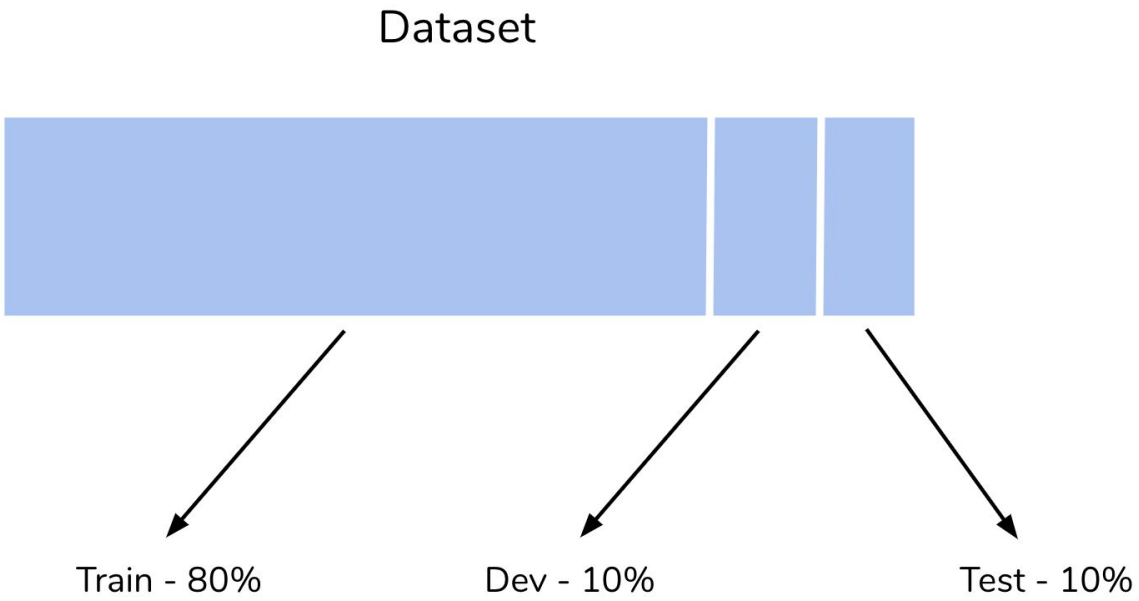
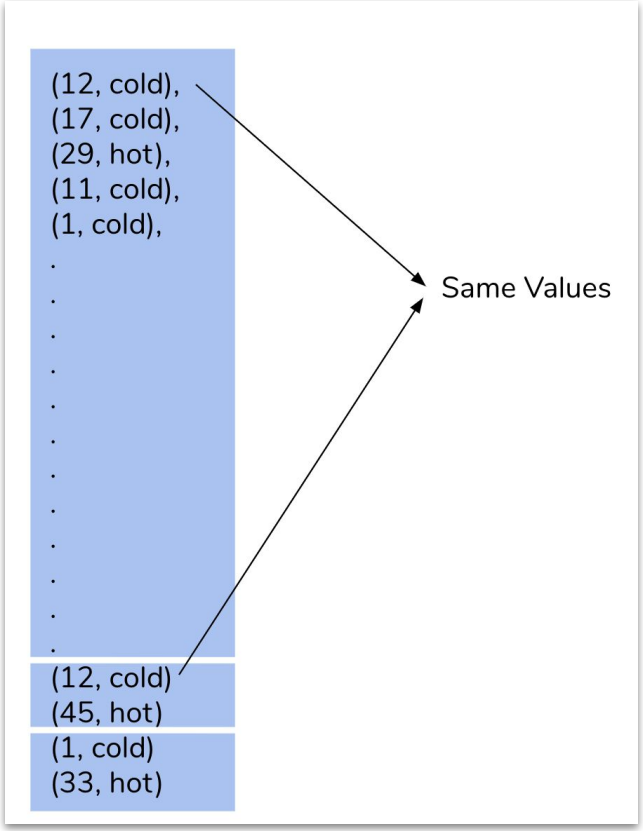
Deep Learning for NLP - Focus on Medical Applications

ANN - Training Procedure

1. Split the dataset into Train/Dev/Test

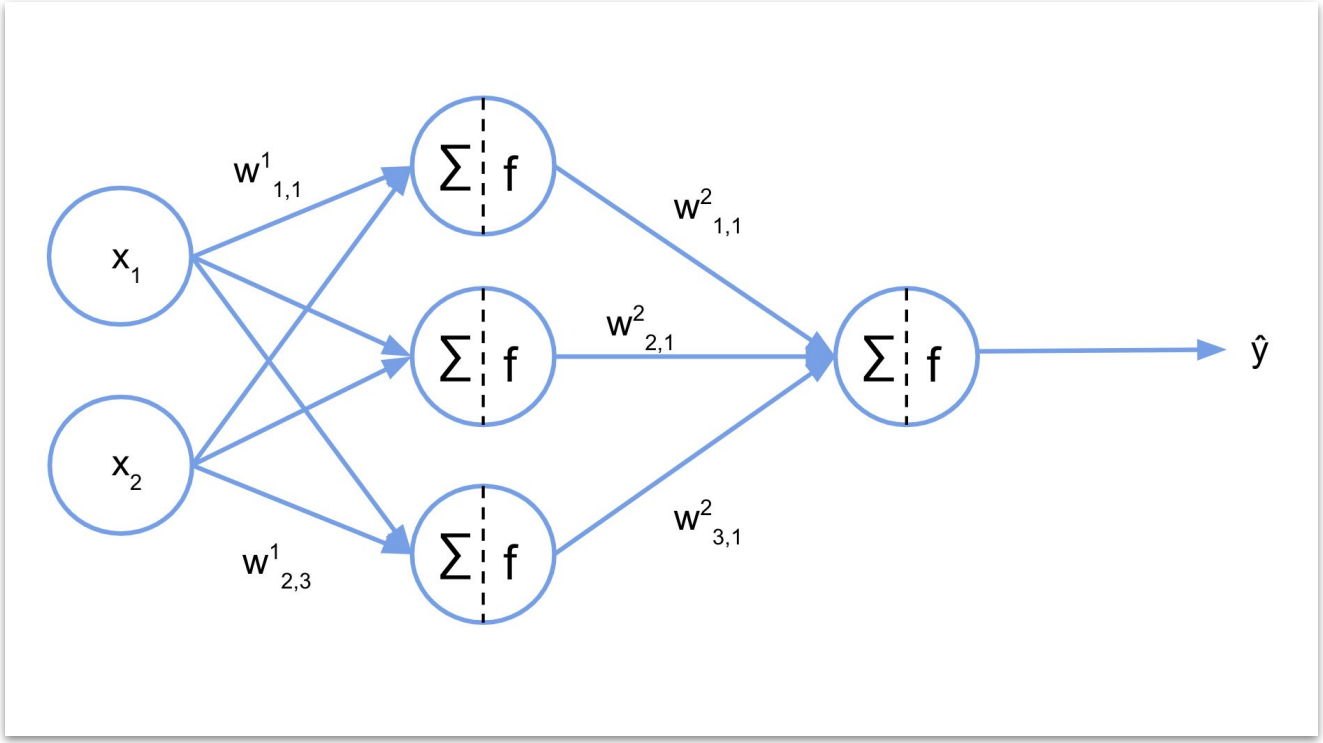


1. Split the dataset into Train/Dev/Test - Disjoint



The values from one portion of the dataset should not appear in any other. If they do, the reported accuracy will not be correct.

2. Build the Network

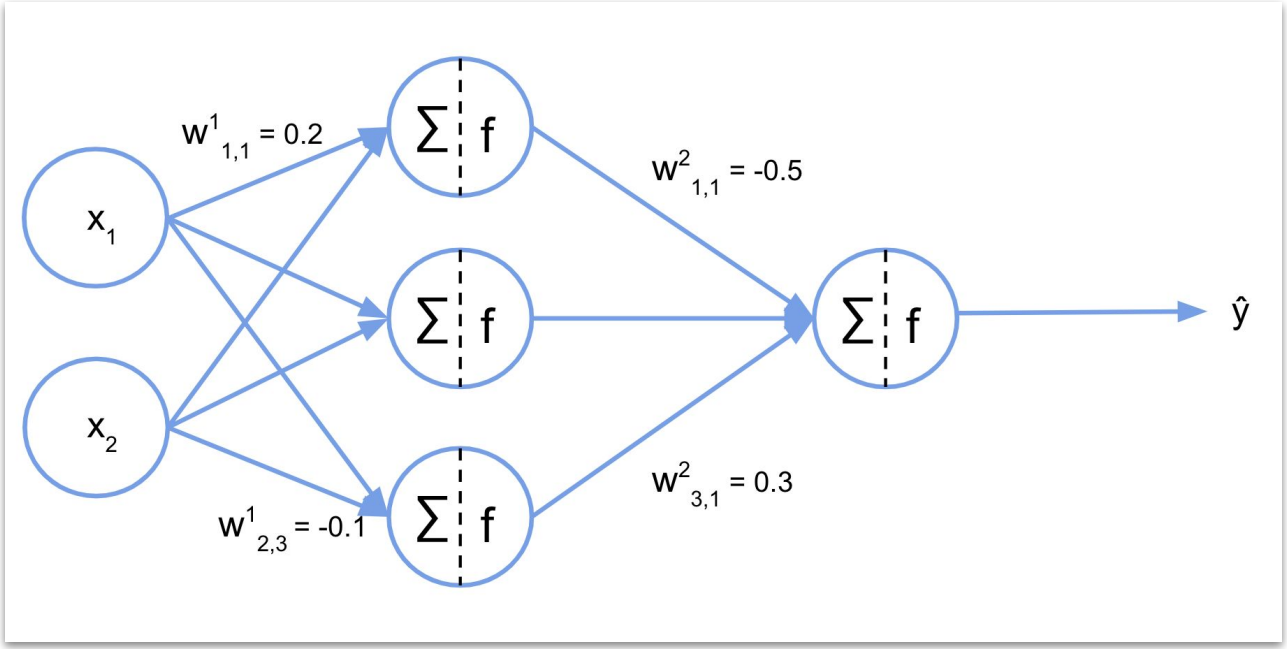


$$W^1 = \begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} \\ w_{2,1} & w_{2,2} & w_{2,3} \end{bmatrix}$$

$$W^2 = \begin{bmatrix} w_{1,1} & w_{2,1} & w_{3,1} \end{bmatrix}$$

We also have the biases:
B1, B2

3. Initialize the Network



$$W^1 = \begin{bmatrix} 0.2 & 0.1 & -0.3 \\ -0.4 & 0.3 & -0.1 \end{bmatrix}$$

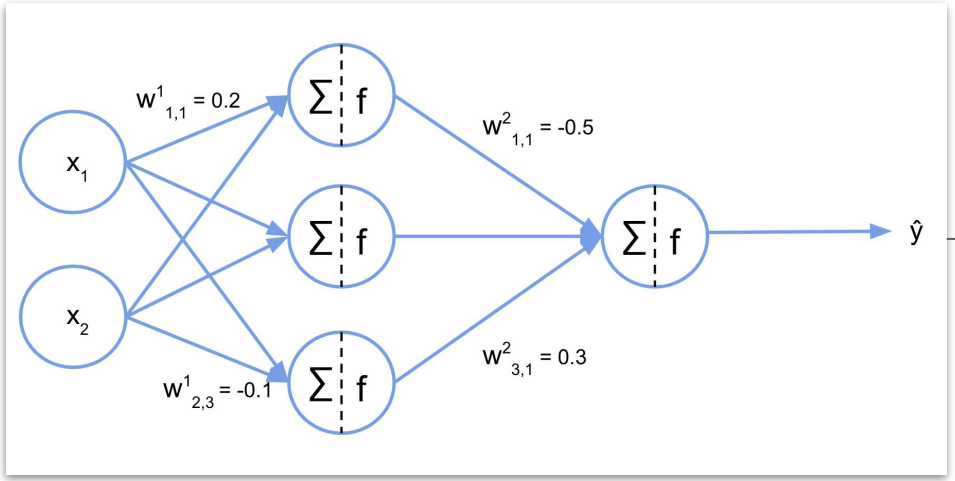
$$B^1 = \begin{bmatrix} 1 \\ 0.4 \\ 0.5 \end{bmatrix}$$

$$W^2 = \begin{bmatrix} -0.5 \\ 0.7 \\ 0.3 \end{bmatrix}$$

$$B^2 = \begin{bmatrix} 0.4 \end{bmatrix}$$

4. Forward Pass - Classification

(12, 2),
(17, 3),
(29, 2),
(11, 1),
(1, 5),
.
.
.
.
.
.
.
.
.
.
.
.
.

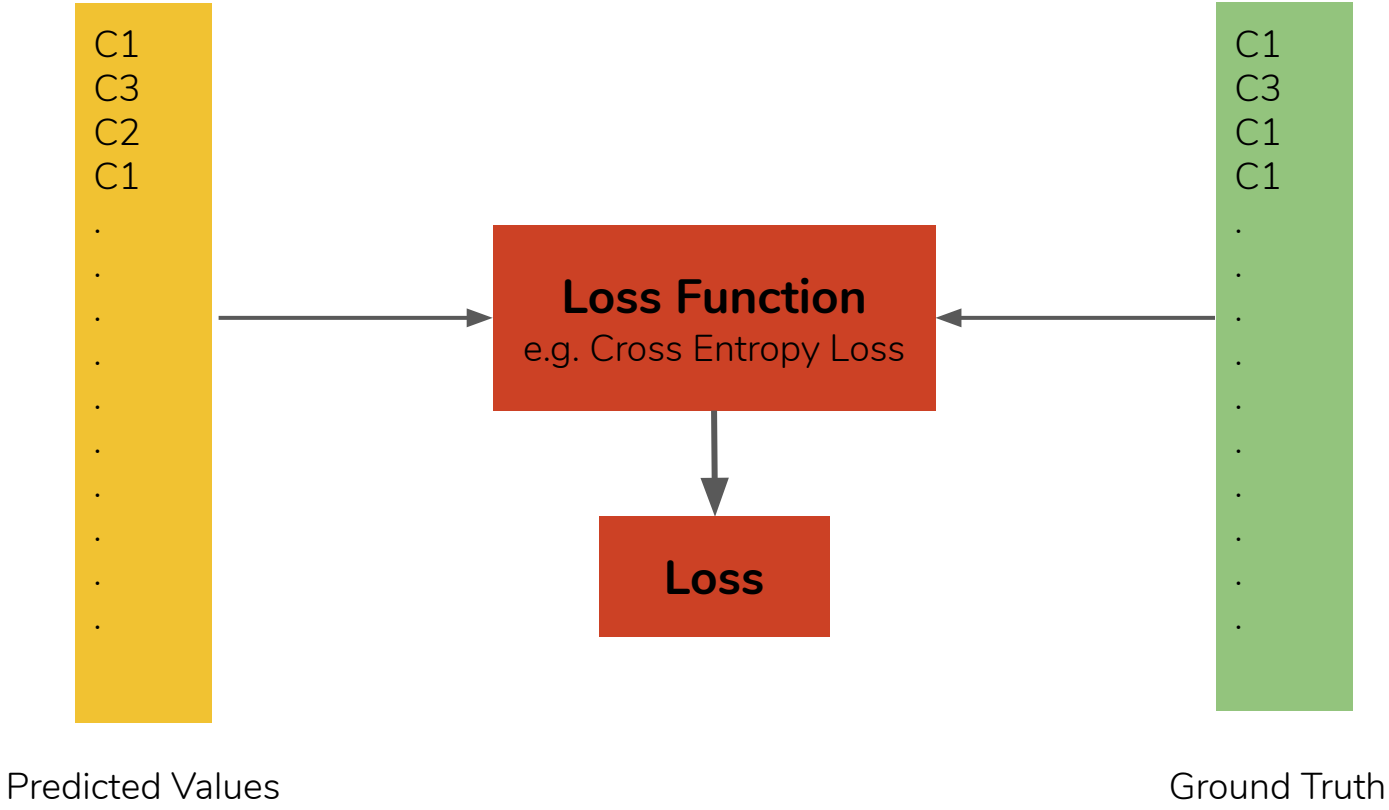


C1
C3
C2
C1
.
.
.
.
.
.
.
.
.
.
.
.
.

Training Dataset (X)

Predicted Classes (Yh)

5. Error Calculation

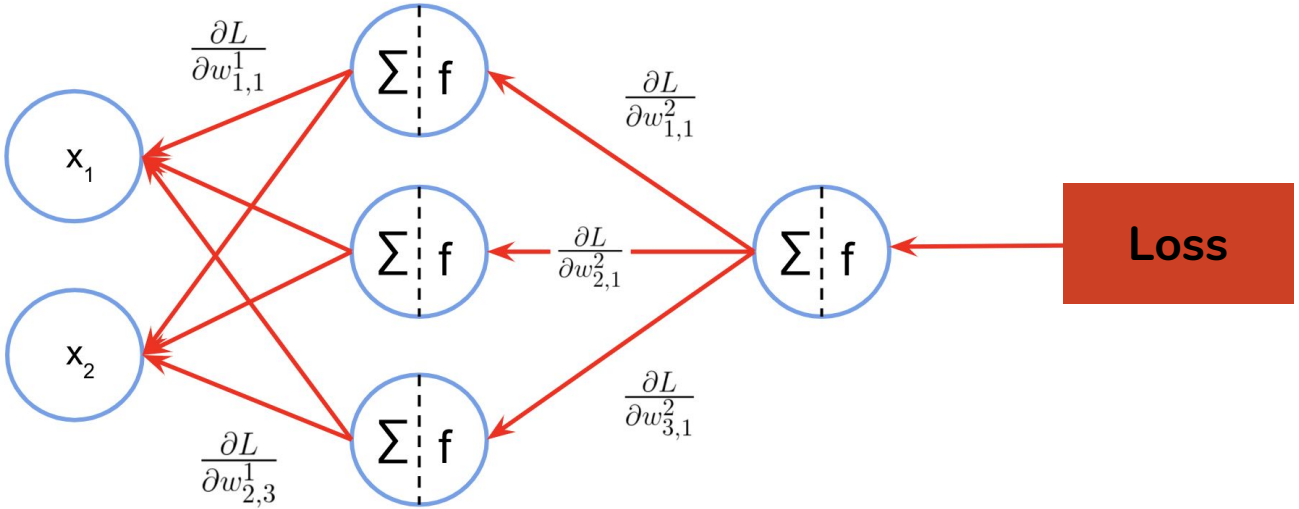


6. Backpropagation

$$\frac{\partial L}{\partial w_{1,1}^2} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial z_1^2} \frac{\partial z_1^2}{\partial w_{1,1}^2}$$

L – Loss

$\hat{y}$ – Last layer output

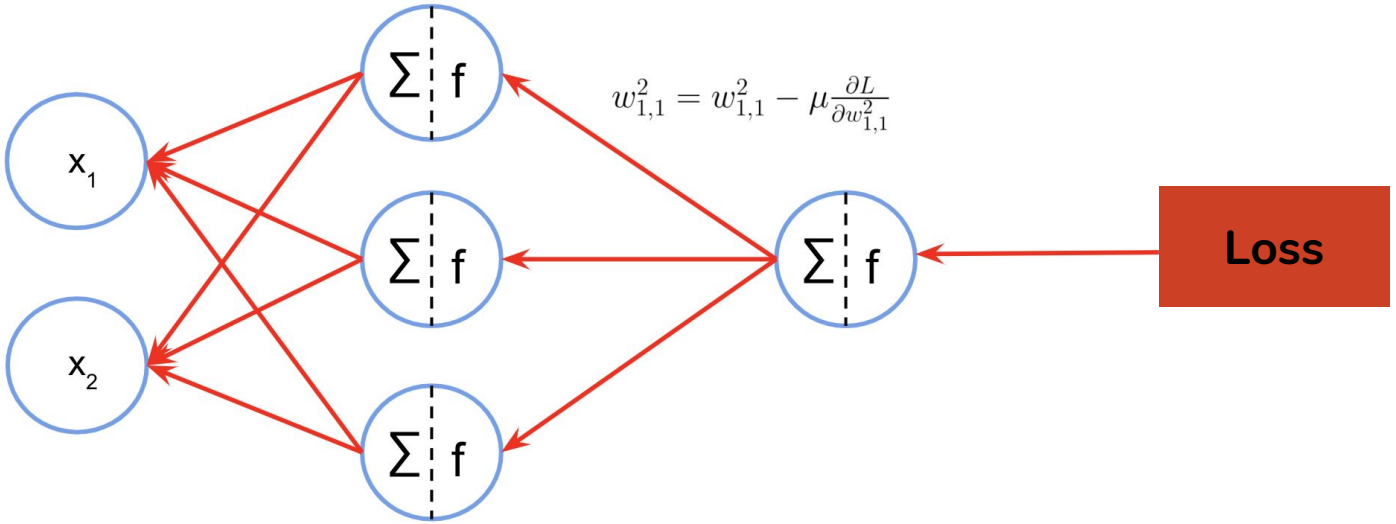


7. Optimization - Weight update

$$w^2_{1,1} = w^2_{1,1} - \mu \frac{\partial L}{\partial w^2_{1,1}}$$

μ – Learning rate

L – Loss



8. Repeat

- Run the forward/backward/optimize part again - repeat until it works
- While running the training plot everything - really everything
- Once we are happy with the accuracy/loss run one final time on the test dataset