Batch Normalization technique, both on Train and Test sets.
- The Train accuracy of the Batch Normalization technique is almost perfect and can get better with more epochs.
- We got the worst Accuracy with the Dropout technique, both on Train and Test sets. Even worse results compared to the Baseline model.
- We used the validation for fine tuning the general model. For example, to choose the optimizer and size of epochs.