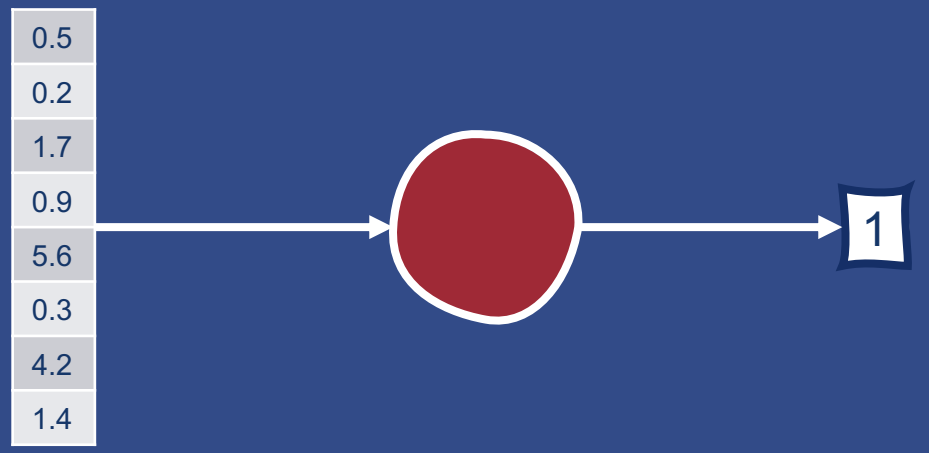
Building Blocks for Neural Networks

Natalie Parde
UIC CS 421

Building Blocks for Neural Networks

- At their core, neural networks are comprised of **computational units**
- Computational units:
  1. Take a set of real-valued numbers as input
  2. Perform some computation on them
  3. Produce a single output



Computational Units

- The computation performed by each unit is a weighted sum of inputs
  - Assign a weight to each input
  - Add one additional bias term

- More formally, given a set of inputs $x_1, \ldots, x_n$, a unit has a set of corresponding weights $w_1, \ldots, w_n$ and a bias b, so the weighted sum z can be represented as:
  - $z = b + \sum_i w_i x_i$

Sound familiar?

- This is exactly the same sort of weighted sum of inputs that we needed to find with logistic regression!
- Recall that we can also represent the weighted sum z using vector notation:
  - $z = w \cdot x + b$

Computational Units

- The weighted sum of inputs computes a **linear function** of x
- As we already saw, neural networks learn **nonlinear functions**
- These nonlinear functions are commonly referred to as **activations**
- The output of a computation unit is thus the **activation value** for the unit, y
  - $y = f(z) = f(w \cdot x + b)$

There are many different activation functions!

exponential linear unit (elu)

softmax

scaled exponential linear unit (selu)

softplus

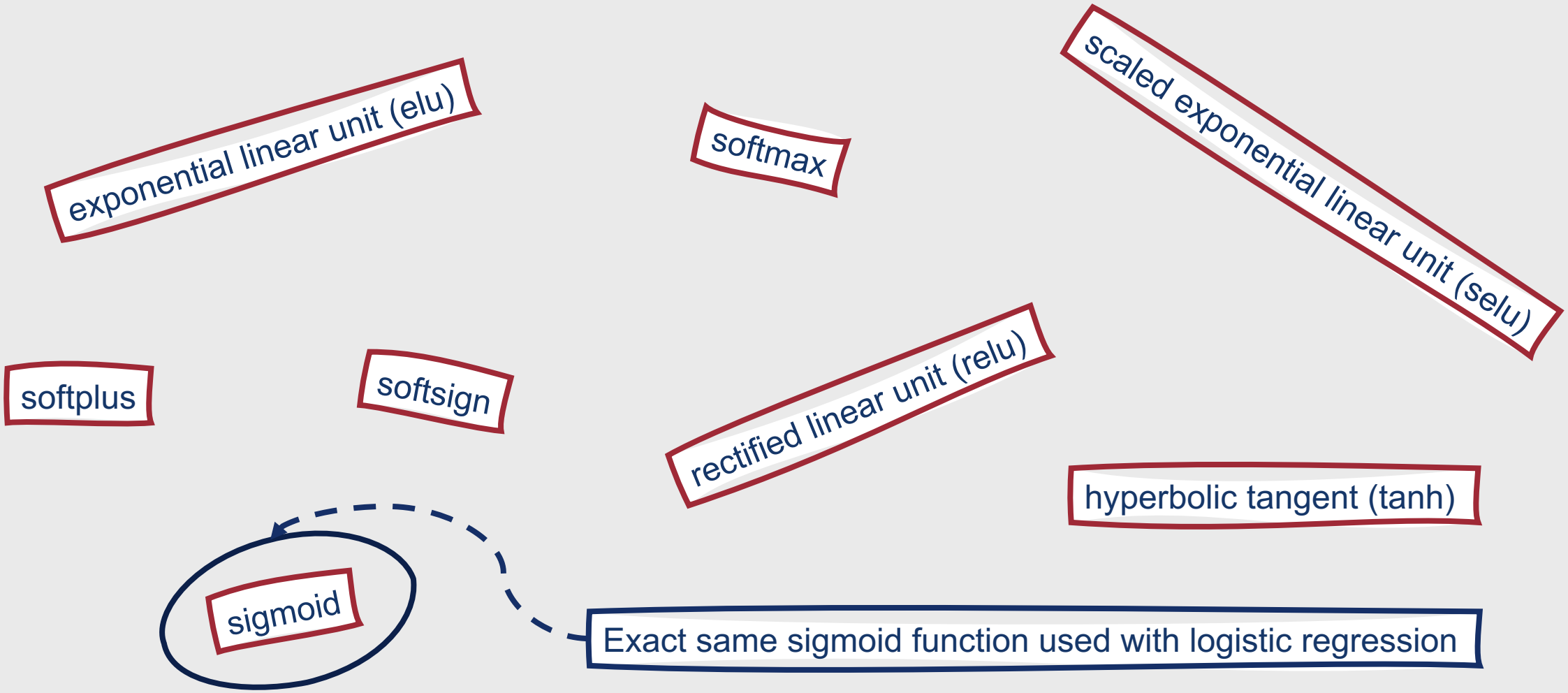
softsign

rectified linear unit (relu)

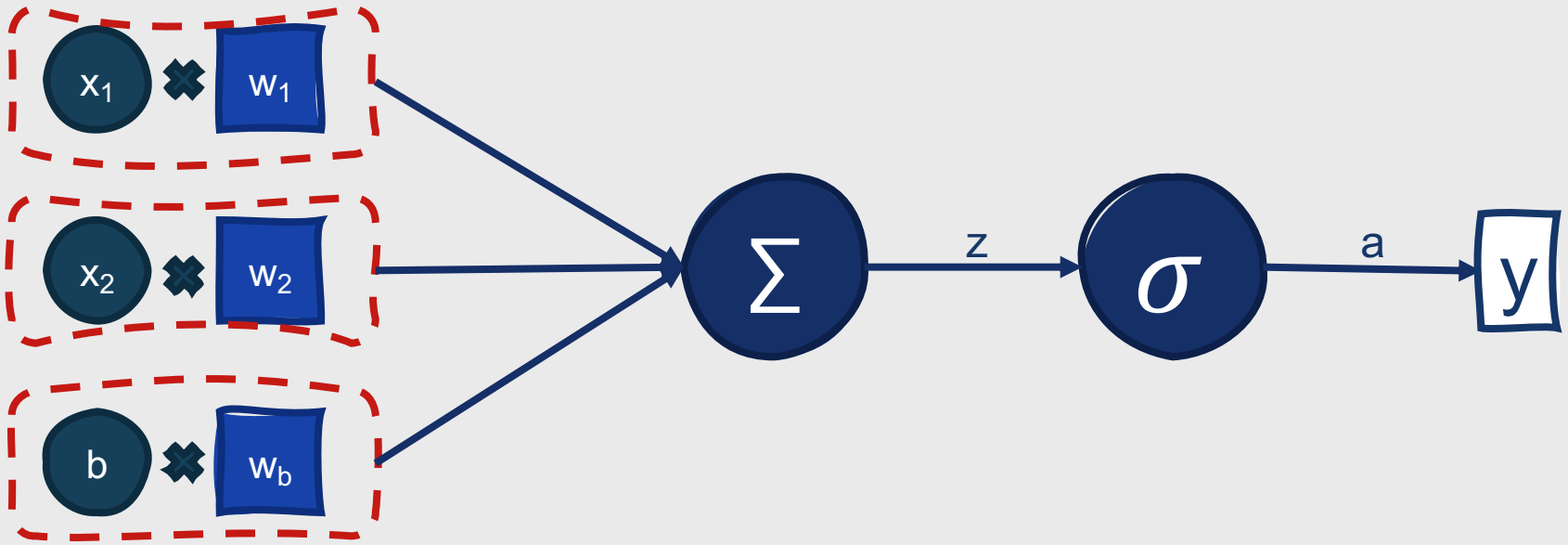
hyperbolic tangent (tanh)

sigmoid

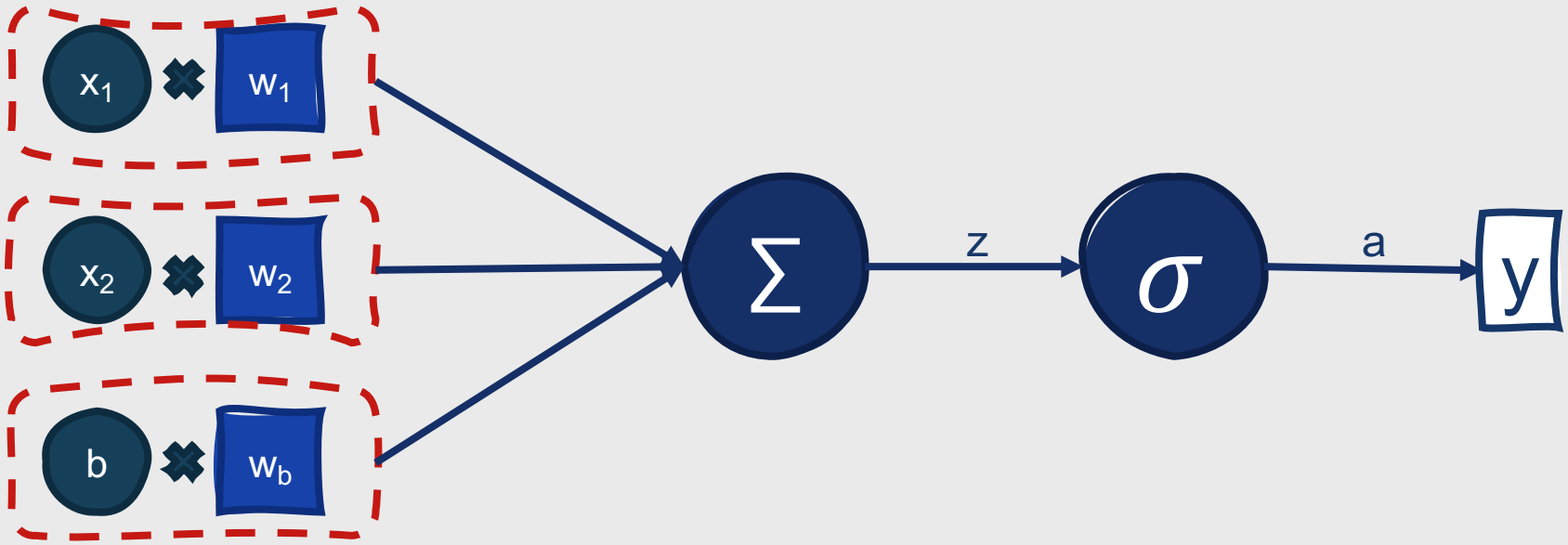
There are many different activation functions!



Computational Unit with Sigmoid Activation



Example: Computational Unit with Sigmoid Activation

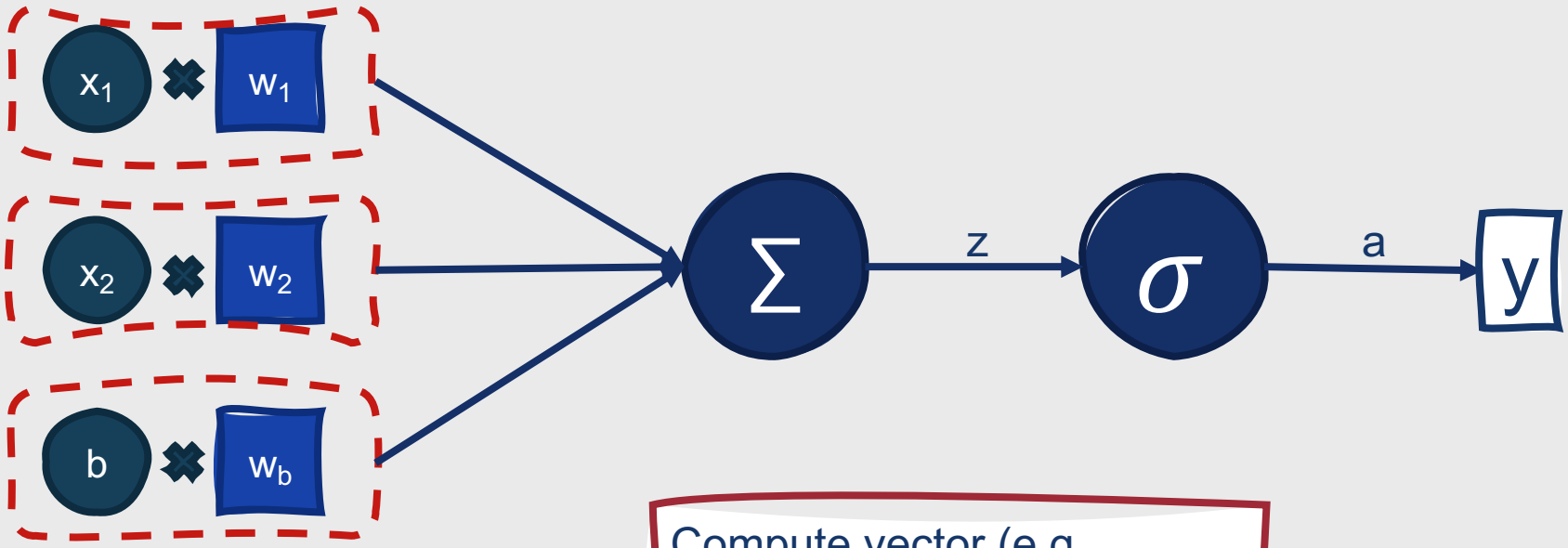


Input: “beautiful brutalist architecture”

Weights (Input): [0.2, 0.3]
Weight (Bias): [0.5]

Bias: 1.0

Example: Computational Unit with Sigmoid Activation



Input: “beautiful brutalist architecture”

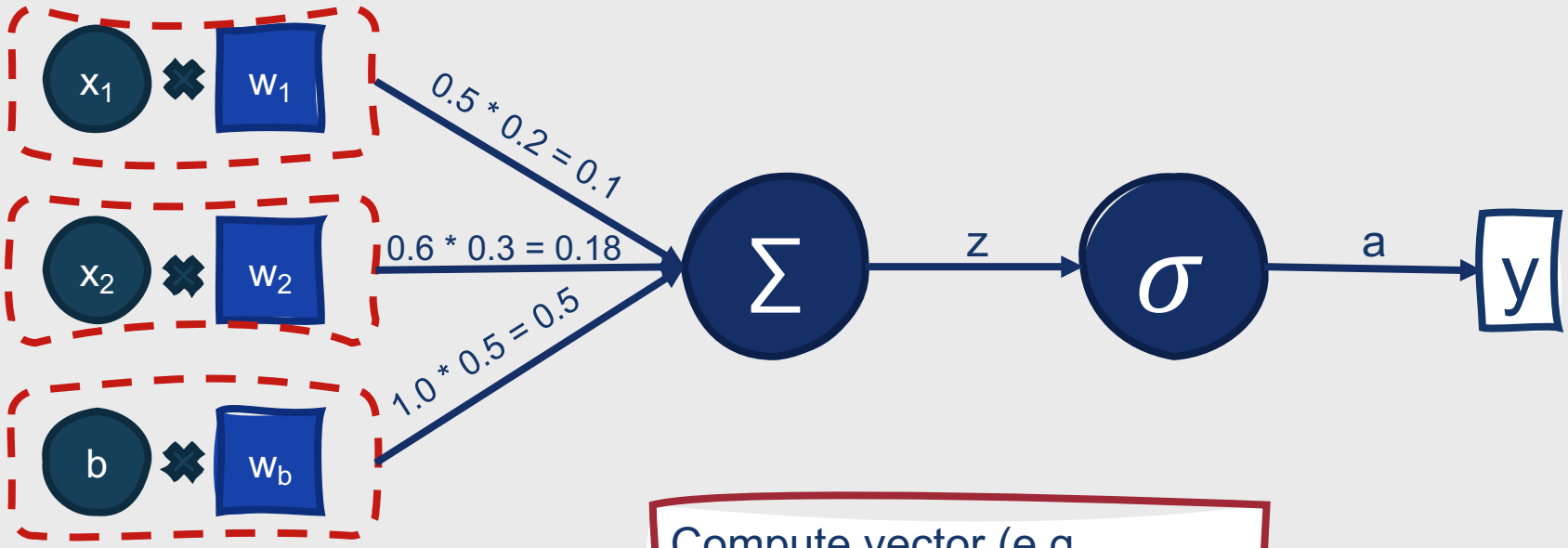
Compute vector (e.g., averaged Word2Vec embeddings for “beautiful,” “brutalist,” and “architecture”)

[0.5, 0.6]

Weights (Input): [0.2, 0.3]
Weight (Bias): [0.5]

Bias: 1.0

Example: Computational Unit with Sigmoid Activation



Input: “beautiful brutalist architecture”

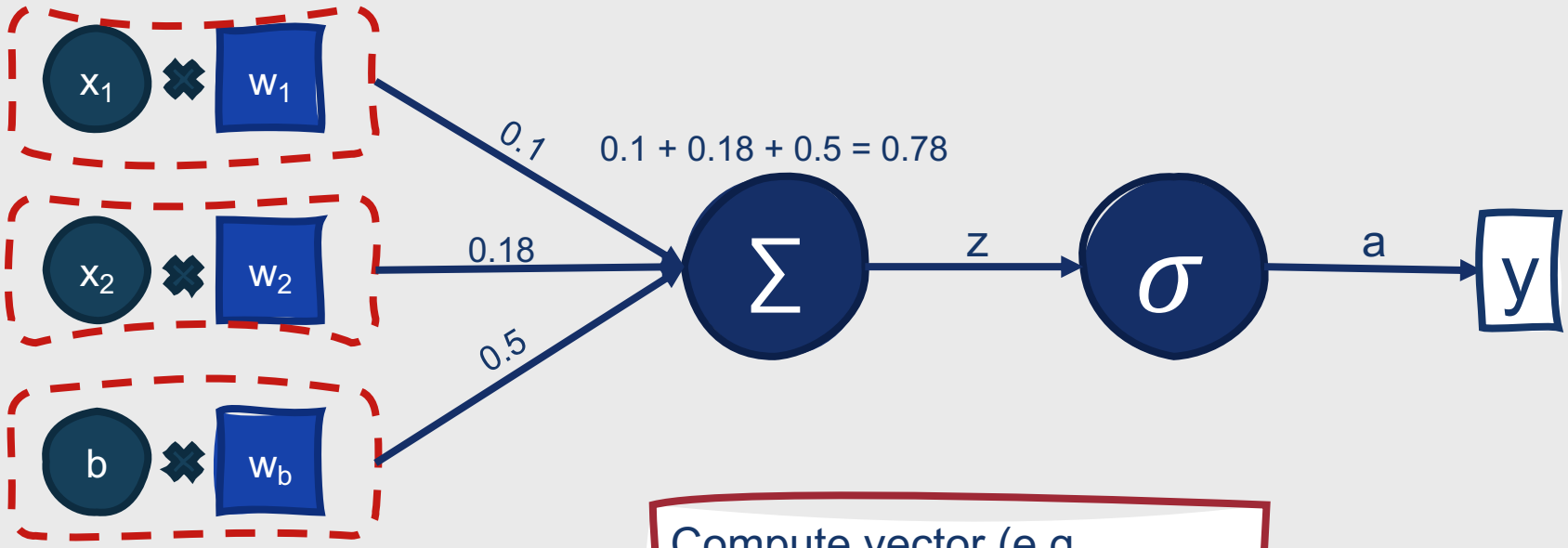
Compute vector (e.g., averaged Word2Vec embeddings for “beautiful,” “brutalist,” and “architecture”)

[0.5, 0.6]

Weights (Input): [0.2, 0.3]
Weight (Bias): [0.5]

Bias: 1.0

Example: Computational Unit with Sigmoid Activation



Input: “beautiful brutalist architecture”

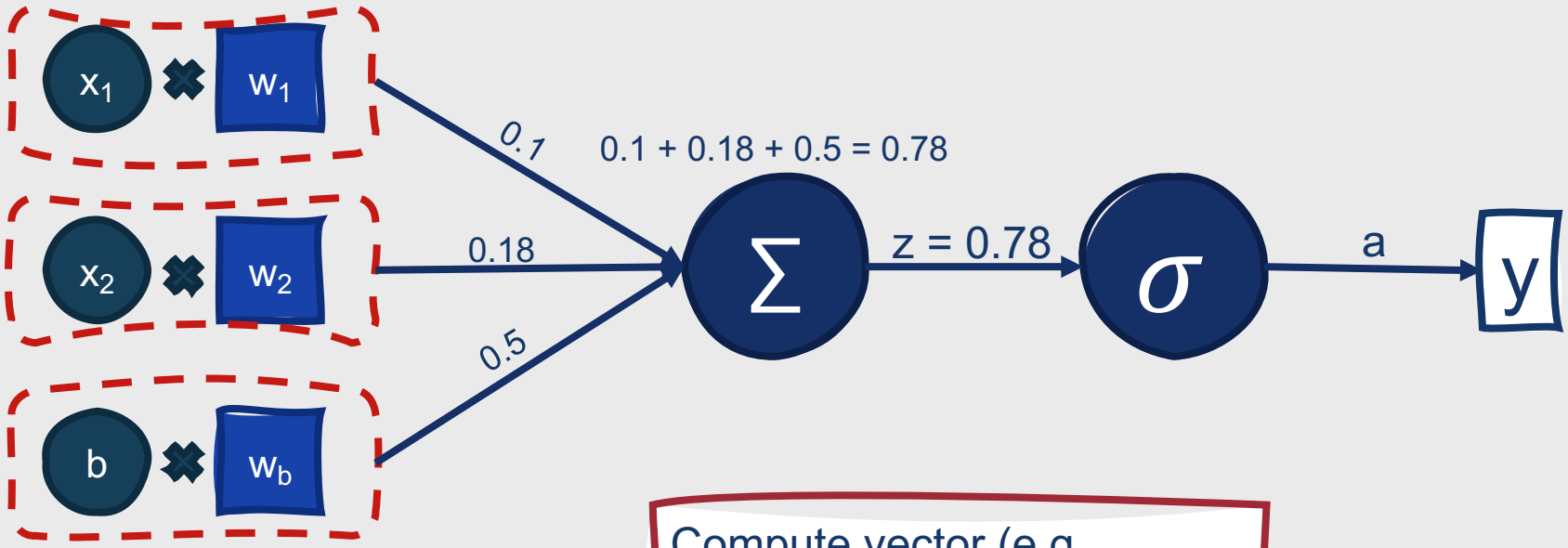
Compute vector (e.g., averaged Word2Vec embeddings for “beautiful,” “brutalist,” and “architecture”)

[0.5, 0.6]

Weights (Input): [0.2, 0.3]
Weight (Bias): [0.5]

Bias: 1.0

Example: Computational Unit with Sigmoid Activation



Input: “beautiful brutalist architecture”

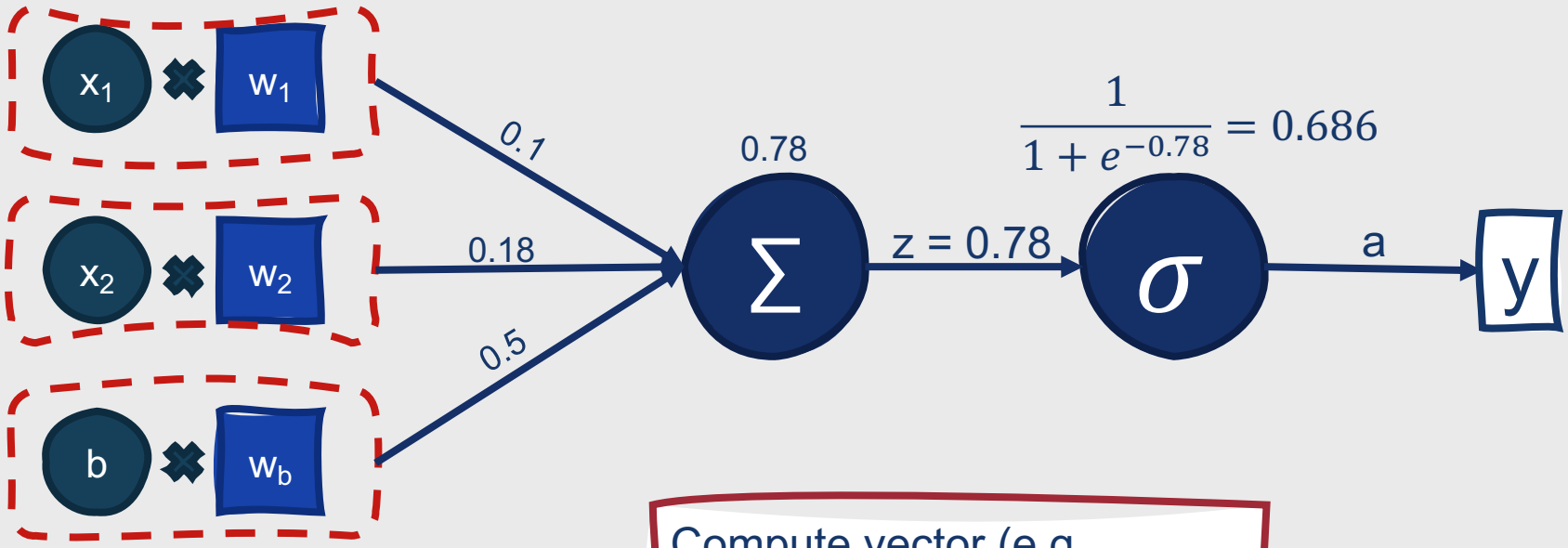
Compute vector (e.g., averaged Word2Vec embeddings for “beautiful,” “brutalist,” and “architecture”)

[0.5, 0.6]

Weights (Input): [0.2, 0.3]
Weight (Bias): [0.5]

Bias: 1.0

Example: Computational Unit with Sigmoid Activation



Input: “beautiful brutalist architecture”

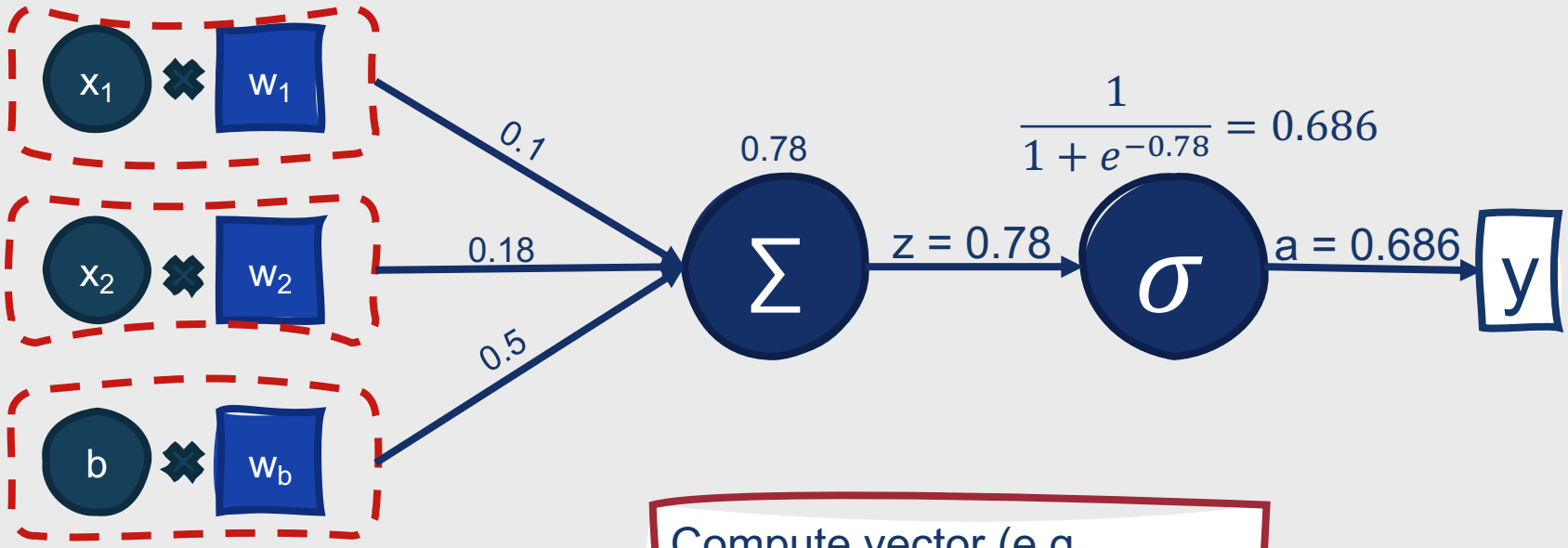
Compute vector (e.g., averaged Word2Vec embeddings for “beautiful,” “brutalist,” and “architecture”)

[0.5, 0.6]

Weights (Input): [0.2, 0.3]
Weight (Bias): [0.5]

Bias: 1.0

Example: Computational Unit with Sigmoid Activation



Input: “beautiful brutalist architecture”

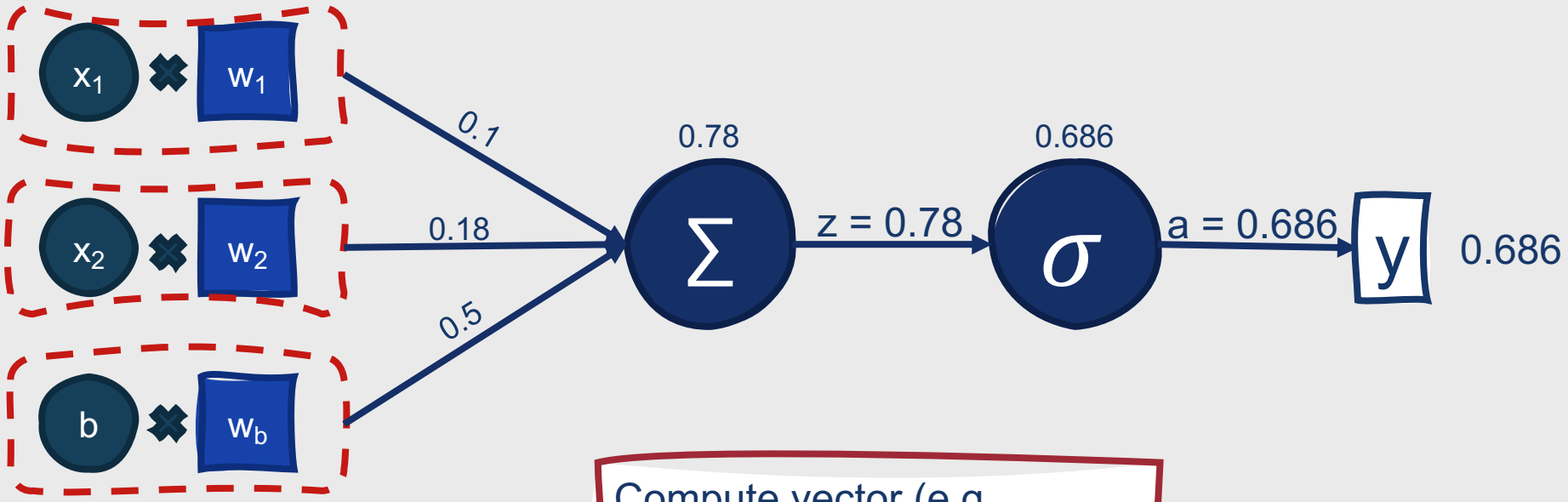
Compute vector (e.g., averaged Word2Vec embeddings for “beautiful,” “brutalist,” and “architecture”)

[0.5, 0.6]

Weights (Input): [0.2, 0.3]
Weight (Bias): [0.5]

Bias: 1.0

Example: Computational Unit with Sigmoid Activation



Input: “beautiful brutalist architecture”

Compute vector (e.g., averaged Word2Vec embeddings for “beautiful,” “brutalist,” and “architecture”)

[0.5, 0.6]

Weights (Input): [0.2, 0.3]
Weight (Bias): [0.5]

Bias: 1.0