

Looking back

Database commands

- Create a database
- Use a database
- Delete a database
- List all databases

Table commands

- Create a table
- Alter a table
- Delete a table
- List all tables
- Describe a table

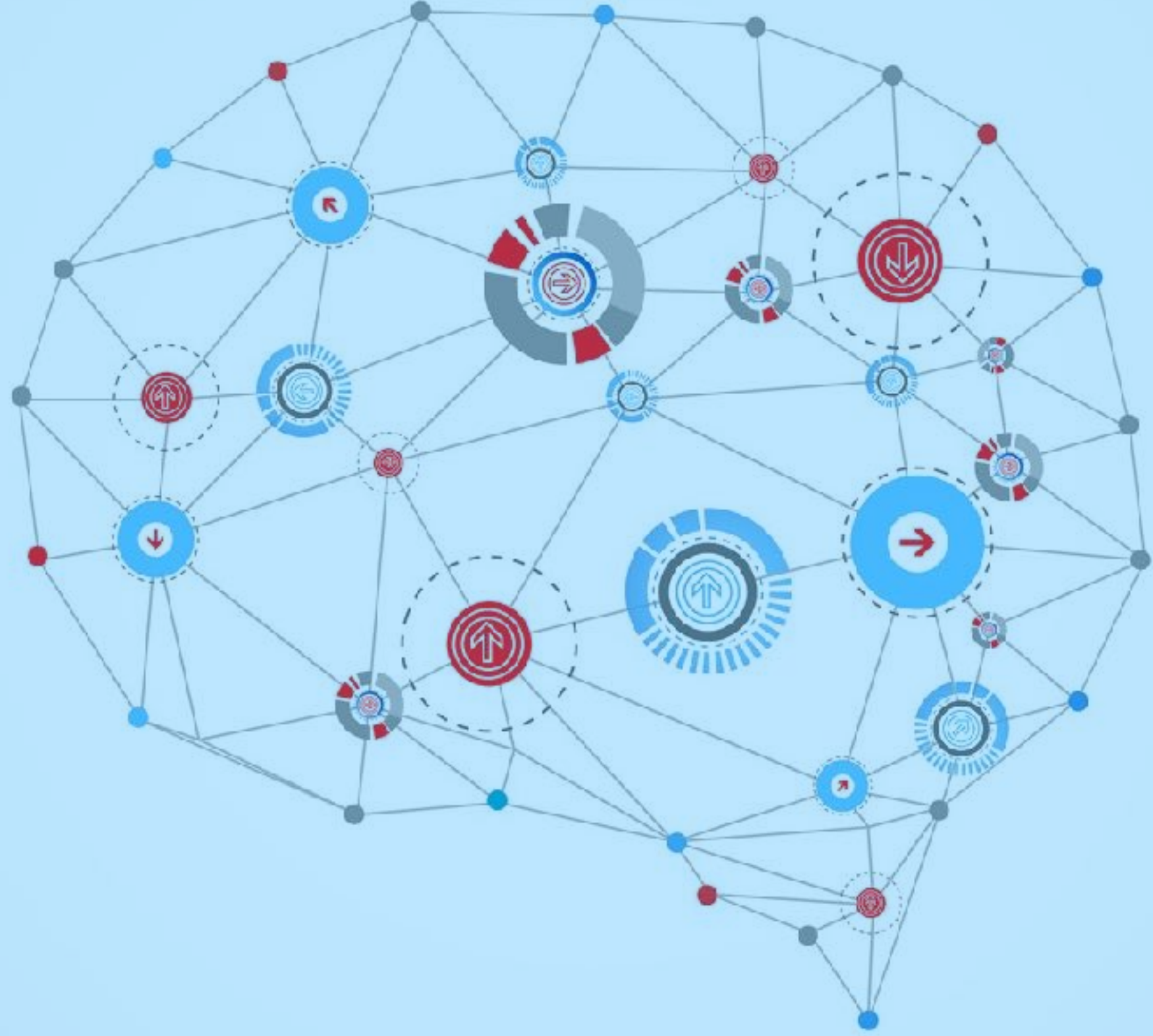
Document commands

- Insert a document
- Show documents
- Delete a document
- Update document

Constraints

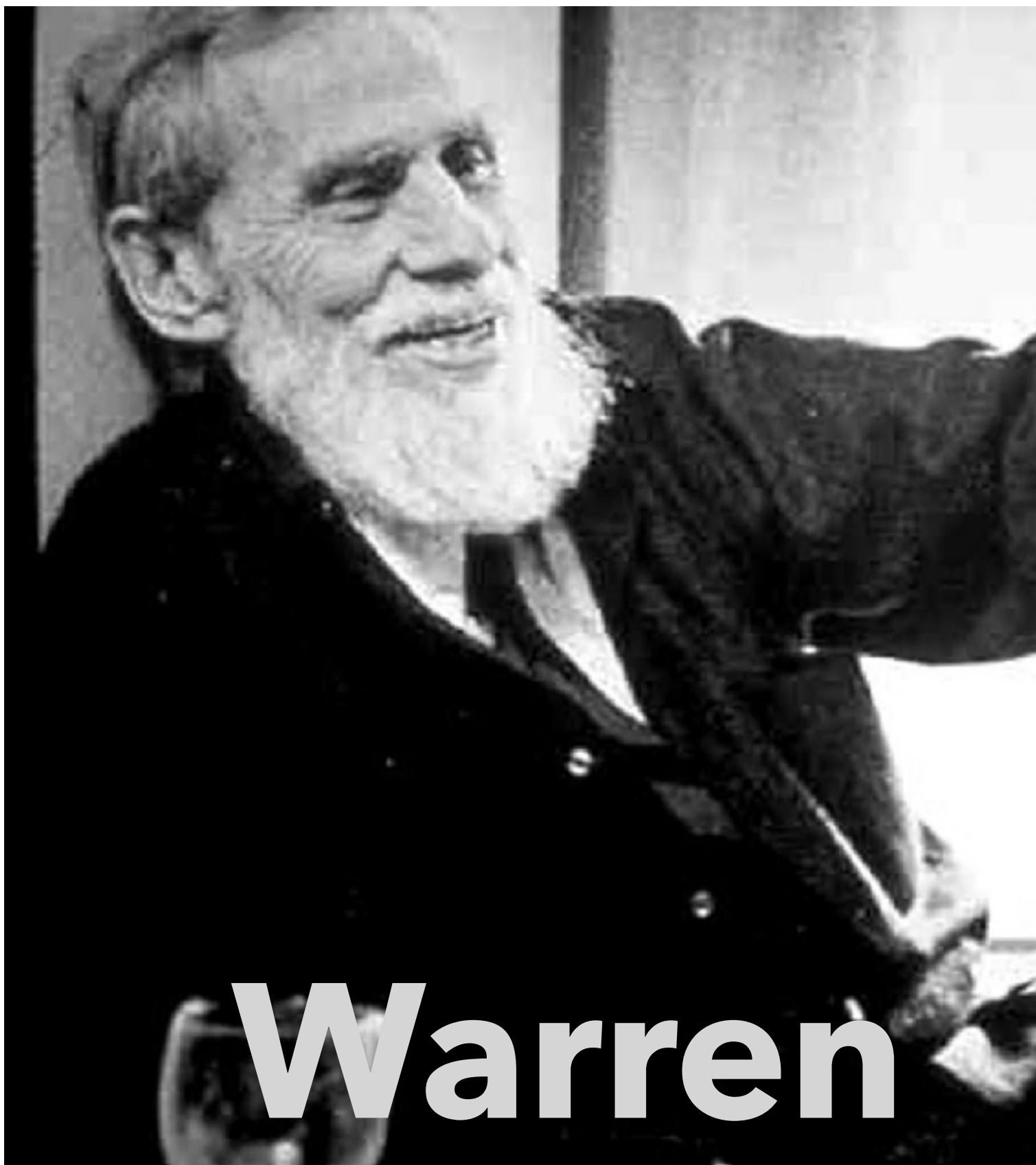
- Not Null constraint
- Default constraint
- Unique constraint
- Primary Key constraint
- Foreign Key constraint
- Check constraint

Deep Learning

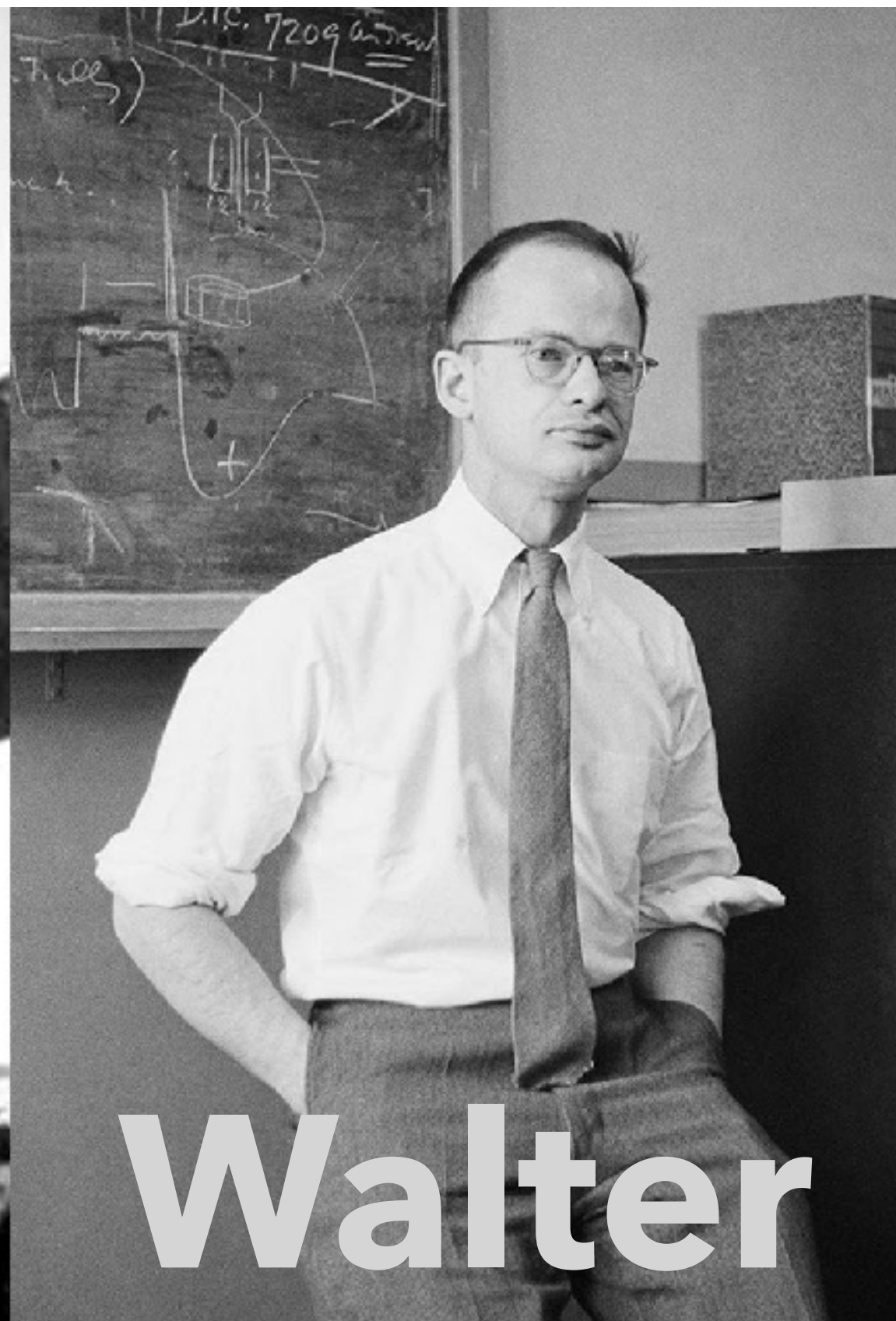


A black and white photograph of a B-29 bomber flying over a mountainous landscape. The aircraft is in the upper center of the frame, flying towards the right. Below it, a large mountain range is visible, with a prominent peak on the right side. The foreground shows a dark, rocky or forested area. The overall tone is historical and dramatic.

1943

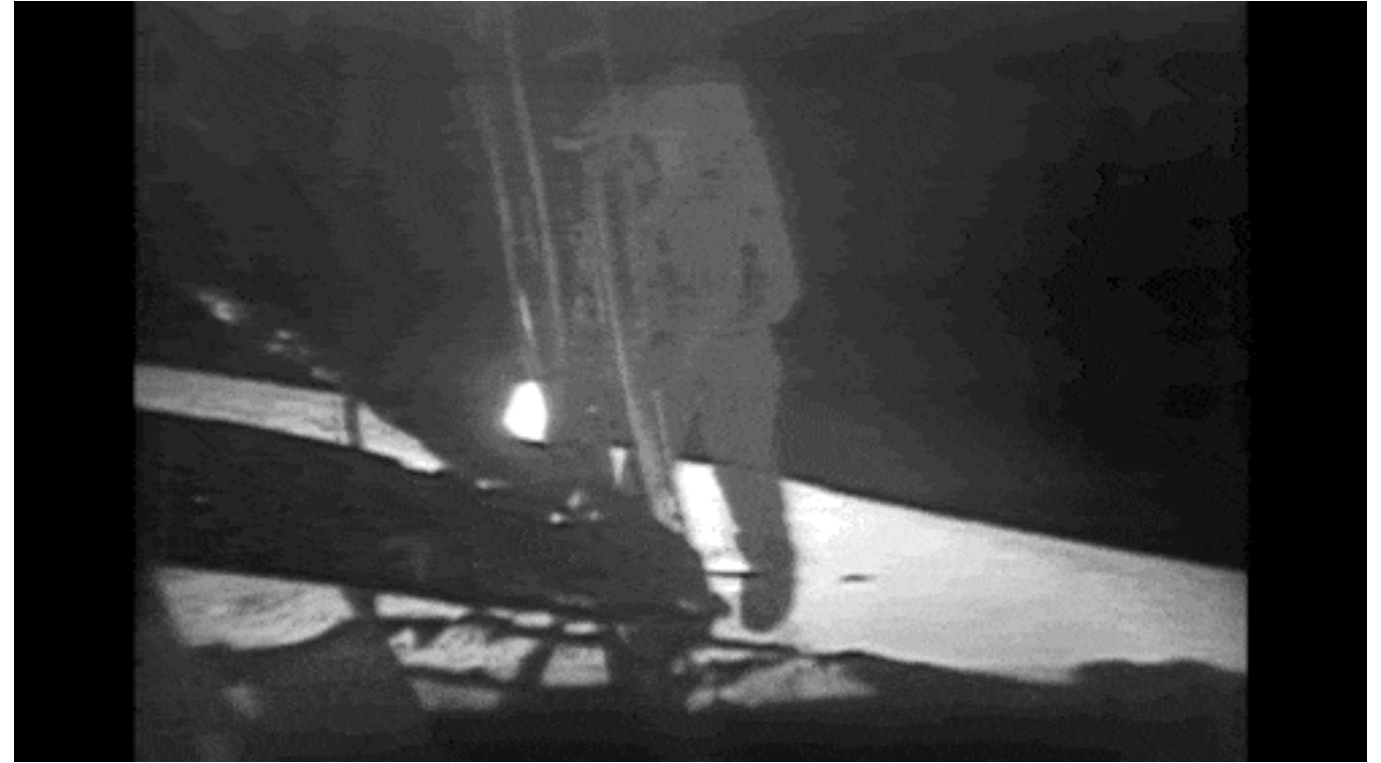
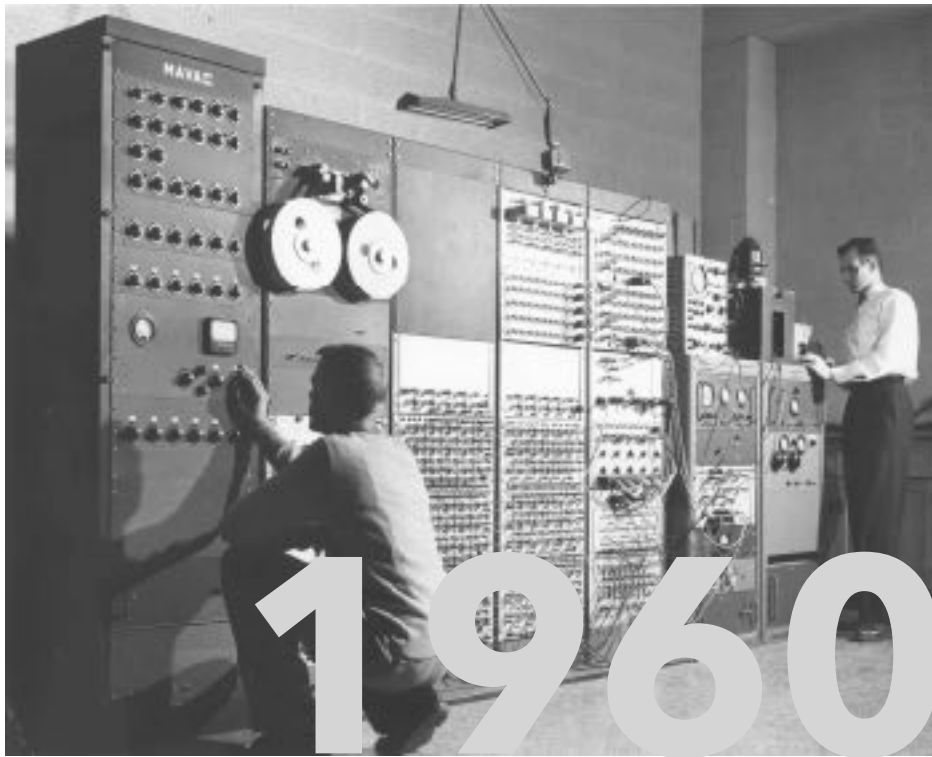


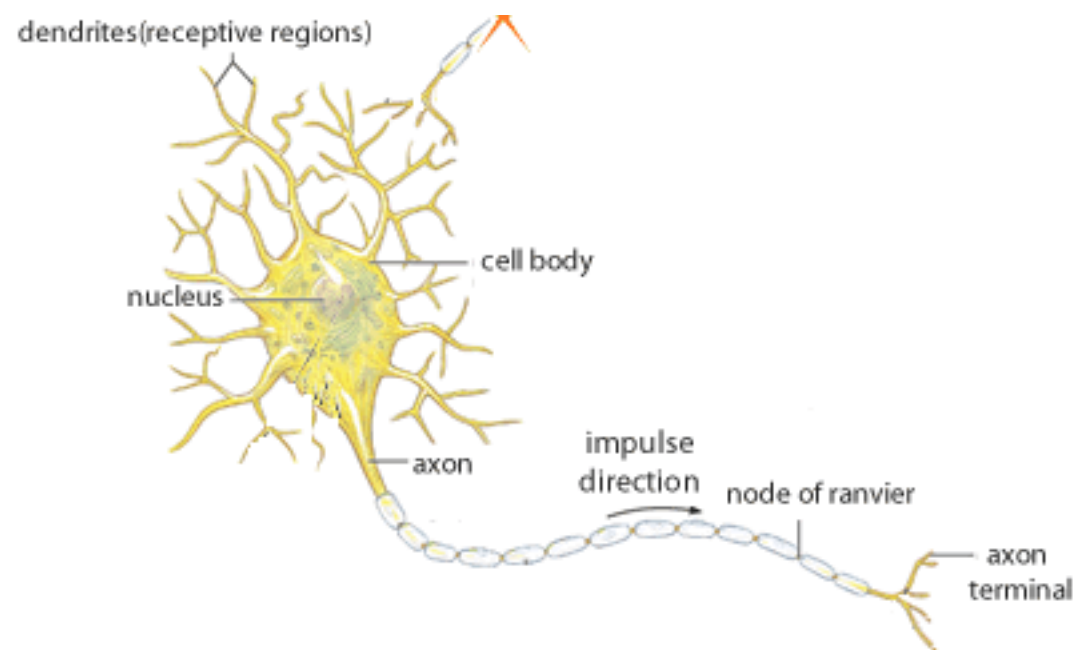
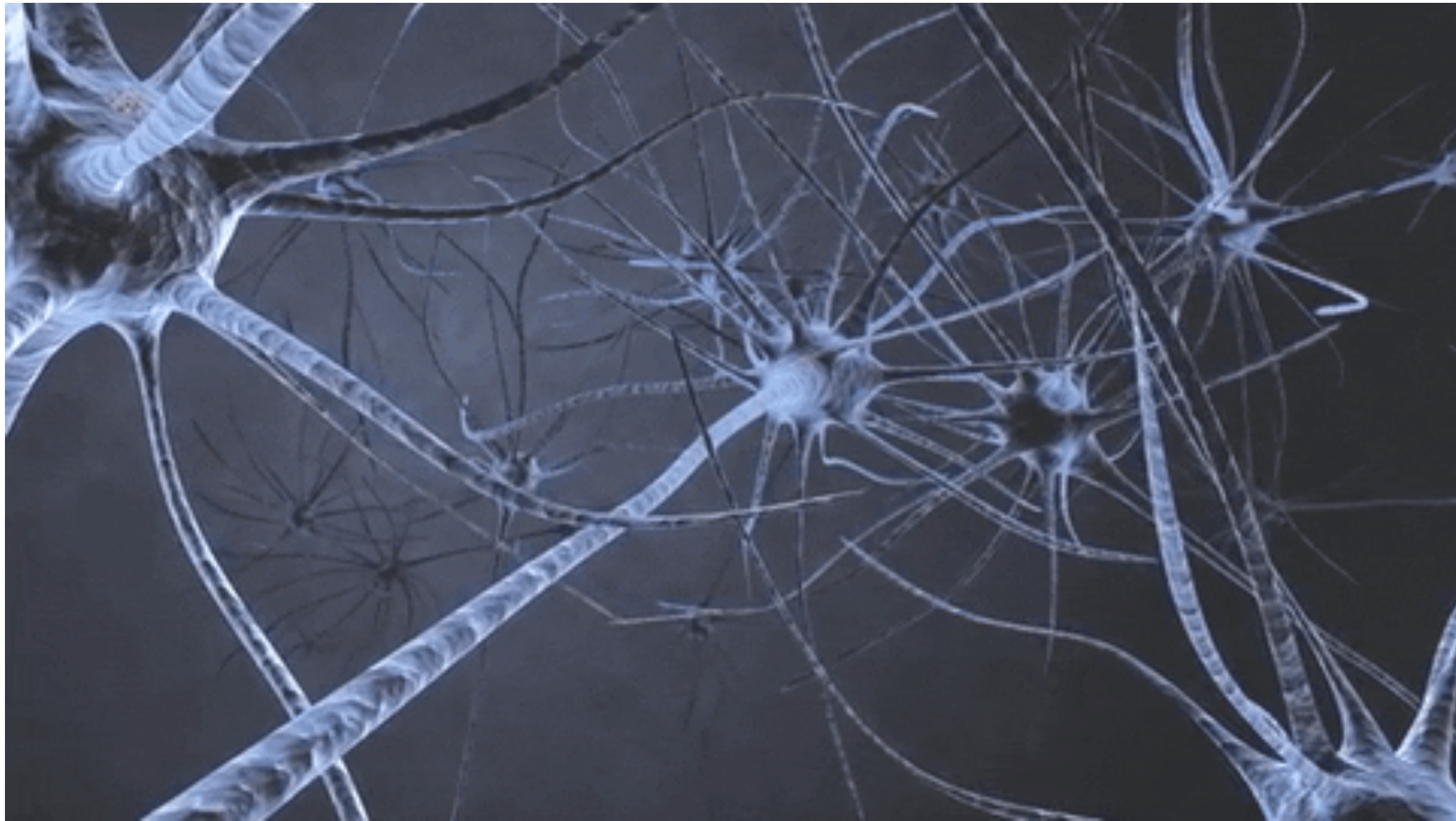
Warren



Walter

Why deep learning got all attention now?





Artificial Neural Network



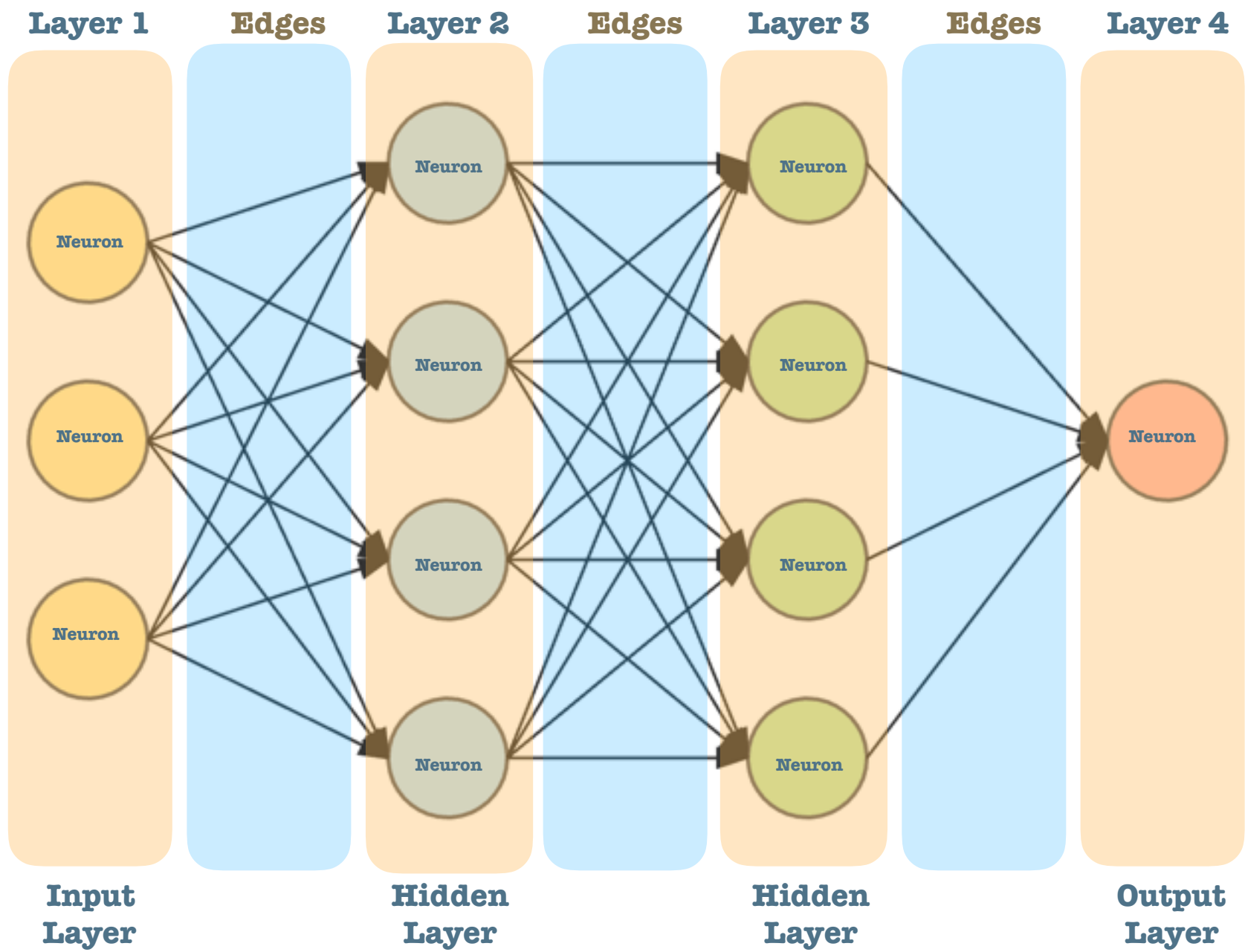
| training inputs | training targets | training outputs | | | |
|--------------------|--------------------|------------------|----------|----------|----------|
| 0.0, 0.0, 0.0, 0.0 | 0.0, 0.0, 0.0, 1.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 0.0, 0.0, 0.0, 1.0 | 0.0, 0.0, 1.0, 0.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 0.0, 0.0, 1.0, 0.0 | 0.0, 0.0, 1.0, 1.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 0.0, 0.0, 1.0, 1.0 | 0.0, 1.0, 0.0, 0.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 0.0, 1.0, 0.0, 0.0 | 0.0, 1.0, 0.0, 1.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 0.0, 1.0, 0.0, 1.0 | 0.0, 1.0, 1.0, 0.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 0.0, 1.0, 1.0, 0.0 | 0.0, 1.0, 1.0, 1.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 0.0, 1.0, 1.0, 1.0 | 1.0, 1.0, 1.0, 1.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| 1.0, 0.0, 0.0, 0.0 | 0.0, 0.0, 0.0, 0.0 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |

iteration: 0
 error: 1.468101
 is Training: true
 FPS: 22

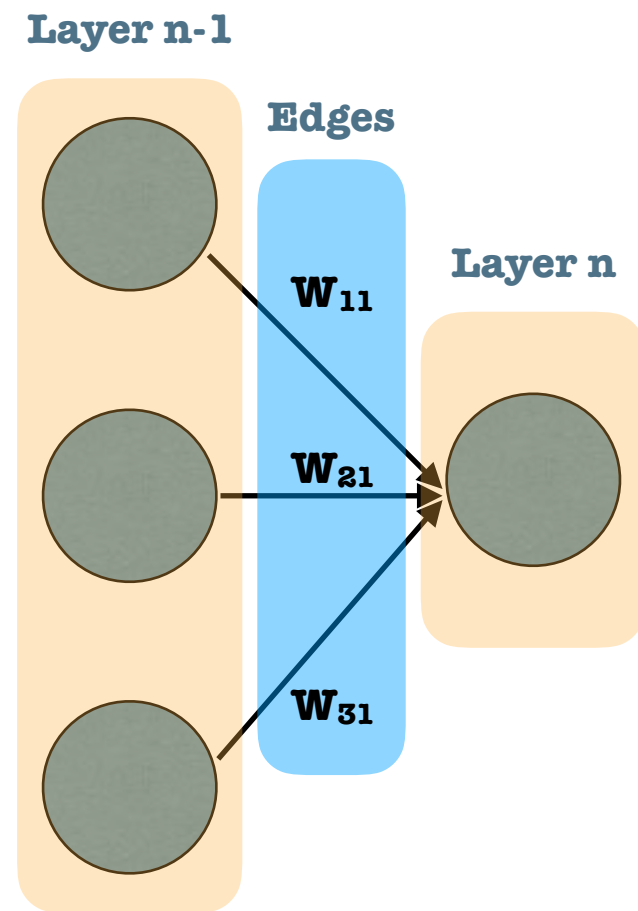
Today's topics

- 1 Artificial Neural Networks**
- 2 Components of neural network**
- 3 Feed forward in neural network**
- 4 Representing NN with matrix**
- 5 Numpy and tensor flow implementation of Feed Forward NN**

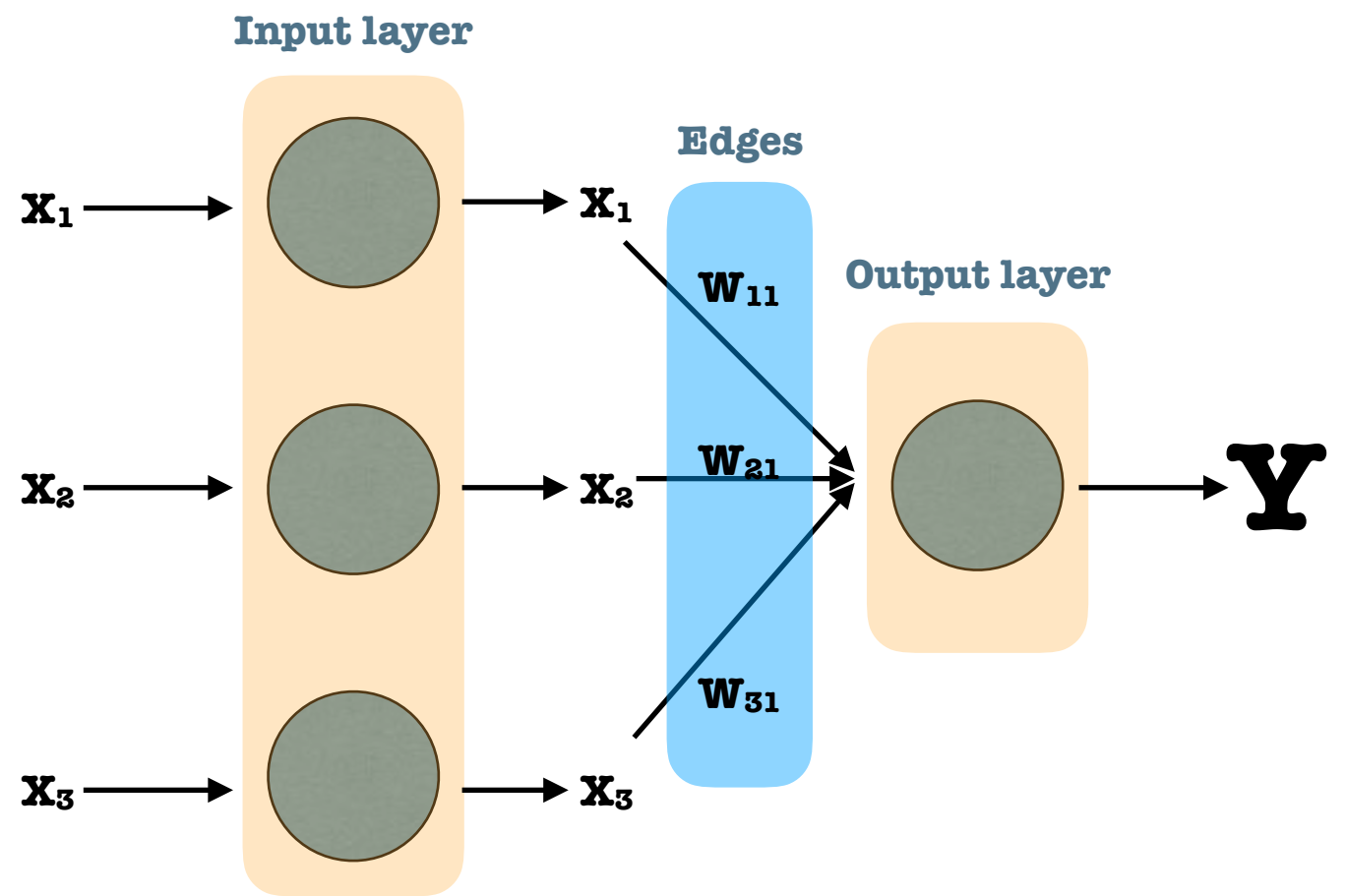
Artificial Neural Network



Examining neuron

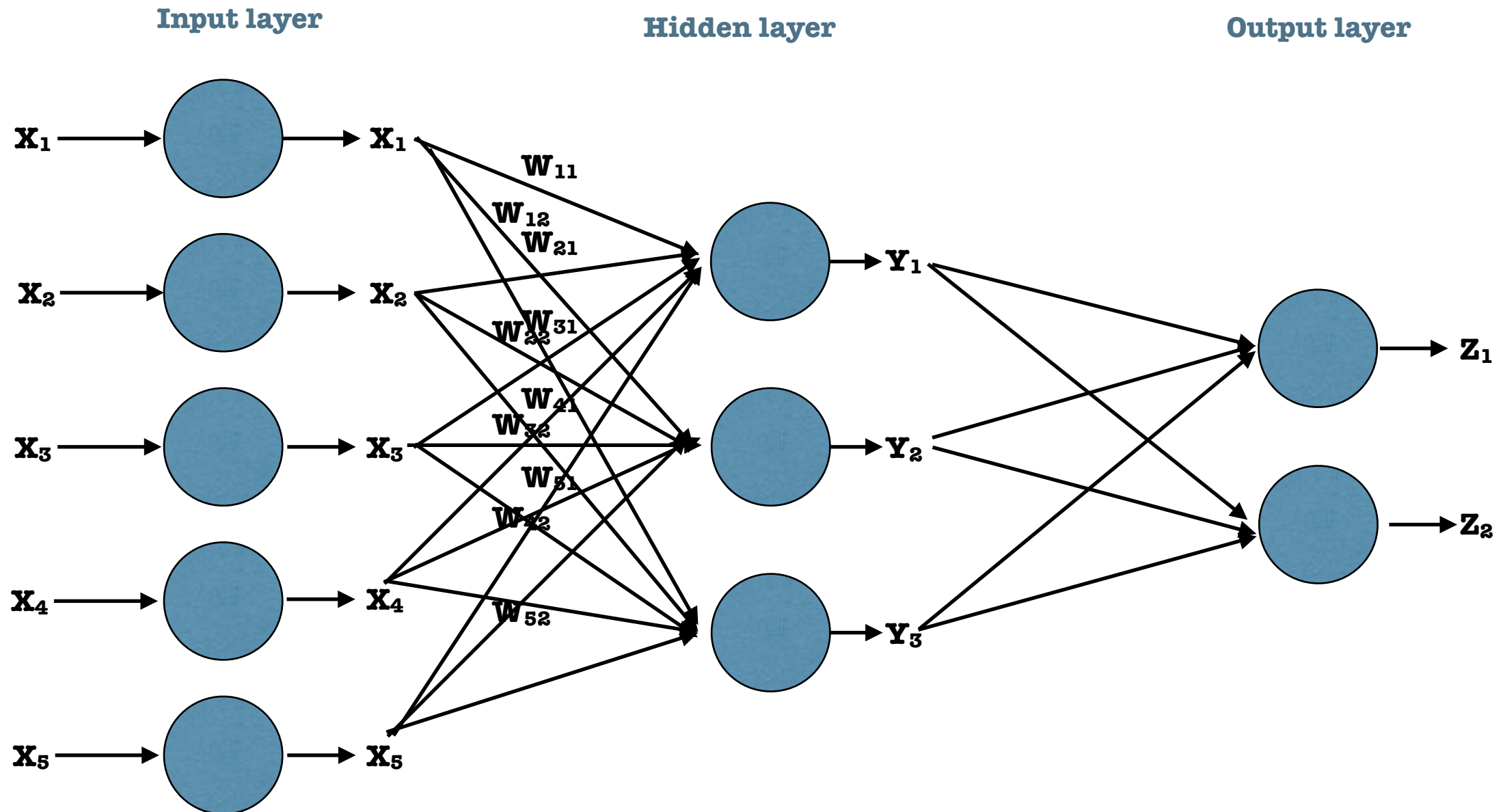


Input to neurons



$$Y = F \left([(x_1 * W_{11}) + (x_2 * W_{21}) + (x_3 * W_{31})] + b_1 \right)$$

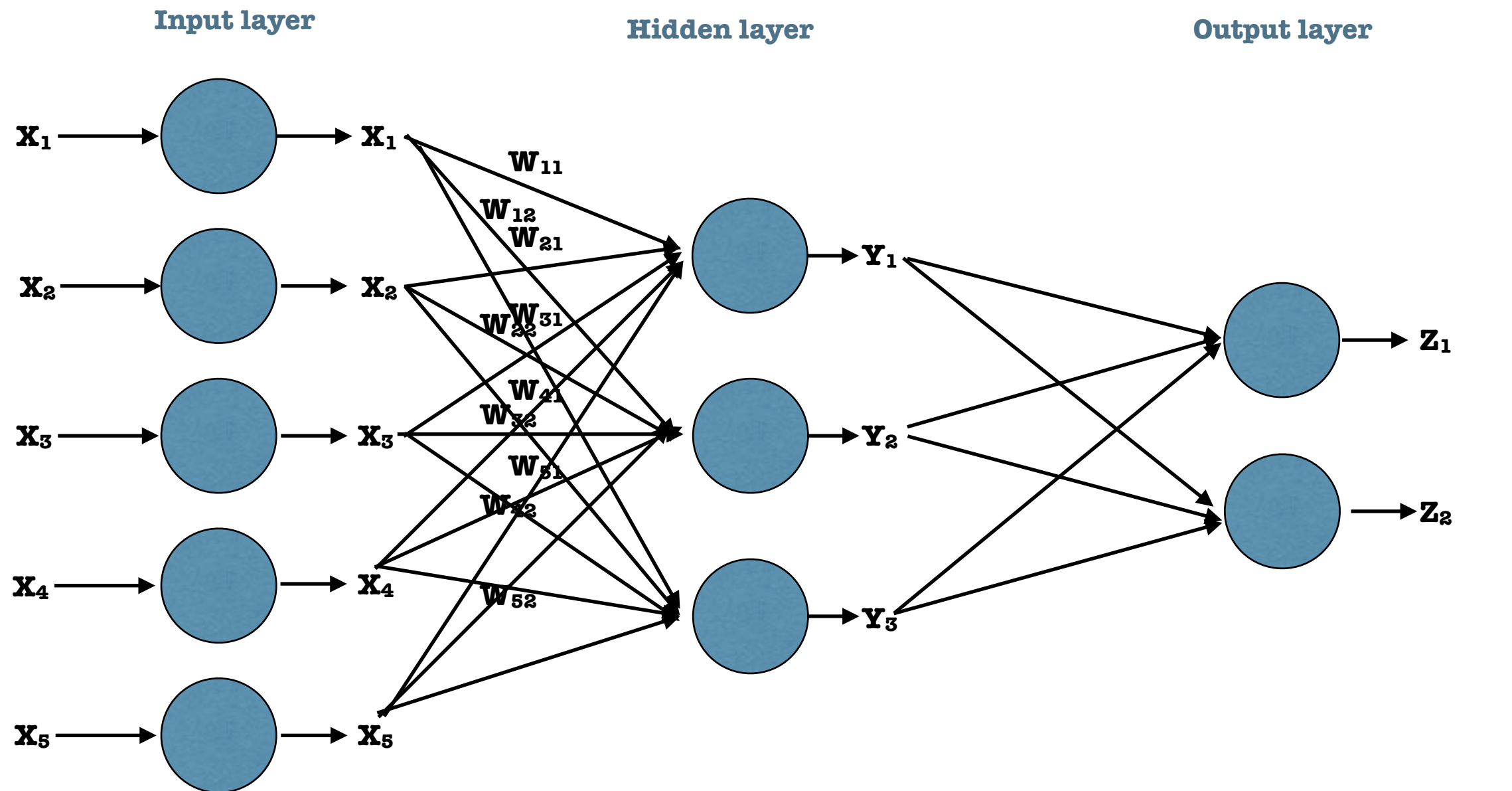
Feed Forward Neural Network



$$Y_1 = F \left([(X_1 * W_{11}) + (X_2 * W_{21}) + (X_3 * W_{31}) + (X_4 * W_{41}) + (X_5 * W_{51})] + HB_1 \right)$$

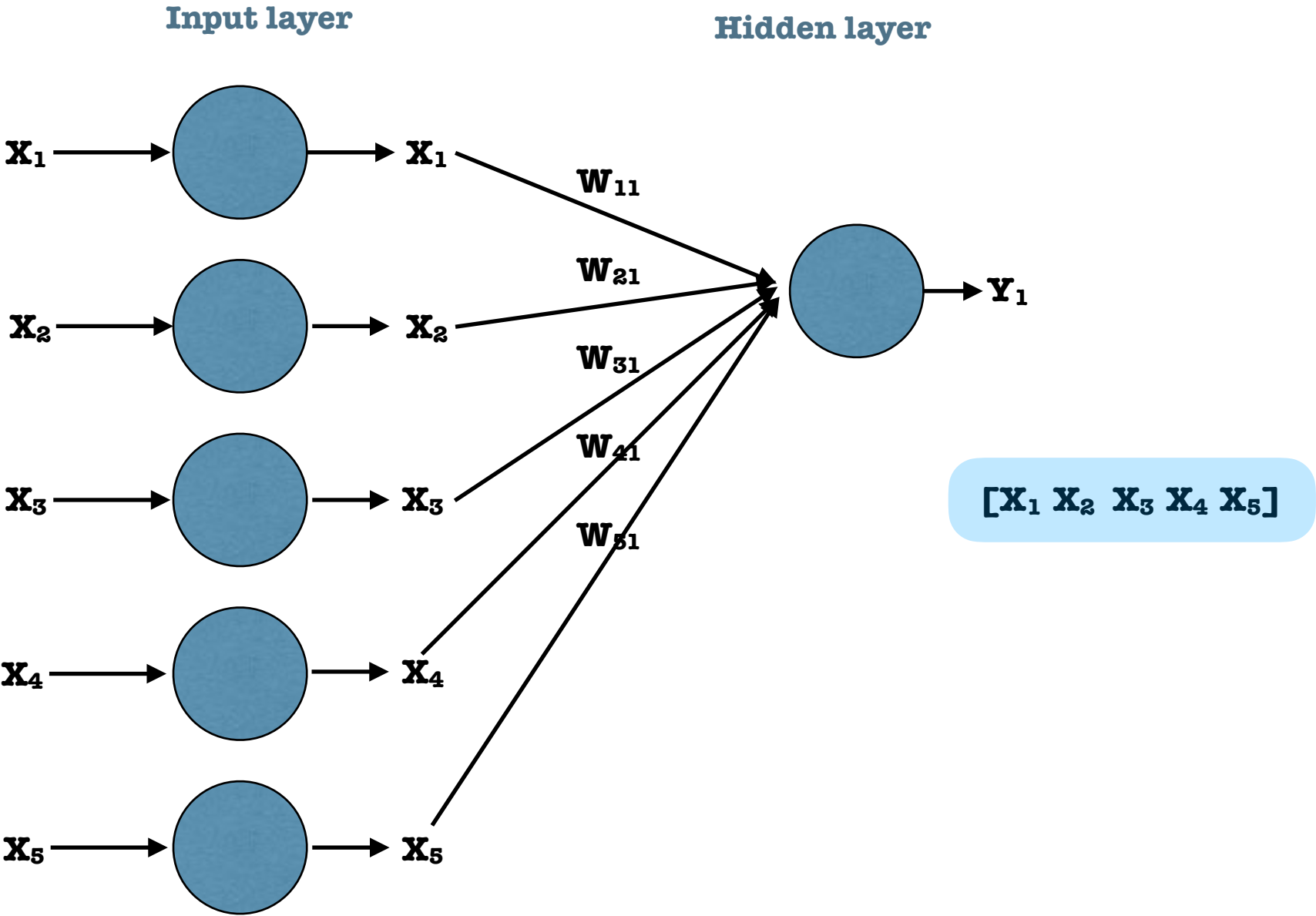
$$Y_2 = F \left([(X_1 * W_{12}) + (X_2 * W_{22}) + (X_3 * W_{32}) + (X_4 * W_{42}) + (X_5 * W_{52})] + HB_2 \right)$$

Neural Network components



| | | | | | | | | | |
|---------------------------------|----------|----------|----------|----------|----------|--|----------|-----------|-----------|
| | | H_{N1} | H_{N2} | H_{N3} | | | O_{N1} | O_{N2} | |
| | I_{N1} | W_{11} | W_{12} | W_{13} | | | H_{N1} | W_{d11} | W_{d12} |
| $[X_1 \ X_2 \ X_3 \ X_4 \ X_5]$ | I_{N2} | W_{21} | W_{22} | W_{23} | $[HB_1]$ | | H_{N2} | W_{d21} | W_{d22} |
| | I_{N3} | W_{31} | W_{32} | W_{33} | $[HB_2]$ | | | | $[OB_1]$ |
| | I_{N4} | W_{41} | W_{42} | W_{43} | $[HB_3]$ | | | | $[OB_2]$ |
| | I_{N5} | W_{51} | W_{52} | W_{53} | | | H_{N3} | W_{d31} | W_{d32} |
| | | W_{61} | W_{62} | W_{63} | | | | | |

Using these components for FF



| | H_{N1} | H_{N2} | H_{N3} |
|----------|----------|----------|----------|
| I_{N1} | w_{11} | w_{12} | w_{13} |
| I_{N2} | w_{21} | w_{22} | w_{23} |
| I_{N3} | w_{31} | w_{32} | w_{33} |
| | w_{41} | w_{42} | w_{43} |
| I_{N4} | w_{51} | w_{52} | w_{53} |

$$y_1 = F \left([(x_1 * w_{11}) + (x_2 * w_{21}) + (x_3 * w_{31}) + (x_4 * w_{41}) + (x_5 * w_{51})] + HB_1 \right)$$

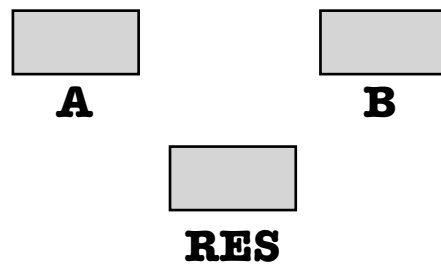
$$\mathbf{Y}_1 = \mathbf{F} \left([(\mathbf{X}_1 \cdot \mathbf{W}_{11}) + (\mathbf{X}_2 \cdot \mathbf{W}_{21}) + (\mathbf{X}_3 \cdot \mathbf{W}_{31}) + (\mathbf{X}_4 \cdot \mathbf{W}_{41}) + (\mathbf{X}_5 \cdot \mathbf{W}_{51})] + \mathbf{H}\mathbf{B}_1 \right)$$

$$\mathbf{Y}_1 = \mathbf{F} \left(\begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \mathbf{X}_3 & \mathbf{X}_4 & \mathbf{X}_5 \end{bmatrix} * \begin{bmatrix} \mathbf{I}_{N1} & \mathbf{W}_{11} & \mathbf{W}_{12} & \mathbf{W}_{13} \\ \mathbf{I}_{N2} & \mathbf{W}_{21} & \mathbf{W}_{22} & \mathbf{W}_{23} \\ \mathbf{I}_{N3} & \mathbf{W}_{31} & \mathbf{W}_{32} & \mathbf{W}_{33} \\ \mathbf{I}_{N4} & \mathbf{W}_{41} & \mathbf{W}_{42} & \mathbf{W}_{43} \\ \mathbf{I}_{N5} & \mathbf{W}_{51} & \mathbf{W}_{52} & \mathbf{W}_{53} \\ \mathbf{I}_{N6} & \mathbf{W}_{61} & \mathbf{W}_{62} & \mathbf{W}_{63} \end{bmatrix} + \begin{bmatrix} \mathbf{HB}_1 \\ \mathbf{HB}_2 \\ \mathbf{HB}_3 \end{bmatrix} \right)$$

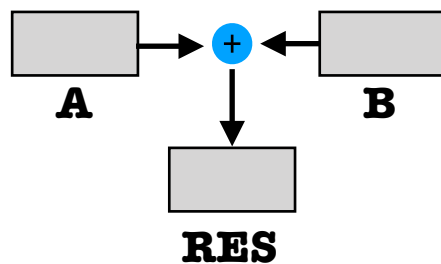
How to code in Tensorflow?

1 Define graph

Let's say you want to add two numbers. Define the placeholder for two numbers. Also create a placeholder to store the result.



Define the operation and store the operation result in result placeholder



2 Execute the graph

Create tensor flow session and initialise the graph

```
With tf.Session() as sess:  
    sess.run(tf.global_variables_initializer())
```

Create a dictionary using name of the placeholders and values to be assigned to each placeholder

```
feed_dict = {  
    A : 1,  
    B : 4  
}
```

Pass the values to the graph placeholders and compute the result

```
sess.run(RES, feed_dict)
```


- 1 Artificial Neural Networks**
- 2 Components of neural network**
- 3 Feed forward in neural network**
- 4 Representing NN with matrix**
- 5 Numpy and tensor flow implementation of Feed Forward NN**

