

Statistics and Machine Learning

**Week 3 – Linear regression
02/01 – 02/05**

Contents of Week 3

Contents of Week 3

- Three elements in ML: Data set, model, loss function

Contents of Week 3

- Three elements in ML: Data set, model, loss function
- Linear regression example: advertising.csv data set

Contents of Week 3

- Three elements in ML: Data set, model, loss function
- Linear regression example: advertising.csv data set
- Training error, test error, optimal model parameters

Contents of Week 3

- Three elements in ML: Data set, model, loss function
- Linear regression example: advertising.csv data set
- Training error, test error, optimal model parameters
- Lab session: Pandas, sklearn and matplotlib

Contents of Week 3

- Three elements in ML: Data set, model, loss function
- Linear regression example: advertising.csv data set
- Training error, test error, optimal model parameters
- Lab session: Pandas, sklearn and matplotlib
- Homework: Frog Jump

Data set

Model

Loss function

Training

Deploying

Data set

Training

00000000000000000
11111111111111111
22222222222222222
33333333333333333
44444444444444444
55555555555555555
66666666666666666
77777777777777777
88888888888888888
99999999999999999

Model

Deploying

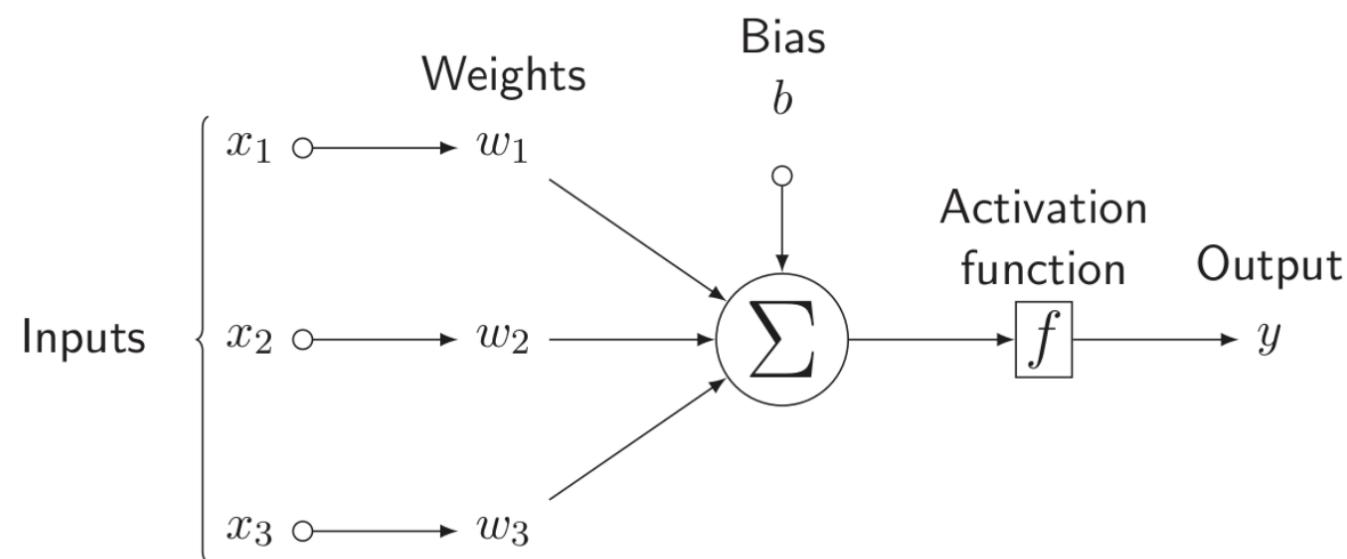
Loss function

Data set

Training

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model



<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

Loss function

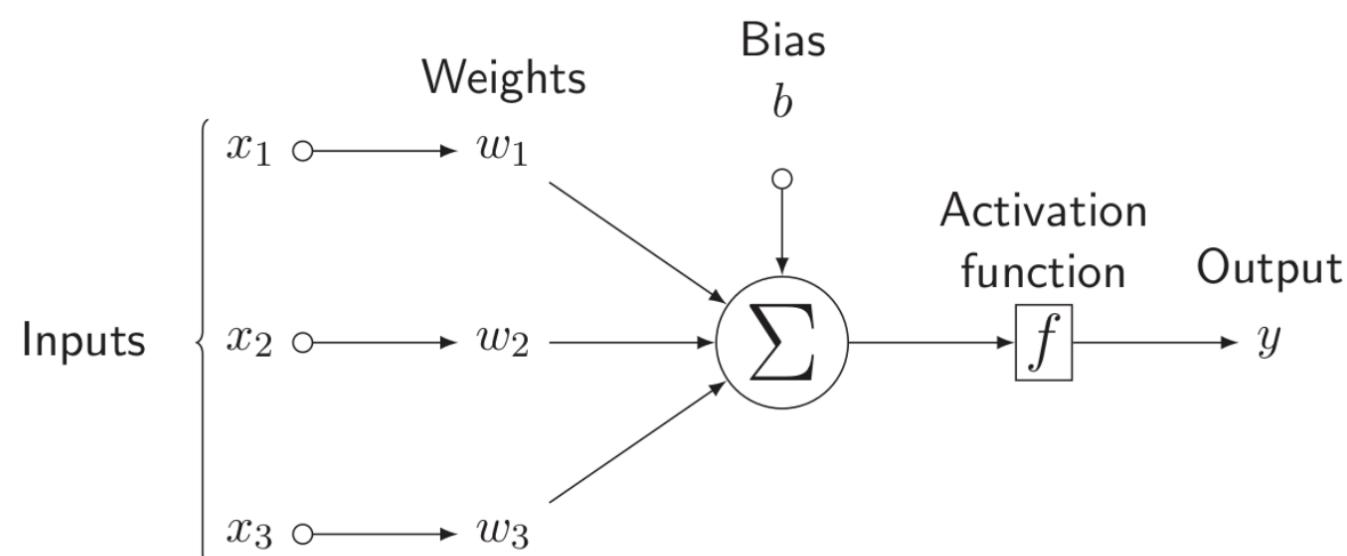
Deploying

Training

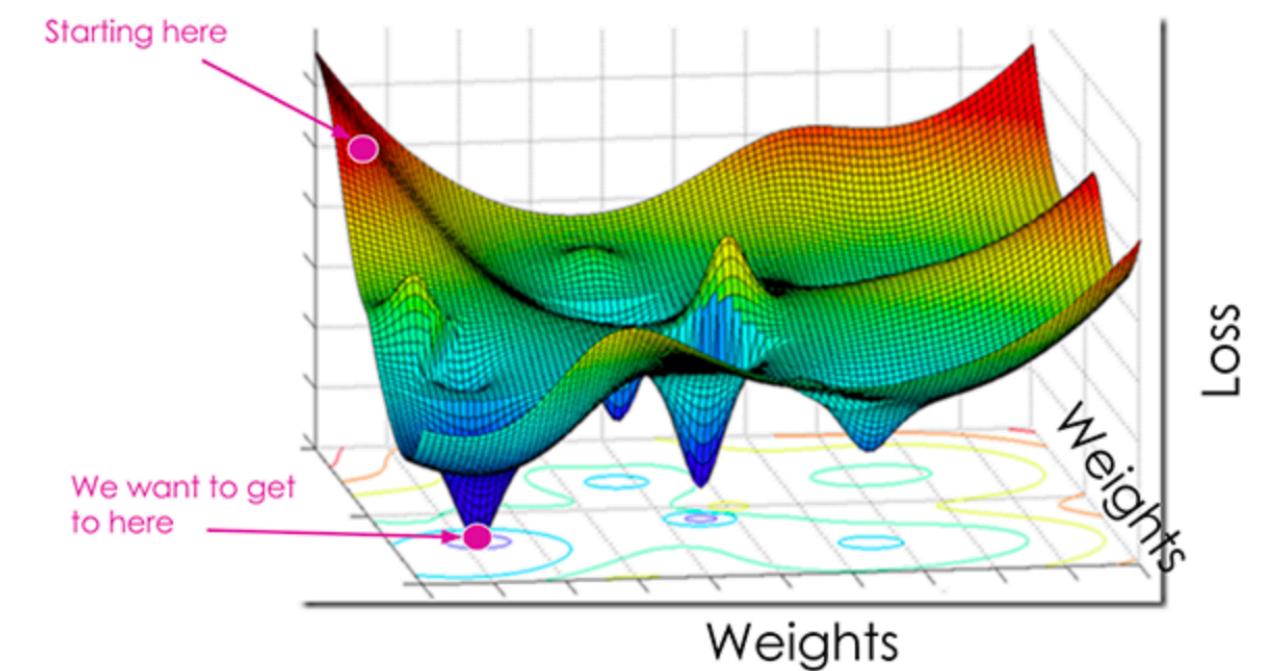
Data set

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model



Loss function



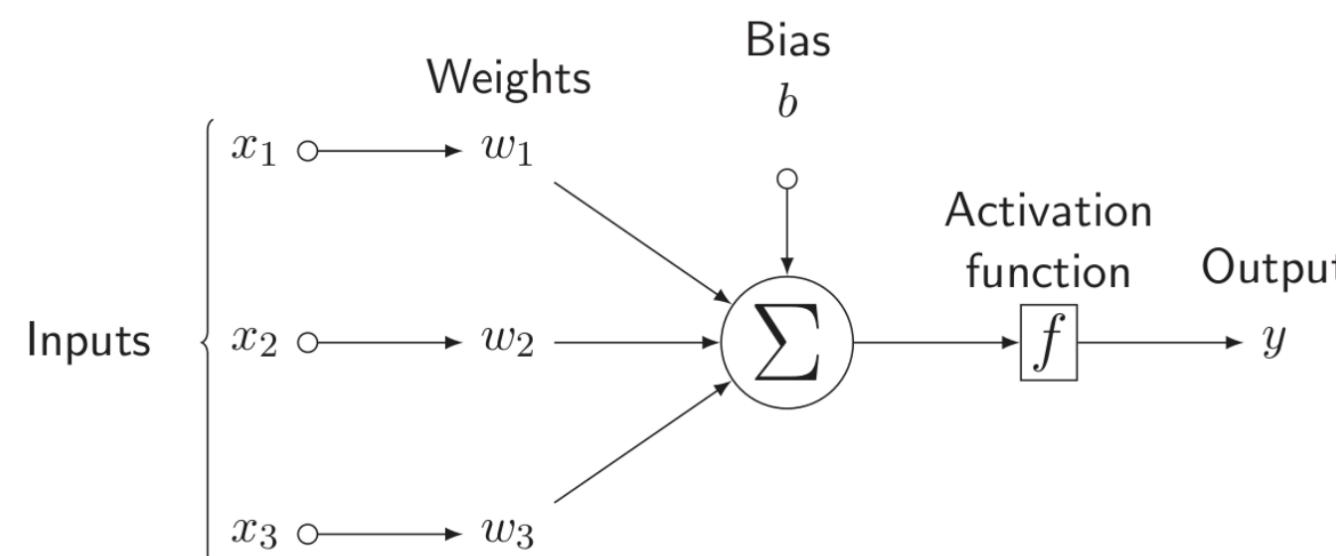
Deploying

Training

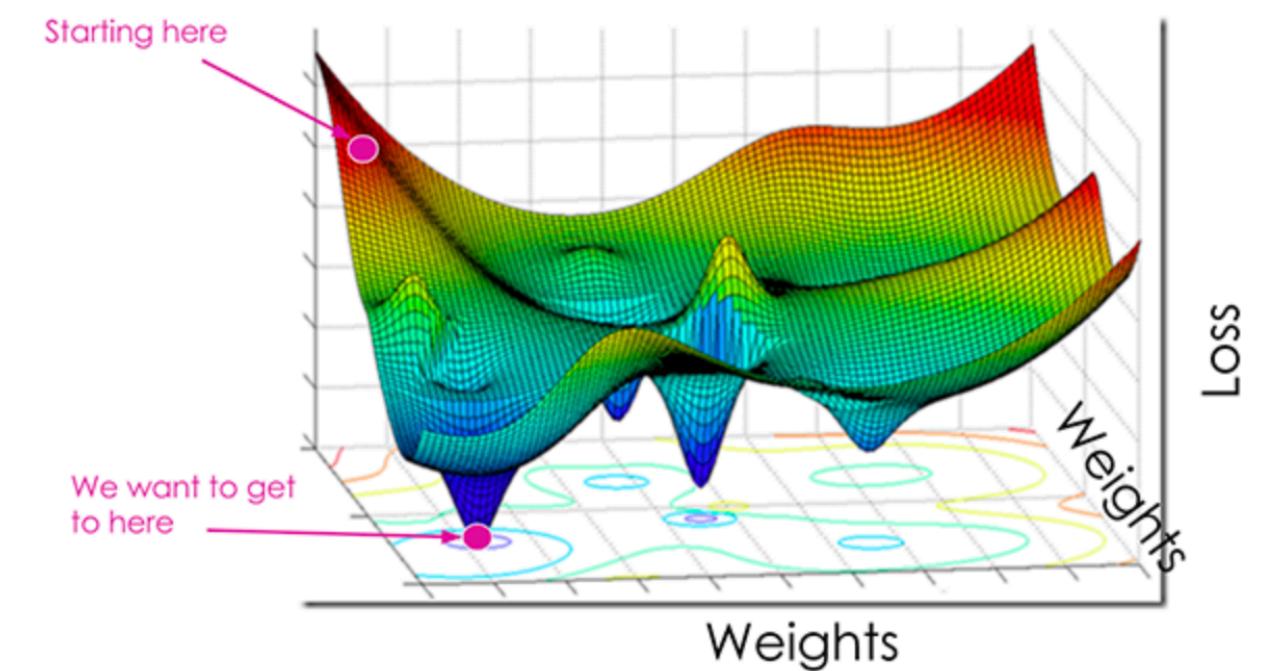
Data set

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model



Loss function



The optimal parameters: $\bar{w}_1, \bar{w}_2, \bar{w}_3, \dots, \bar{b}$

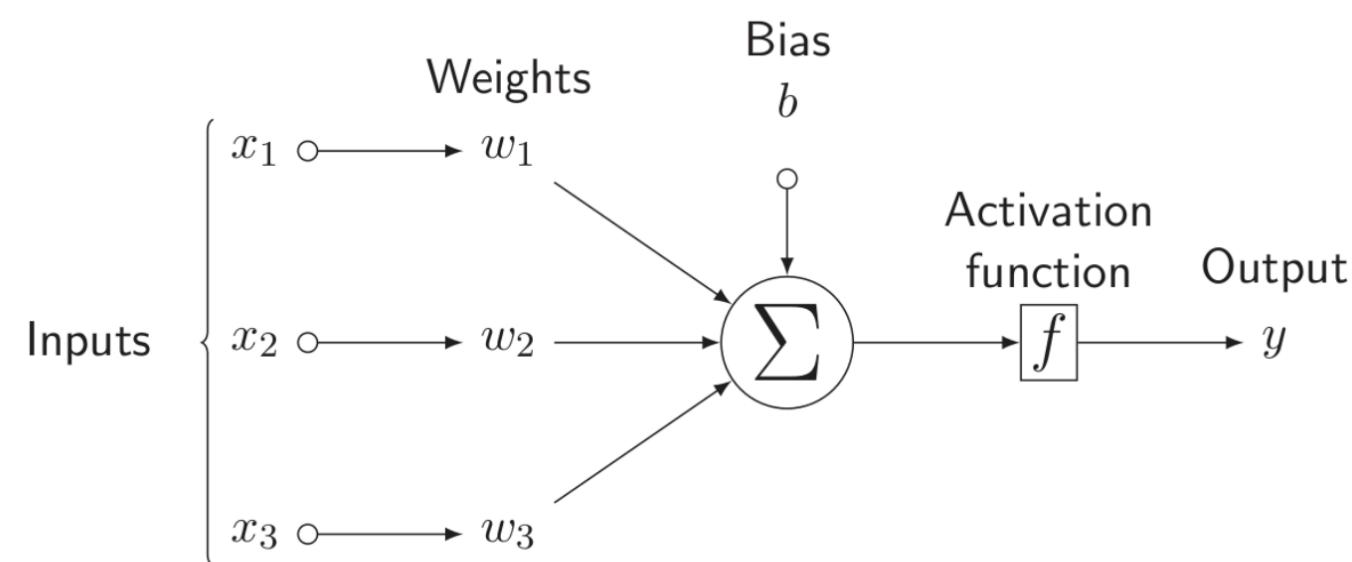
Deploying

Training

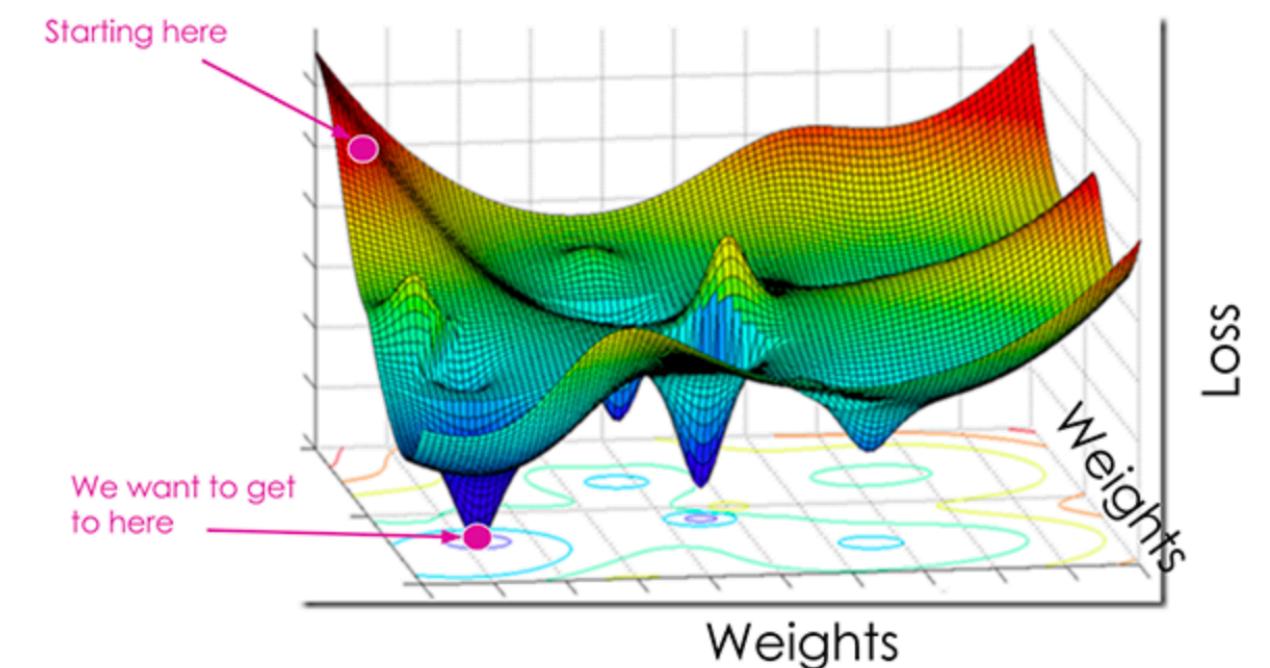
Data set

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model



Loss function



The optimal parameters: $\bar{w}_1, \bar{w}_2, \bar{w}_3, \dots, \bar{b}$

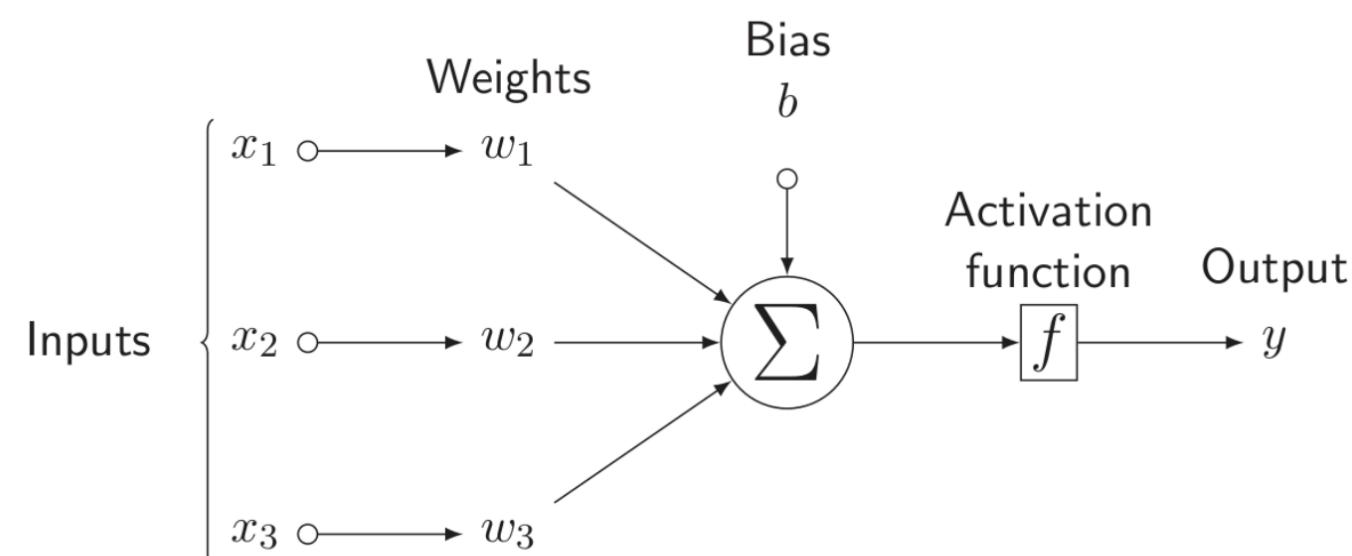
Deploying

Training

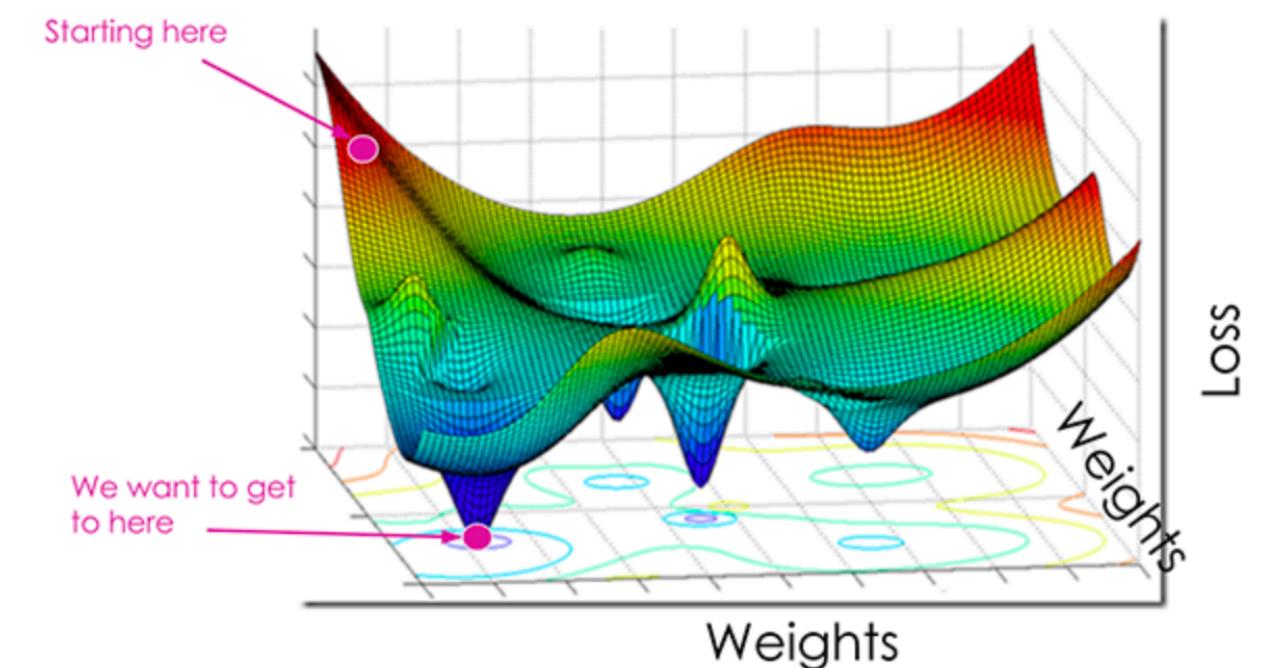
Data set

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model

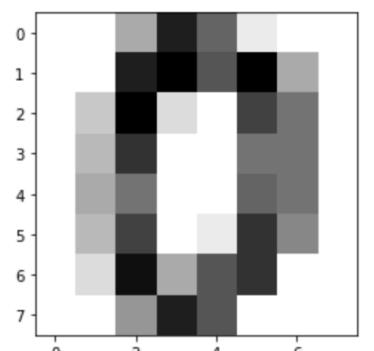


Loss function



The optimal parameters: $\bar{w}_1, \bar{w}_2, \bar{w}_3, \dots, \bar{b}$

Deploying

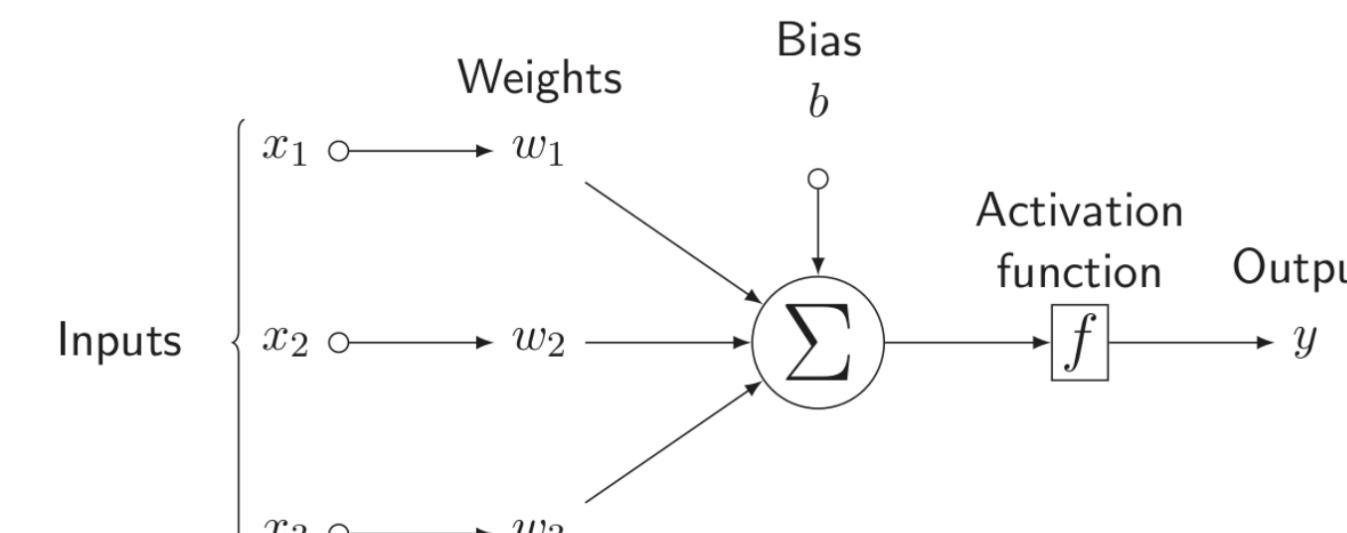


Training

Data set

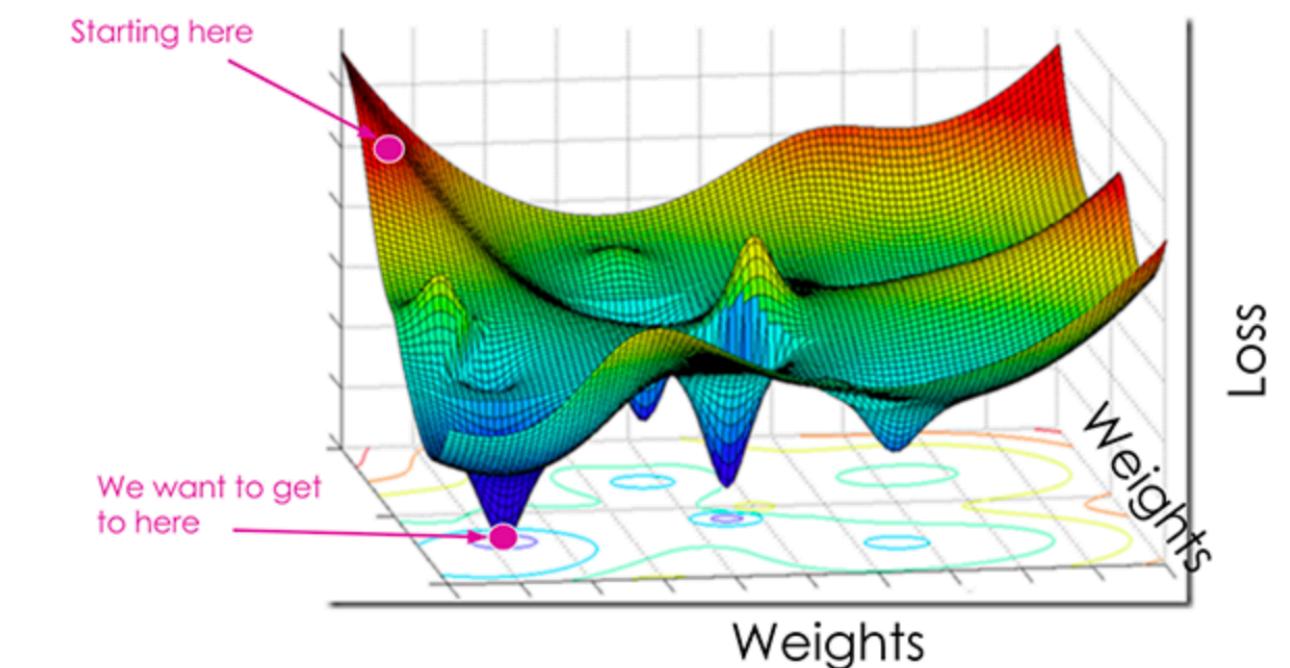
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model

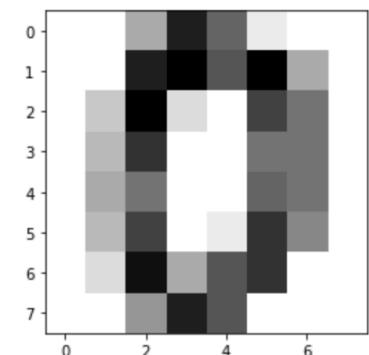


<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

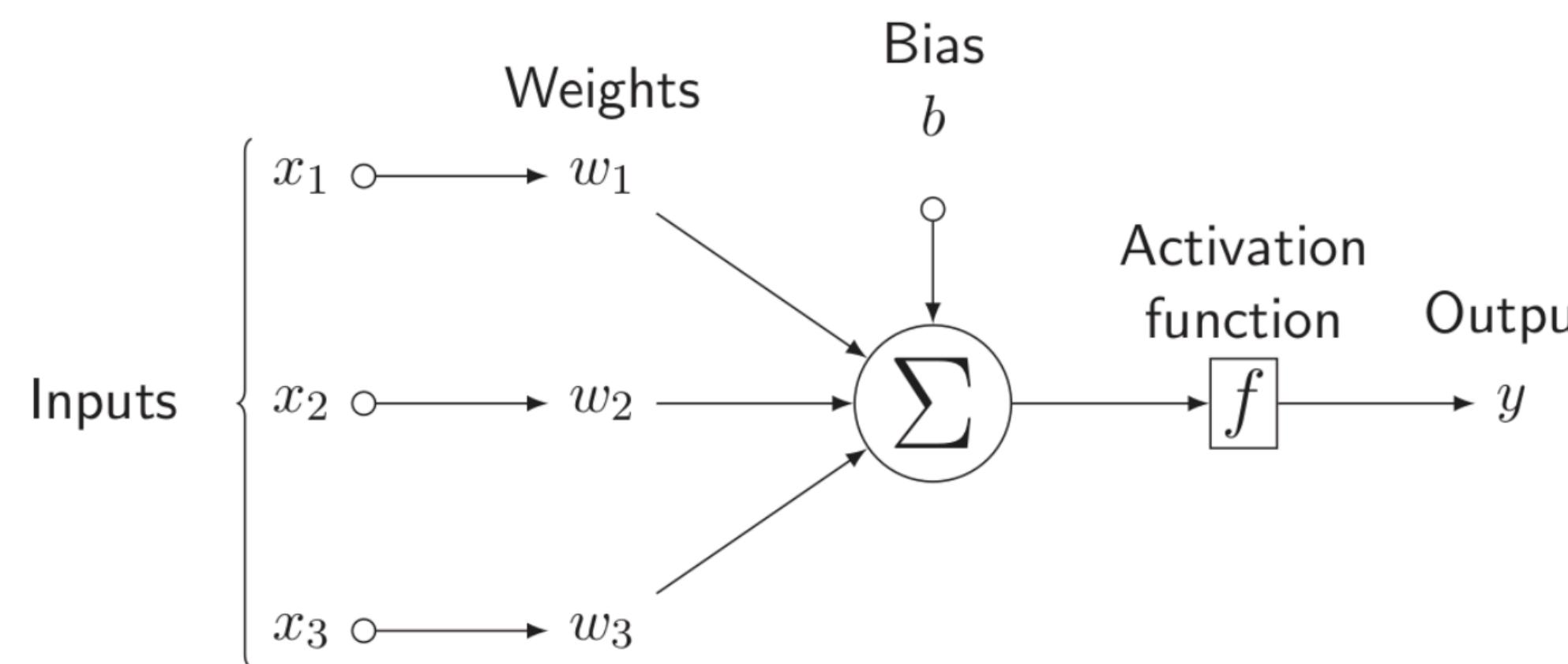
Loss function



Deploying



The optimal parameters: $\bar{w}_1, \bar{w}_2, \bar{w}_3, \dots, \bar{b}$



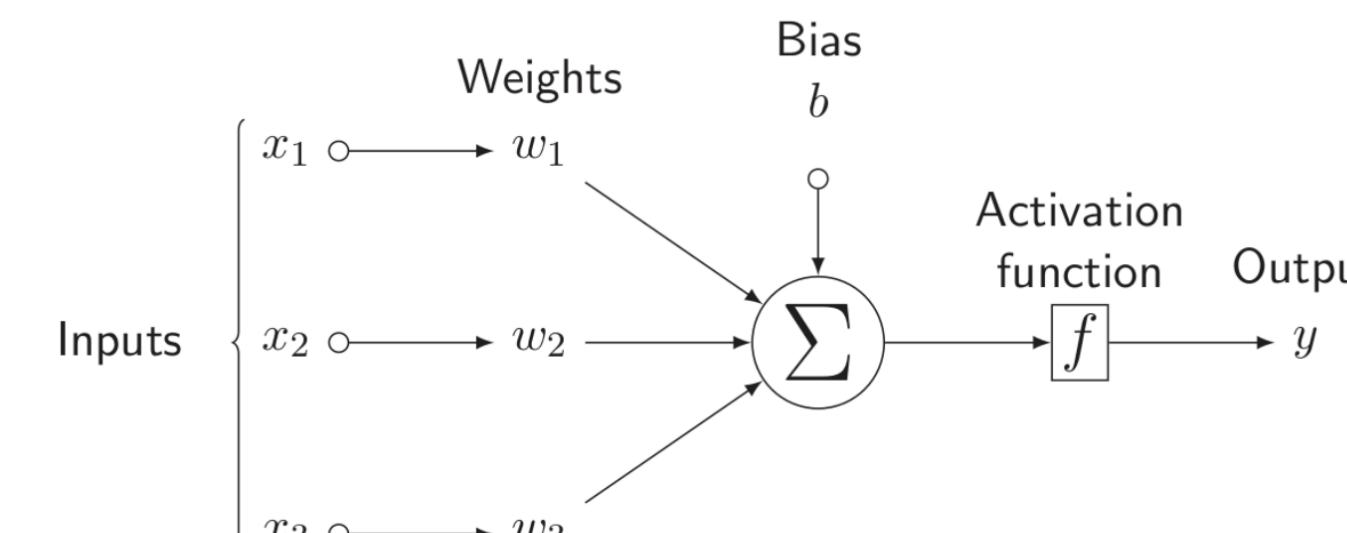
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

Training

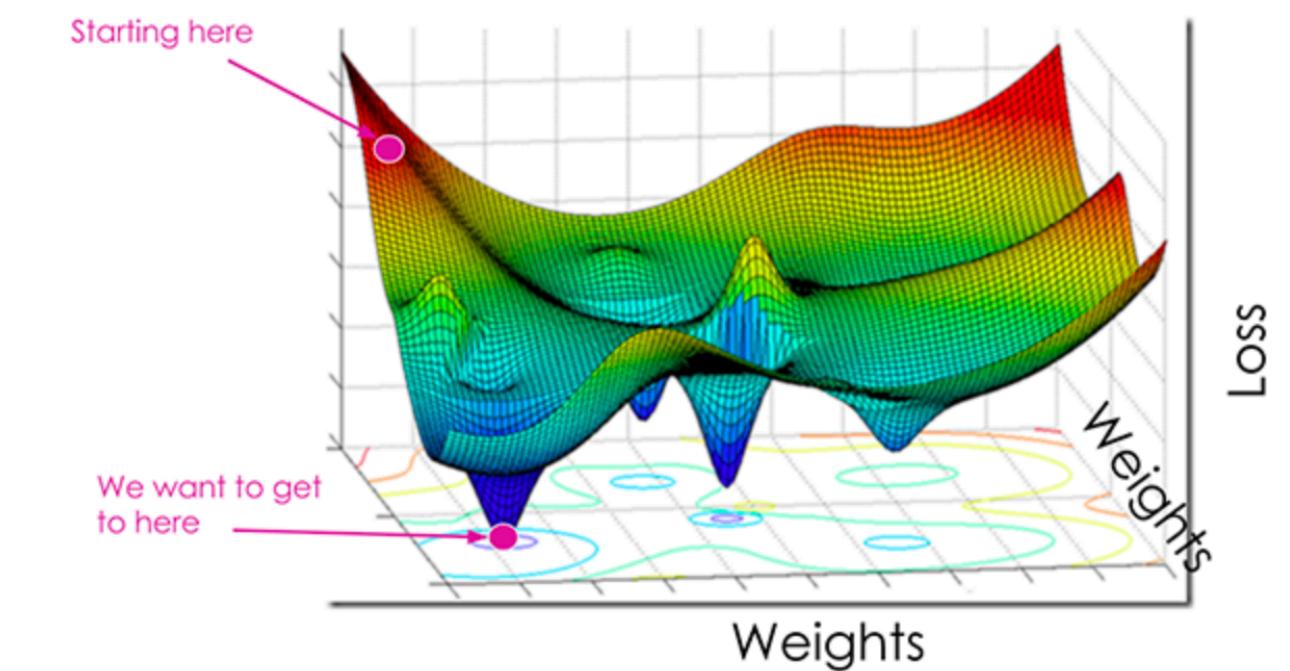
Data set

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model

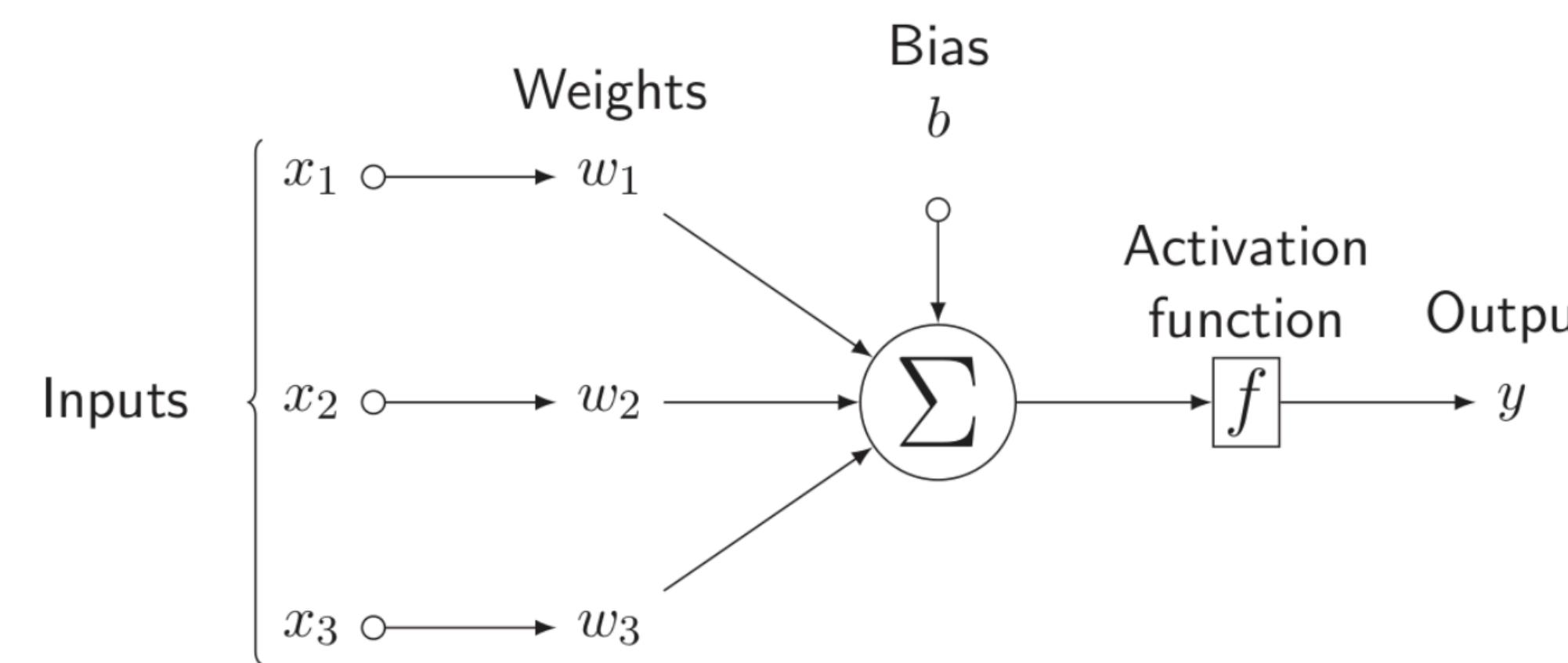
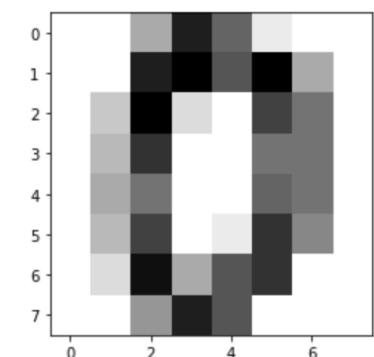


Loss function



Deploying

The optimal parameters: $\bar{w}_1, \bar{w}_2, \bar{w}_3, \dots, \bar{b}$

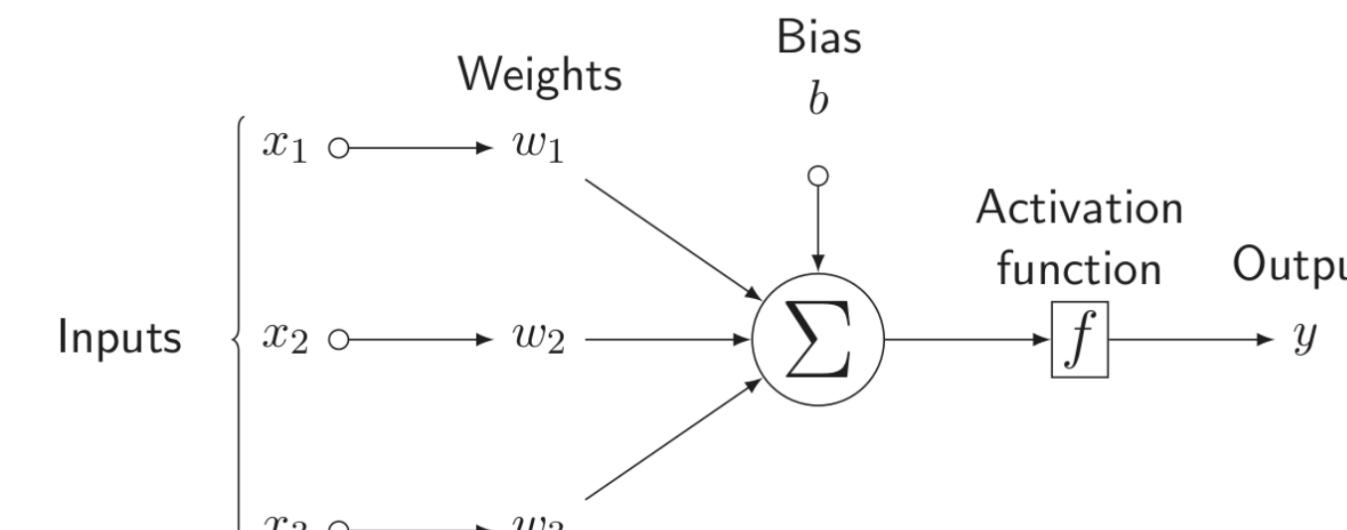


Data set

Training

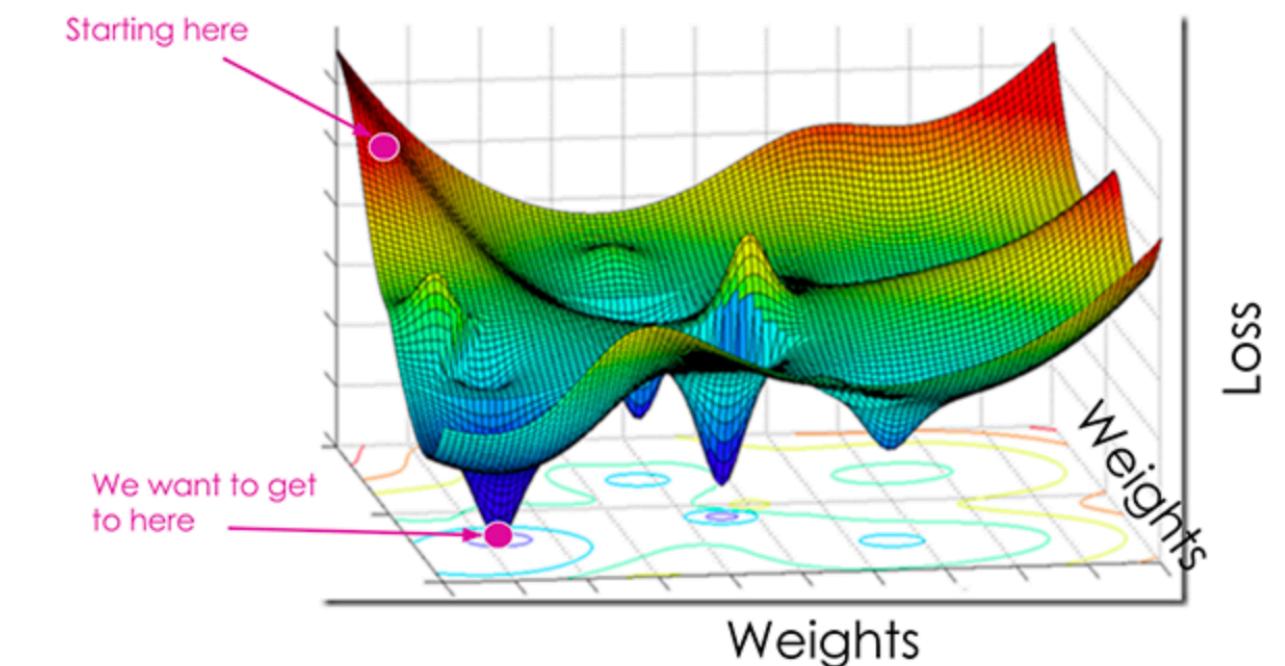
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model



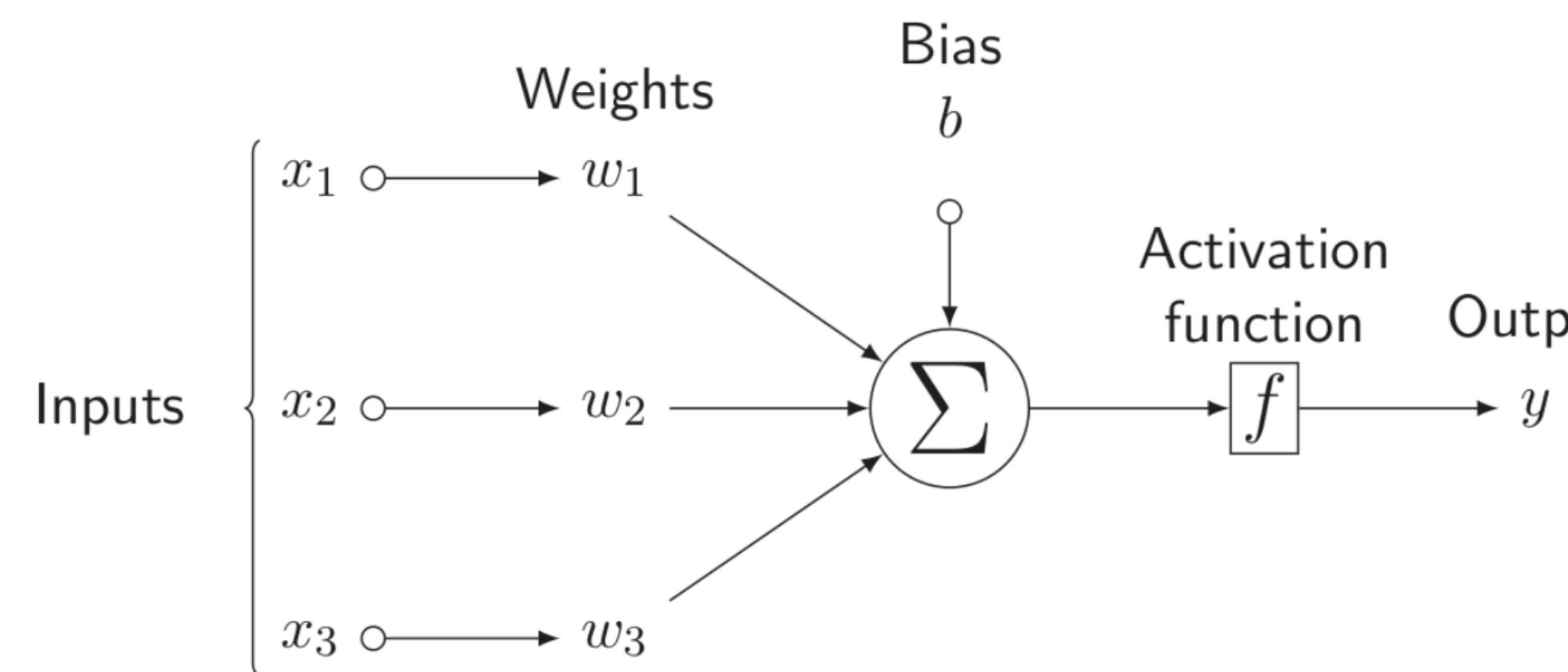
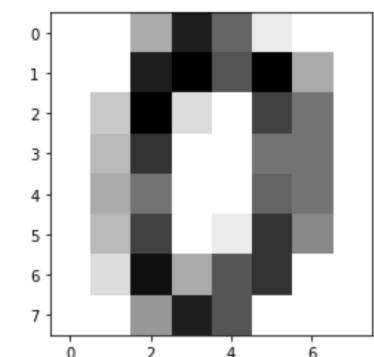
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

Loss function



Deploying

The optimal parameters: $\bar{w}_1, \bar{w}_2, \bar{w}_3, \dots, \bar{b}$



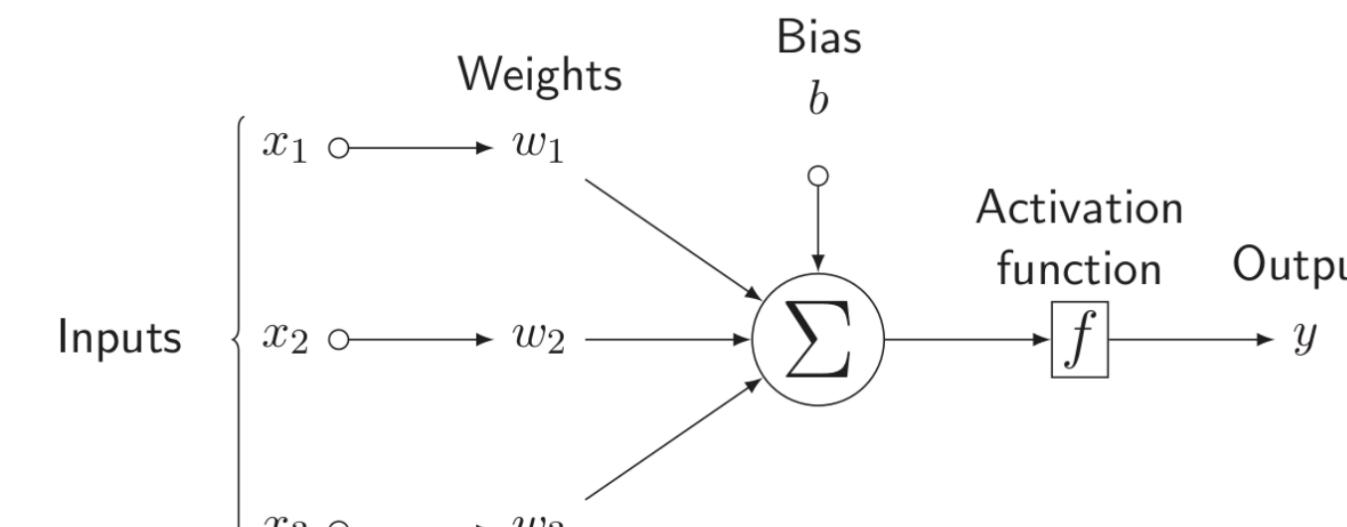
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

Data set

Training

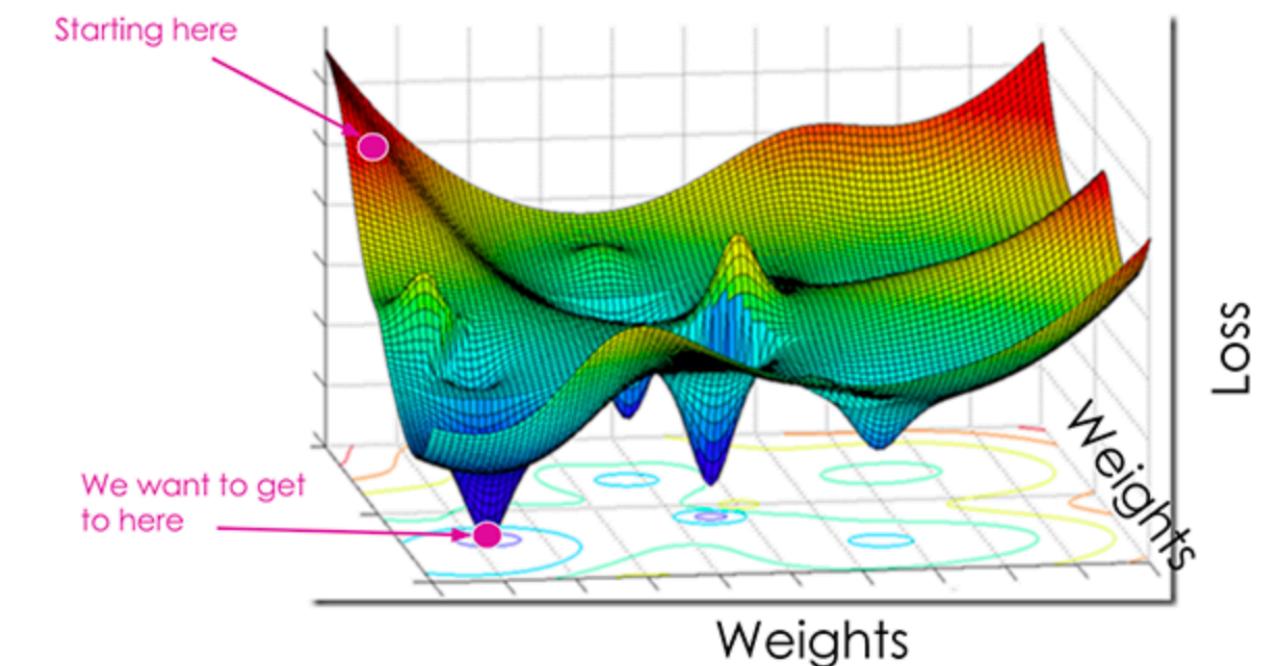
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9

Model



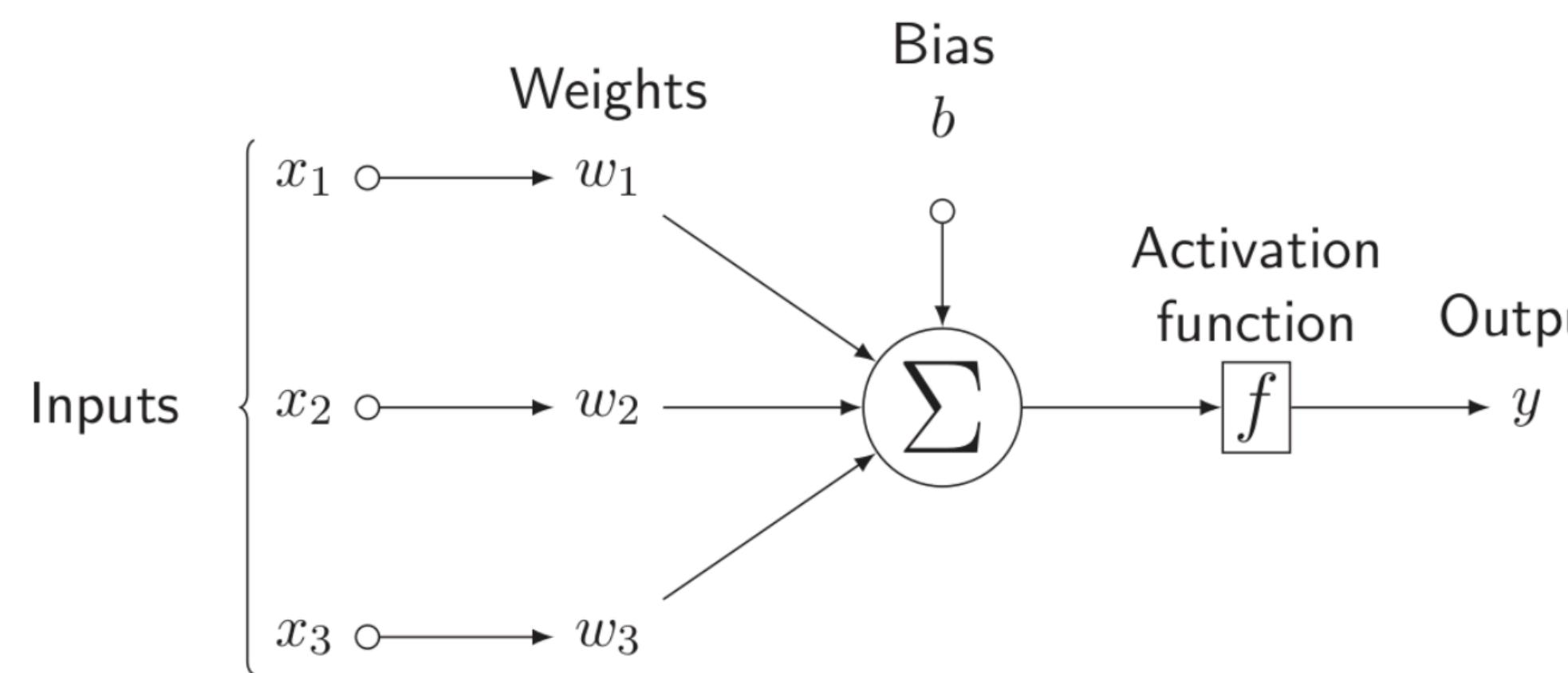
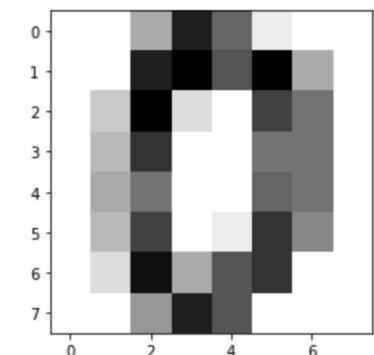
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

Loss function



The optimal parameters: $\bar{w}_1, \bar{w}_2, \bar{w}_3, \dots, \bar{b}$

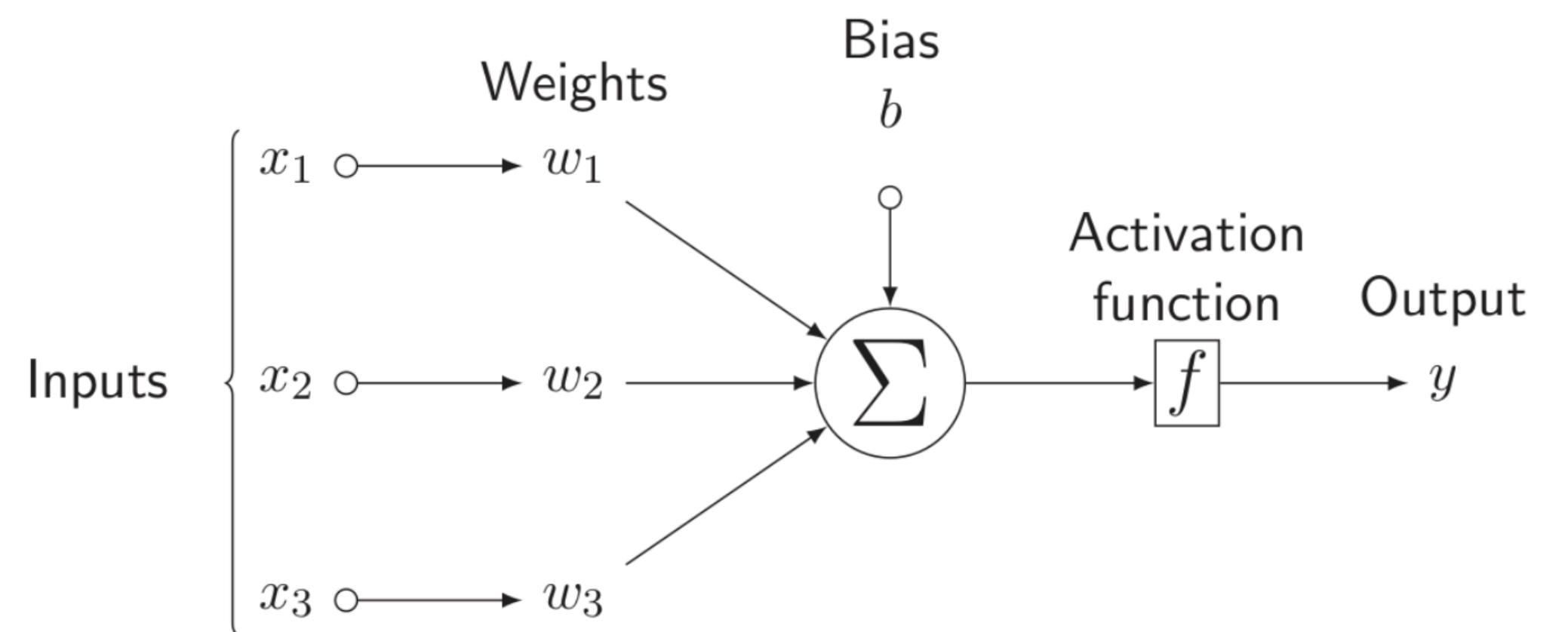
Deploying



<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

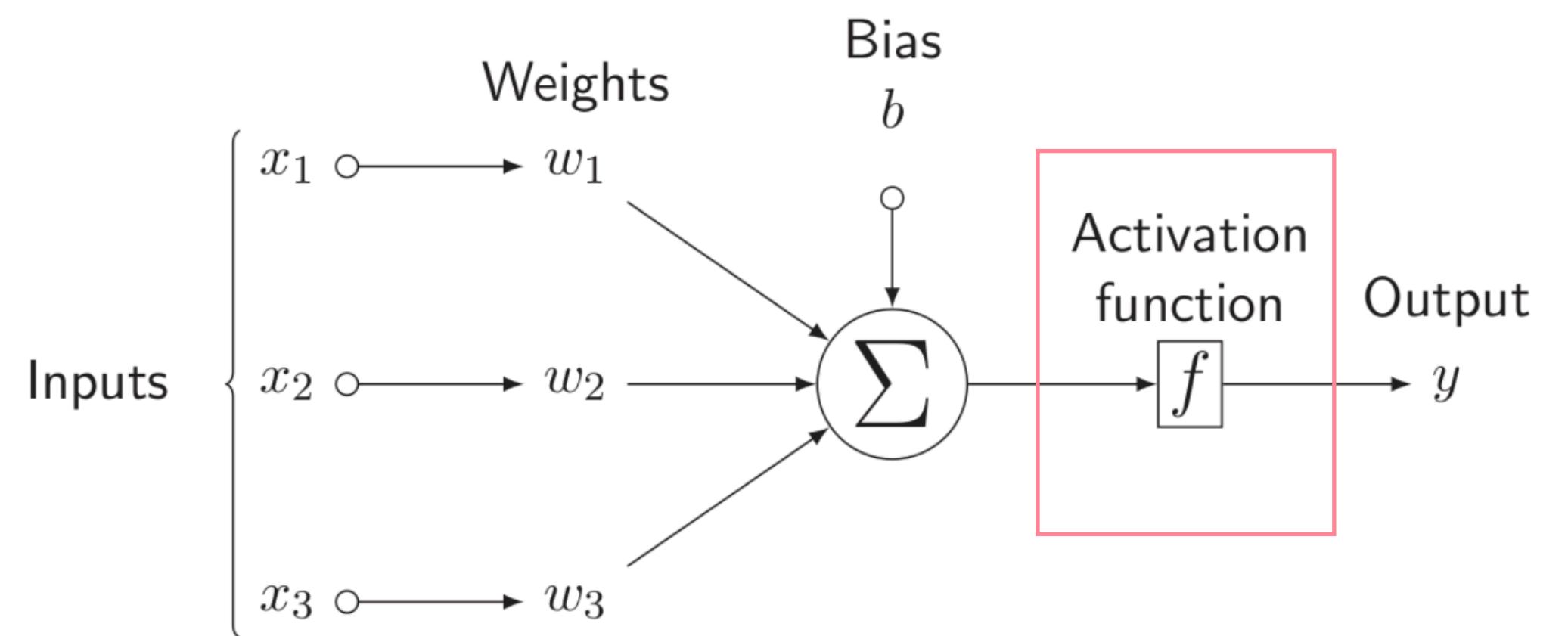
It is a zero

The classification is a bit advanced
because the model is complex



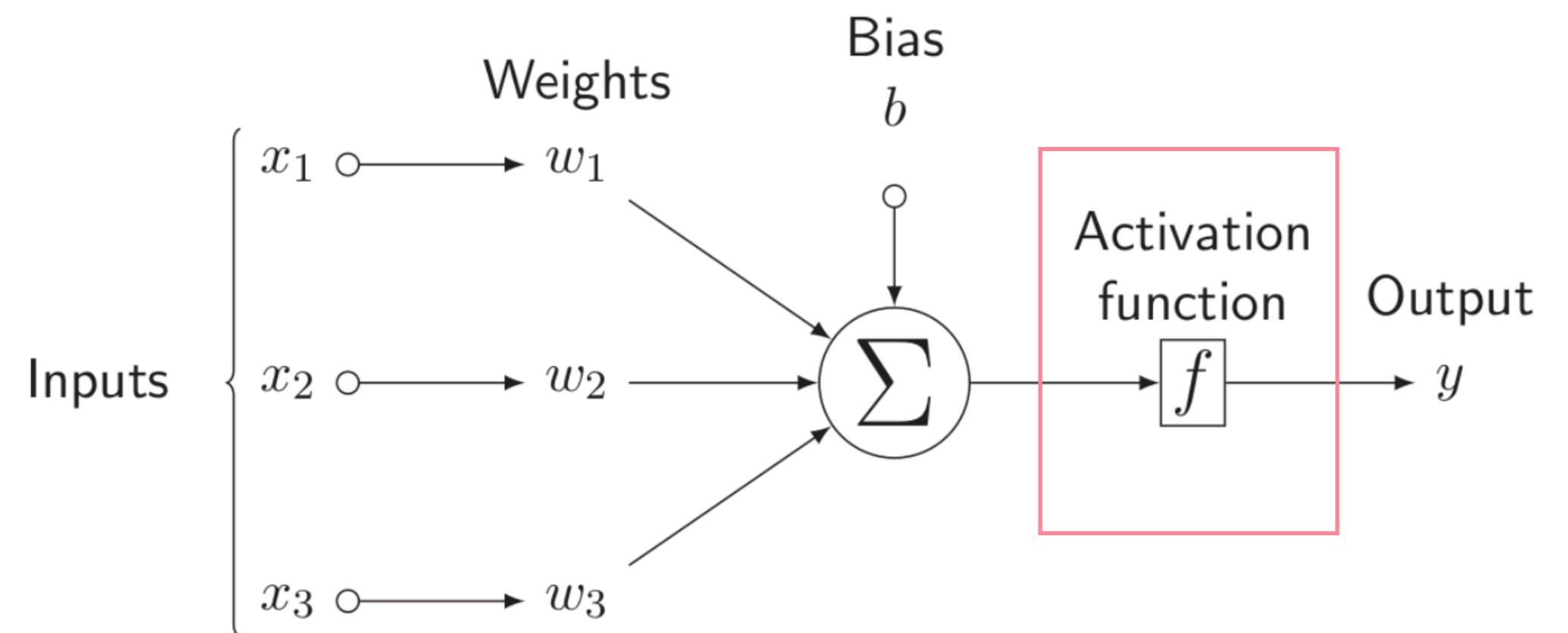
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

The classification is a bit advanced
because the model is complex

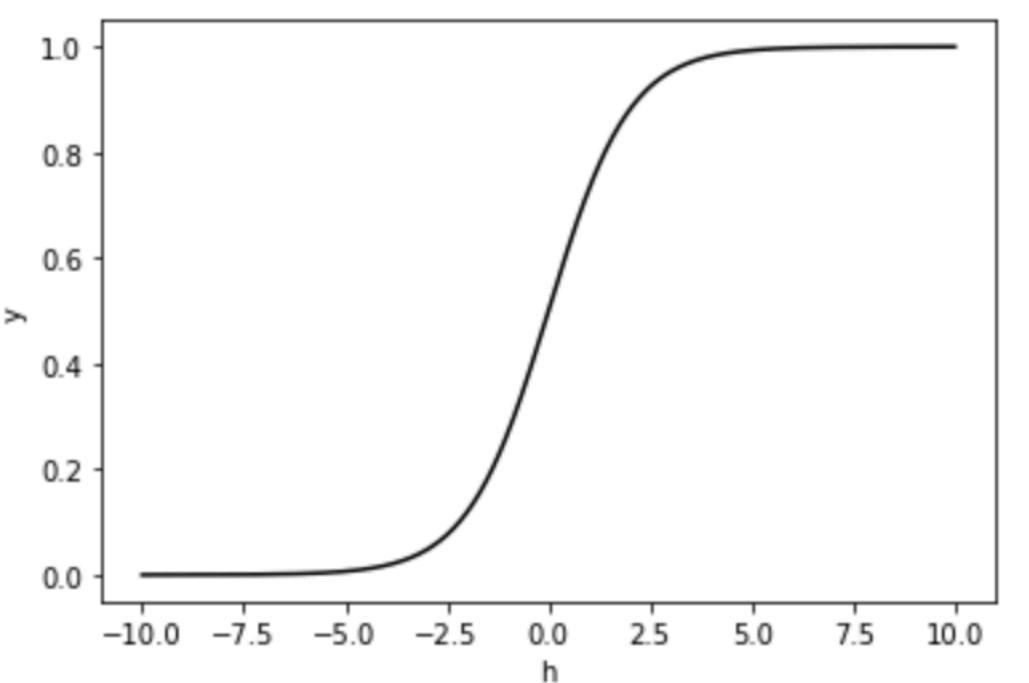


<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

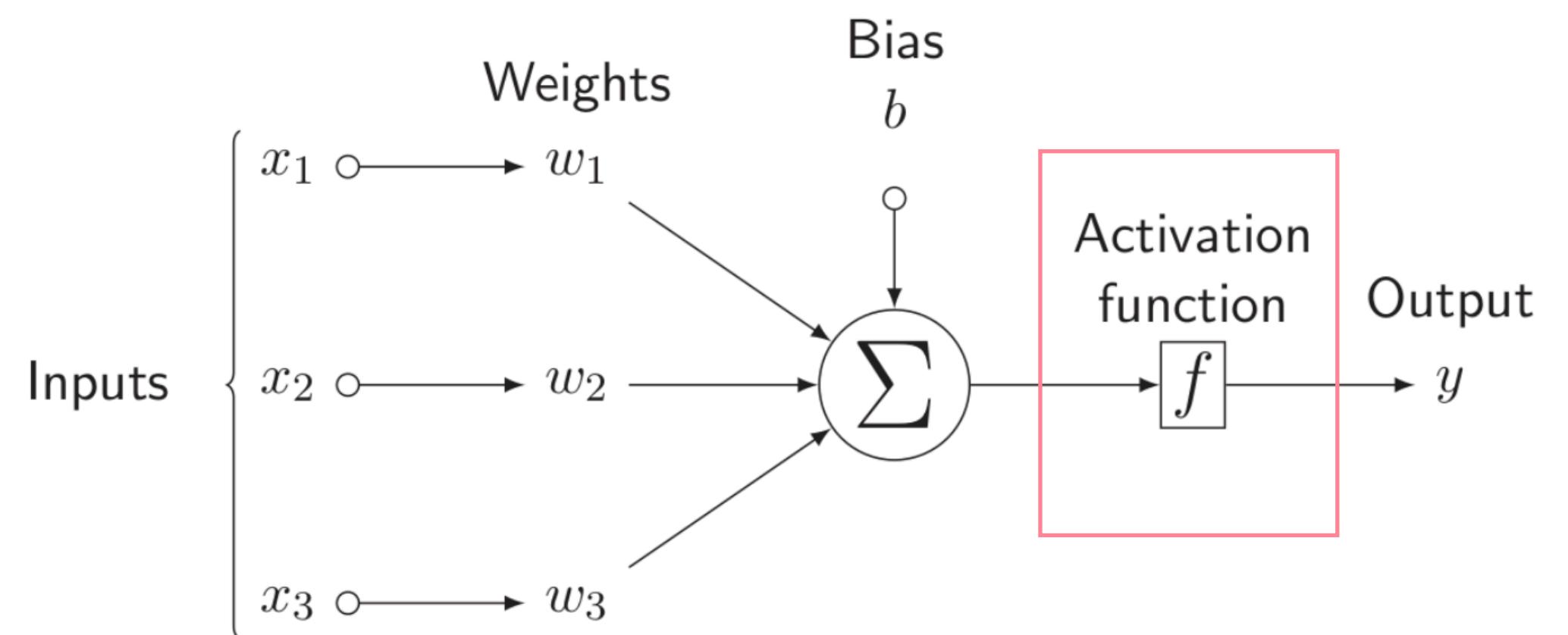
The classification is a bit advanced
because the model is complex



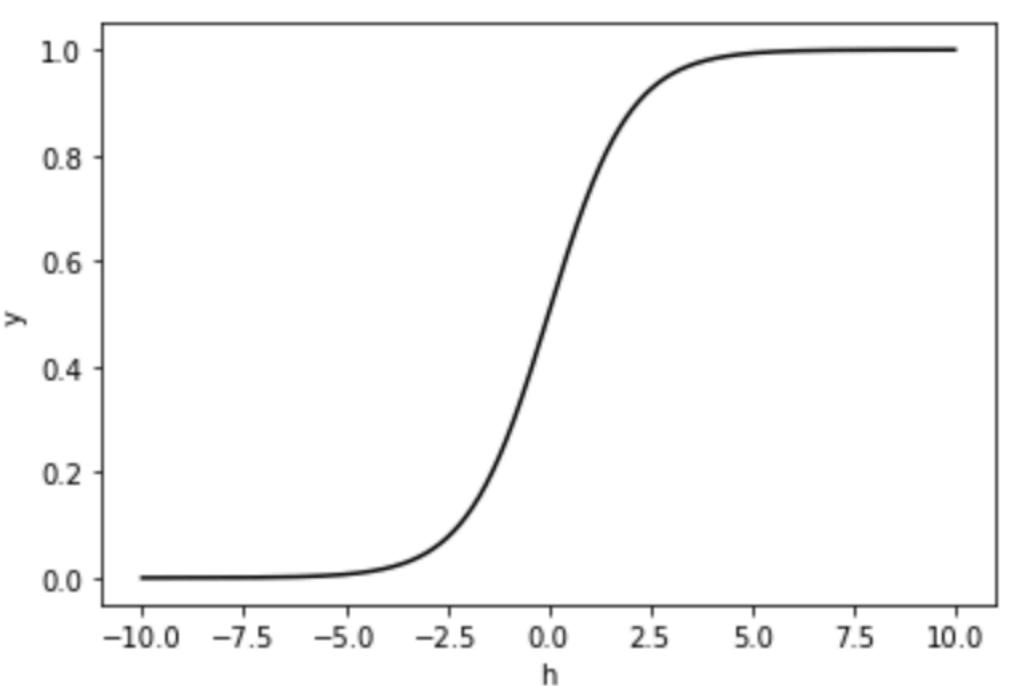
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>



The classification is a bit advanced
because the model is complex

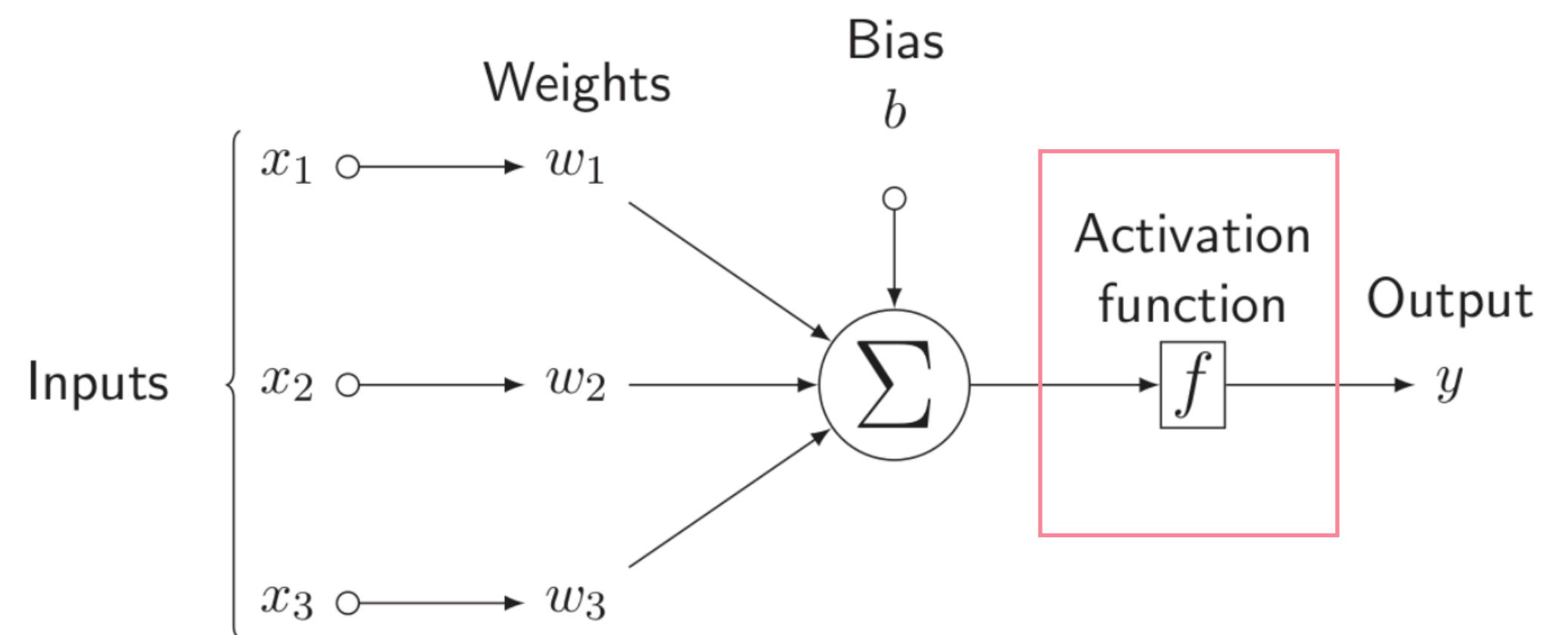


<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

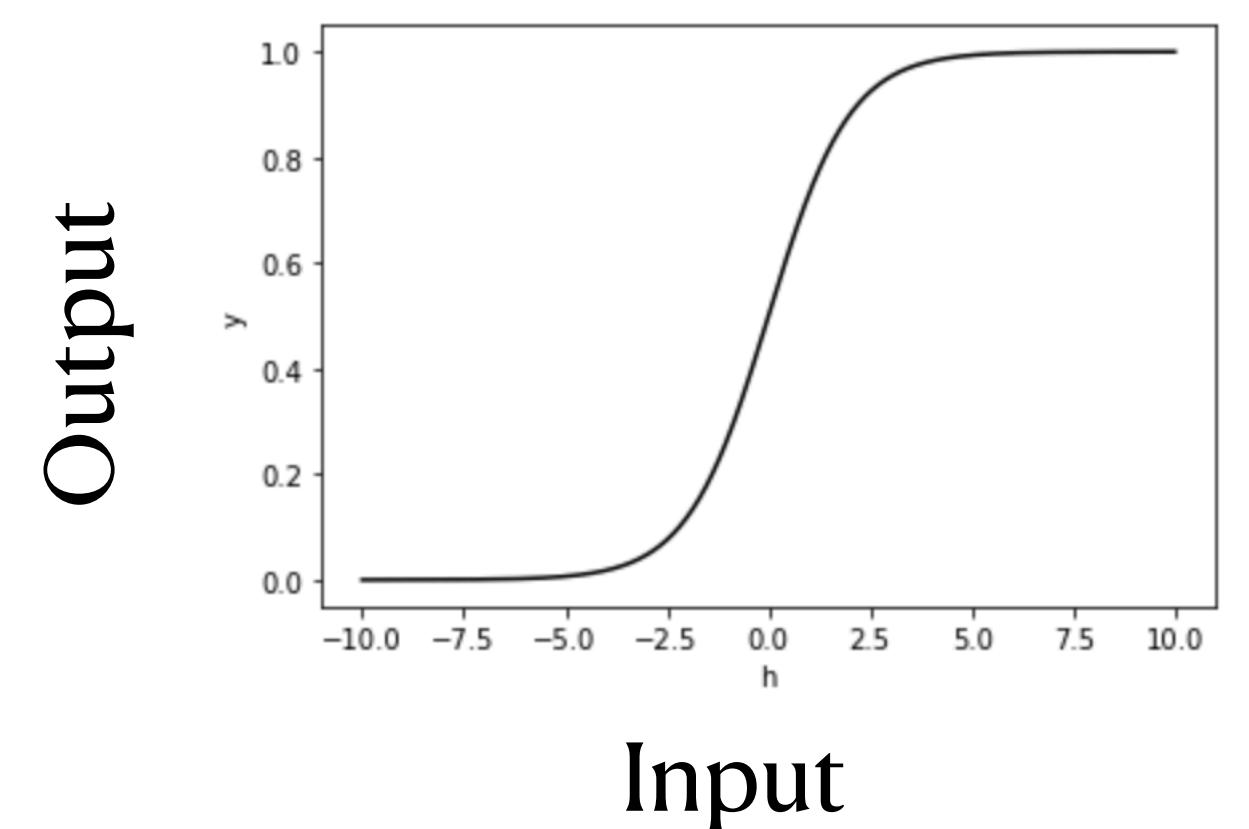


Input

The classification is a bit advanced
because the model is complex

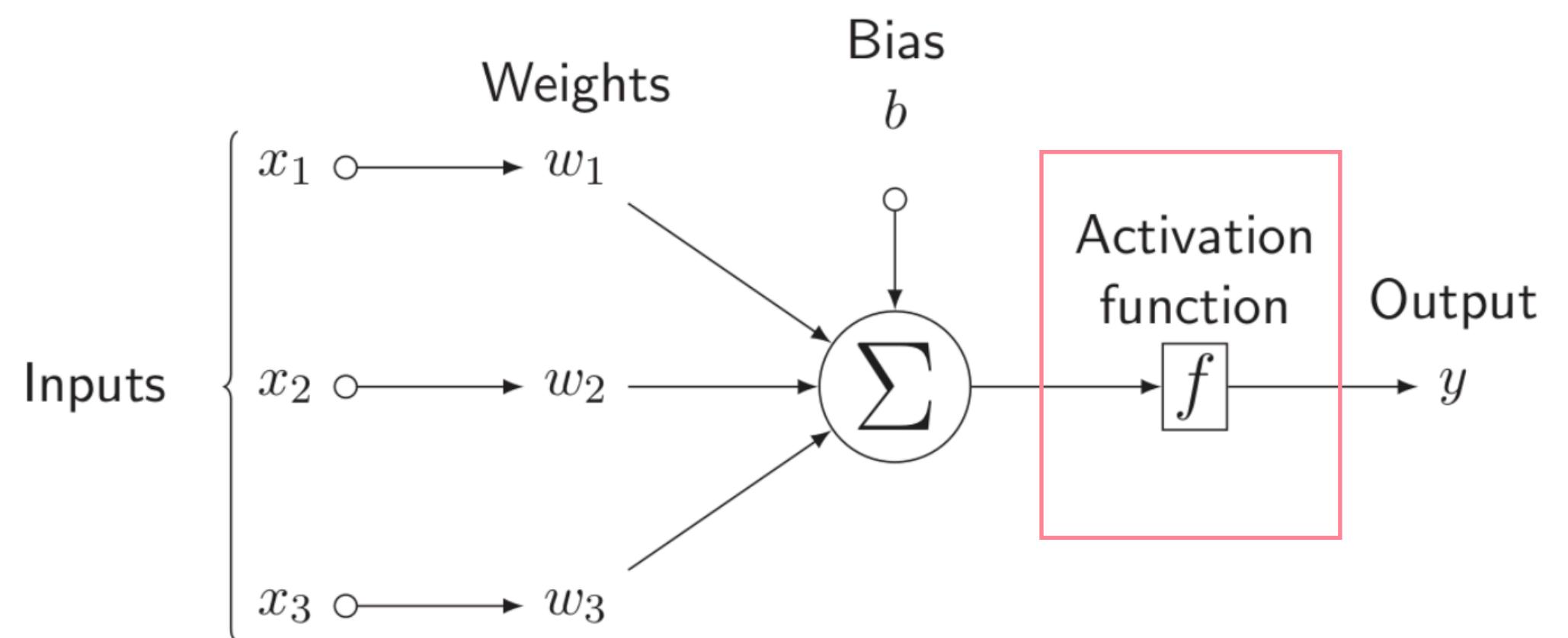


<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

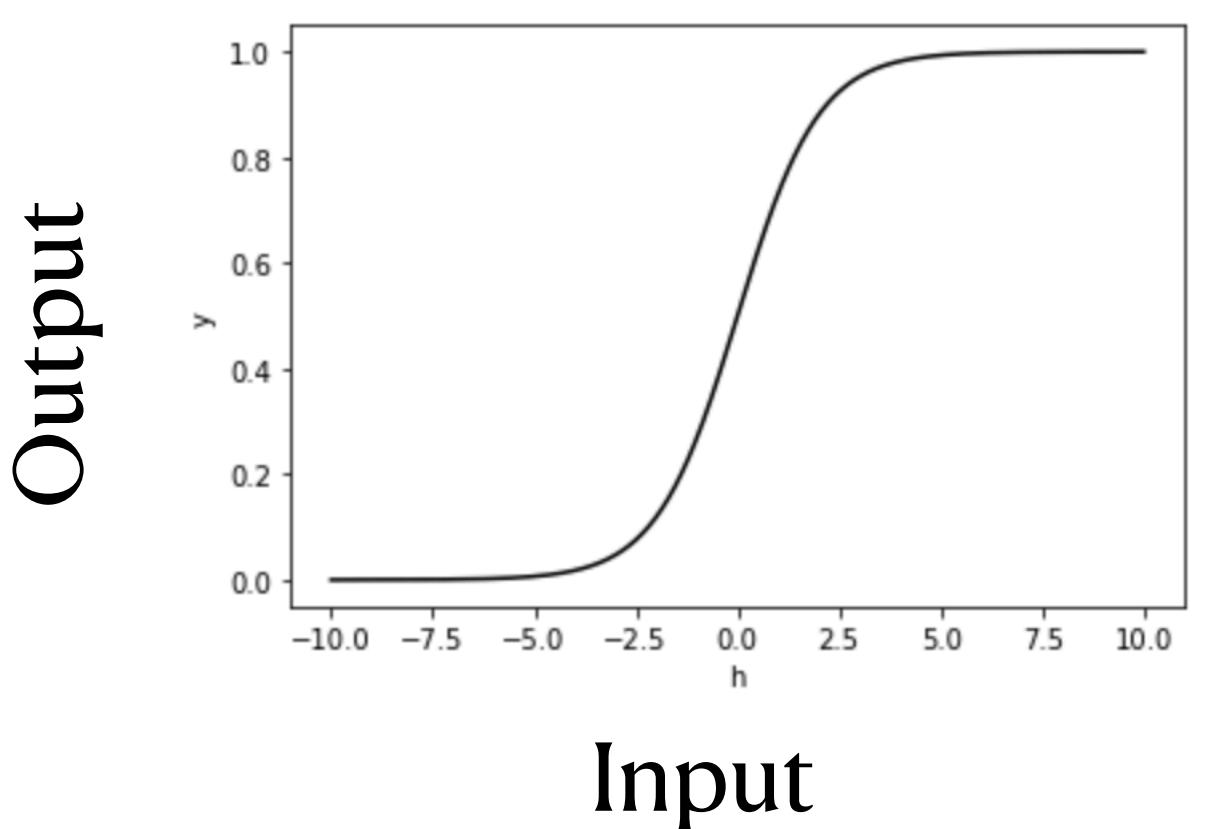


The classification is a bit advanced
because the model is complex

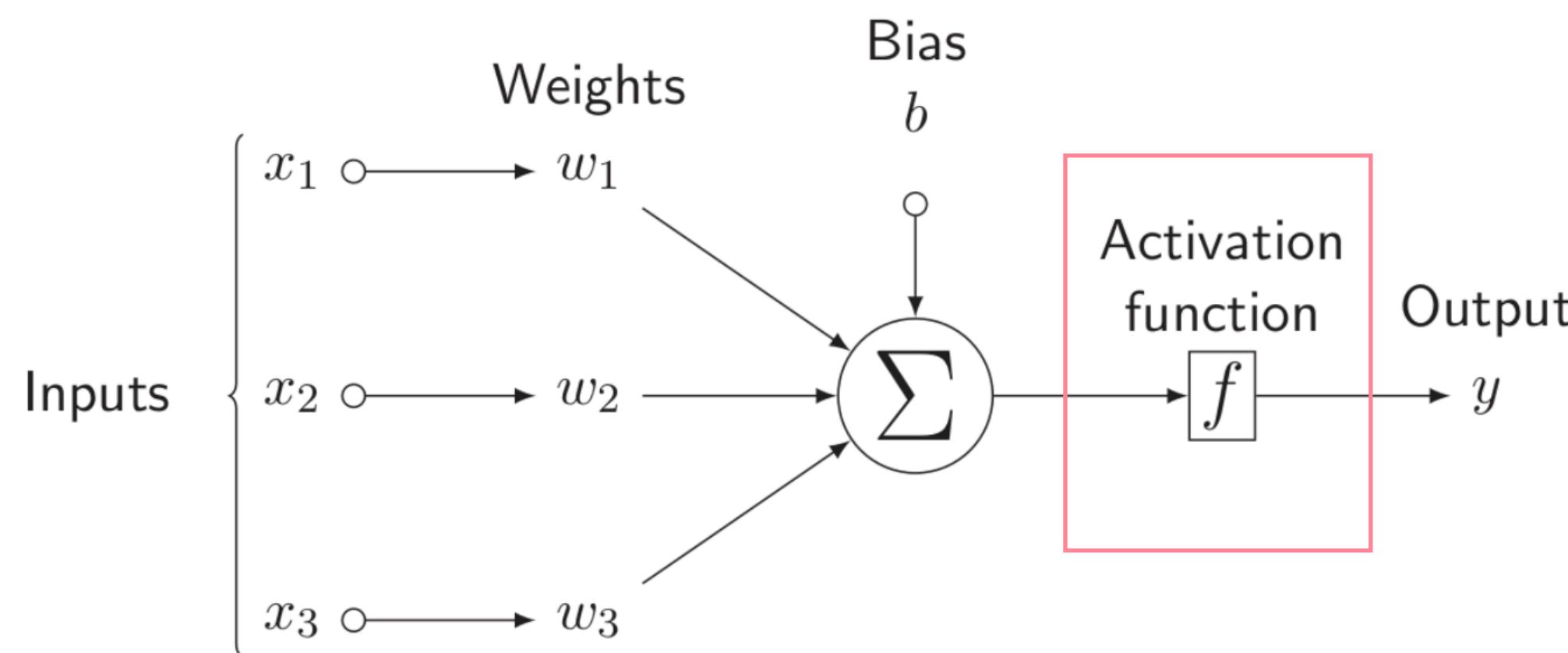
Now we want to consider a simpler task



<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

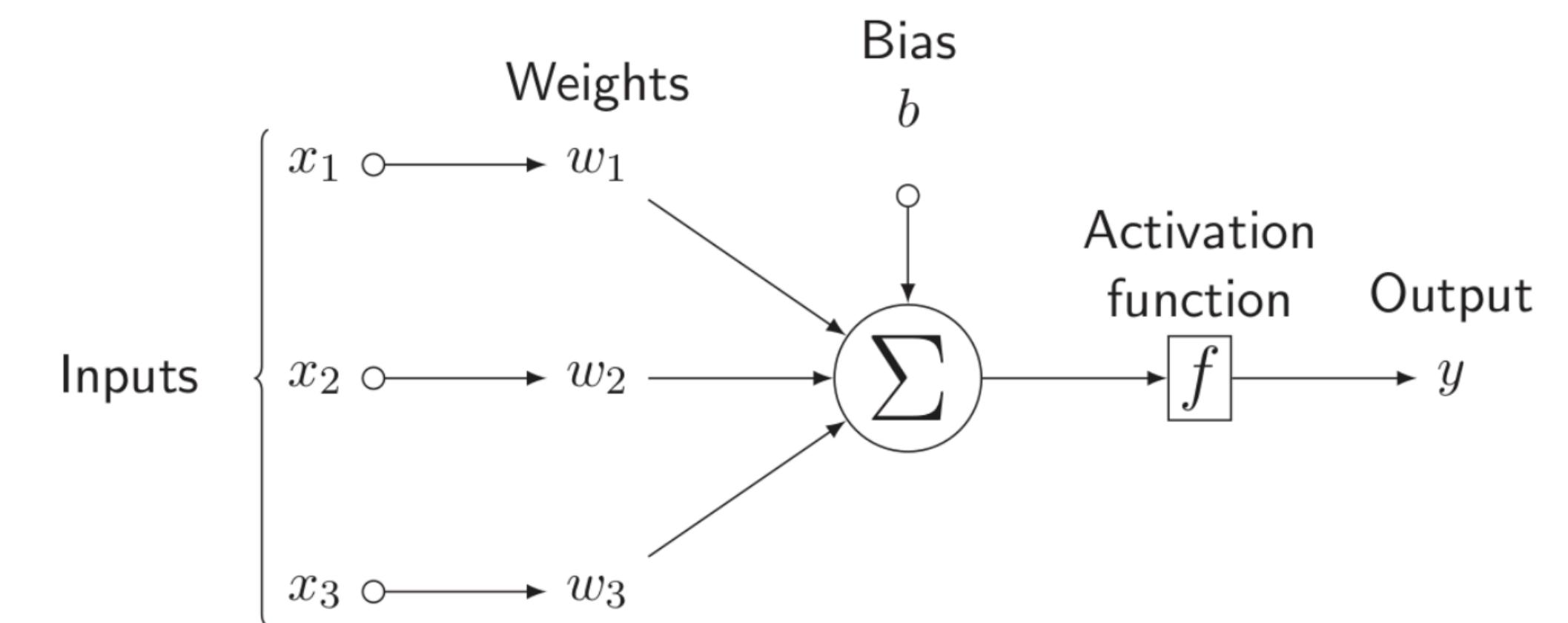


The classification is a bit advanced
because the model is complex

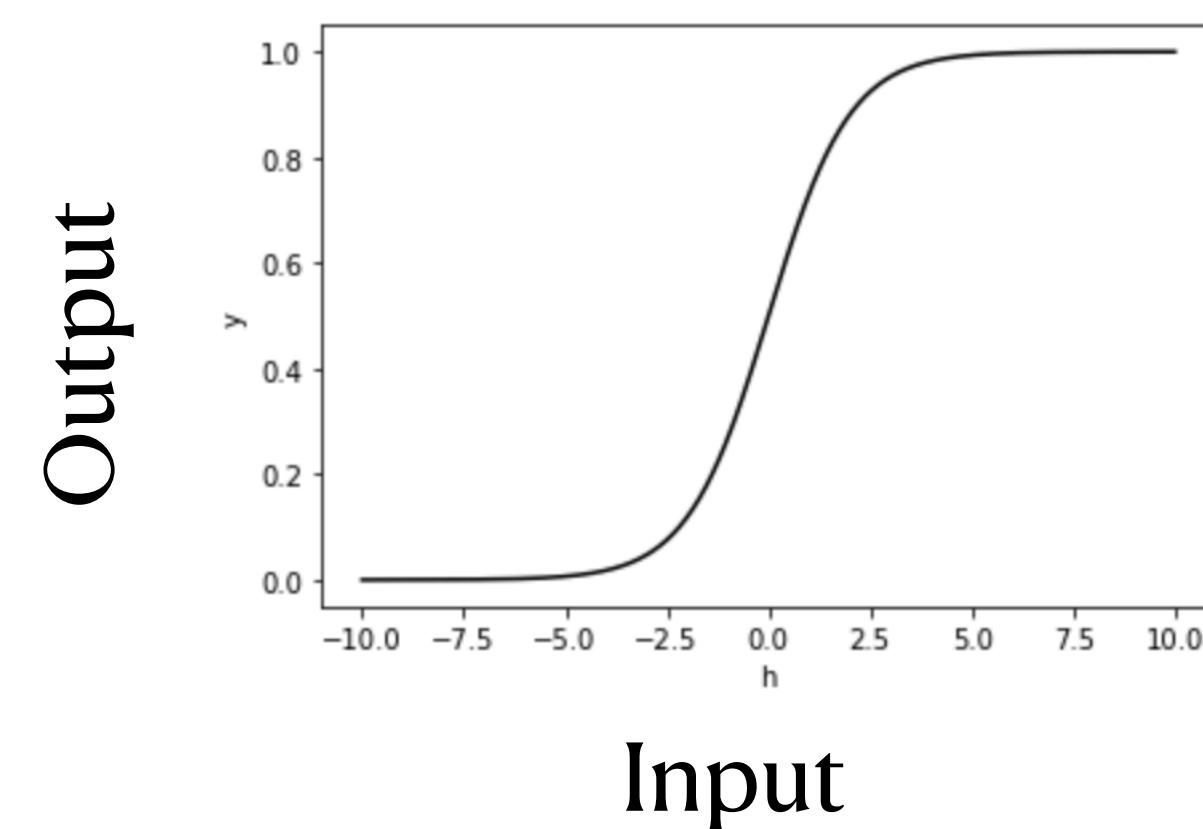


<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

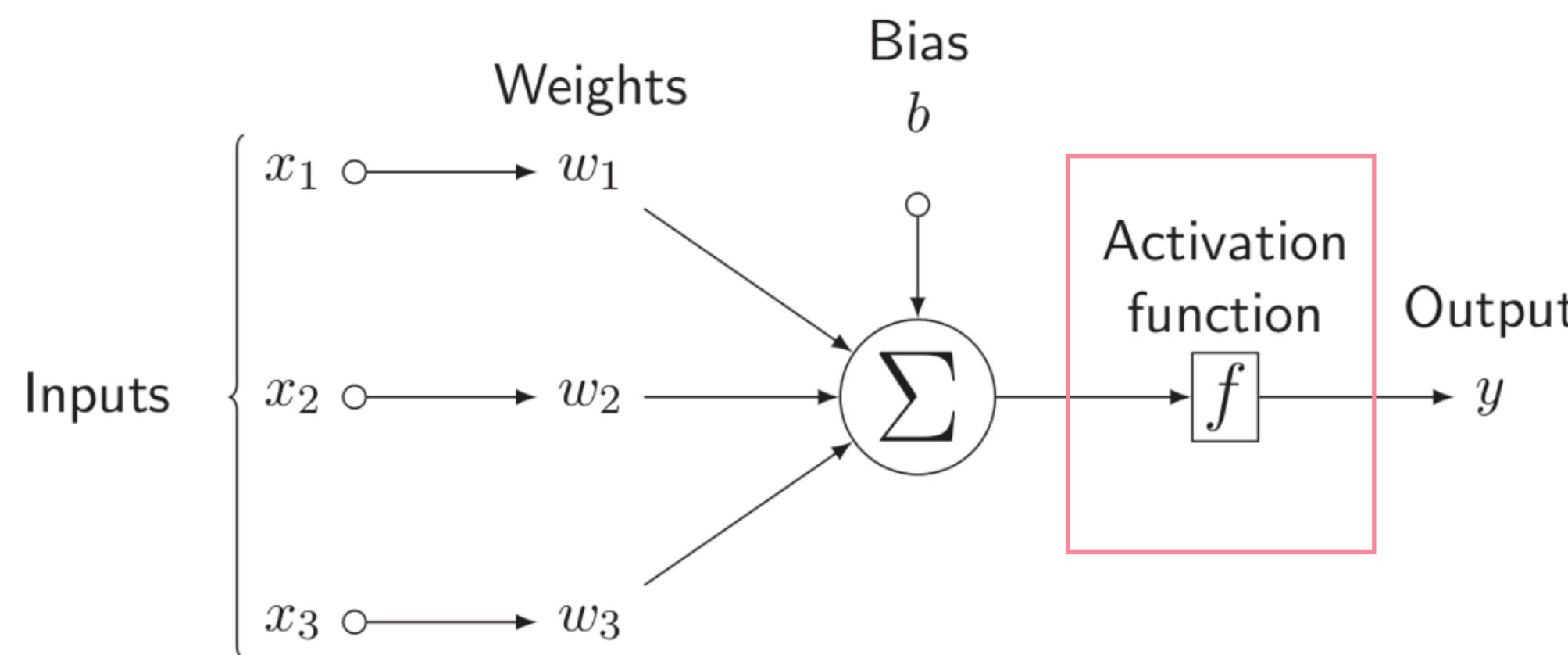
Now we want to consider a simpler task



<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

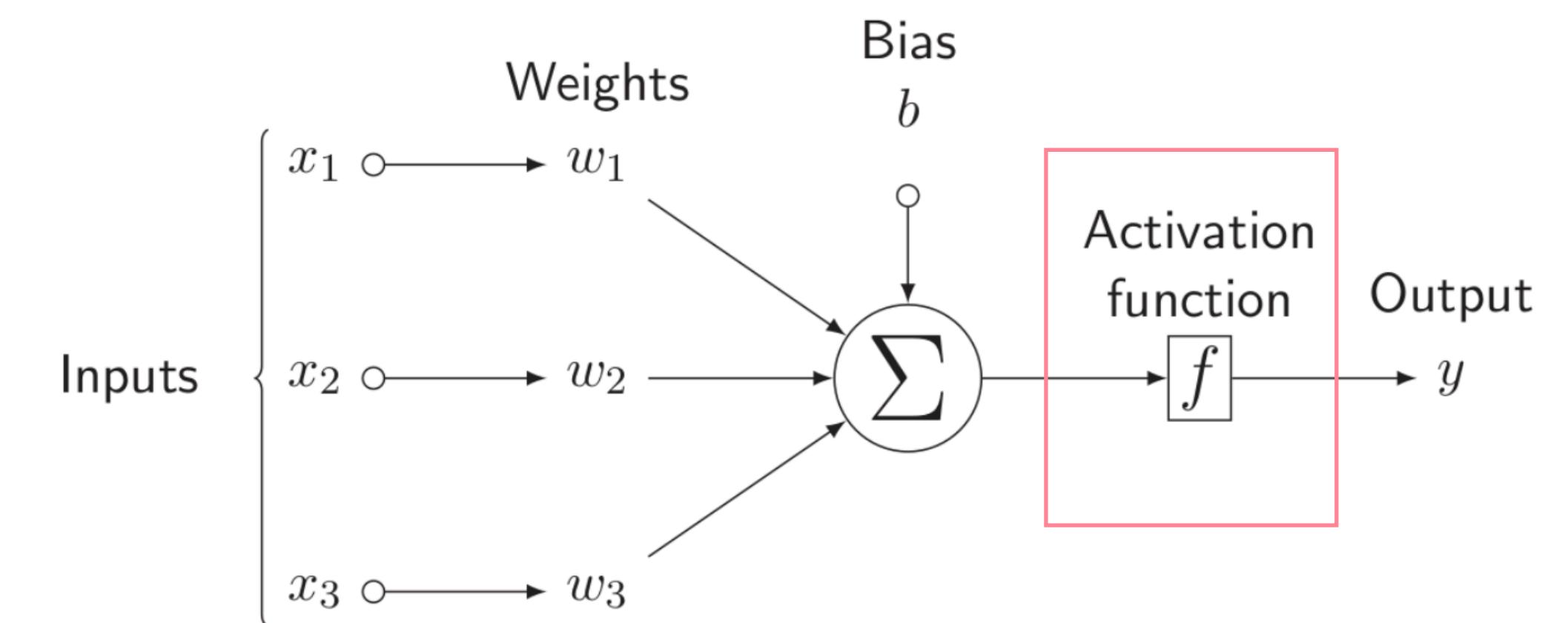


The classification is a bit advanced
because the model is complex

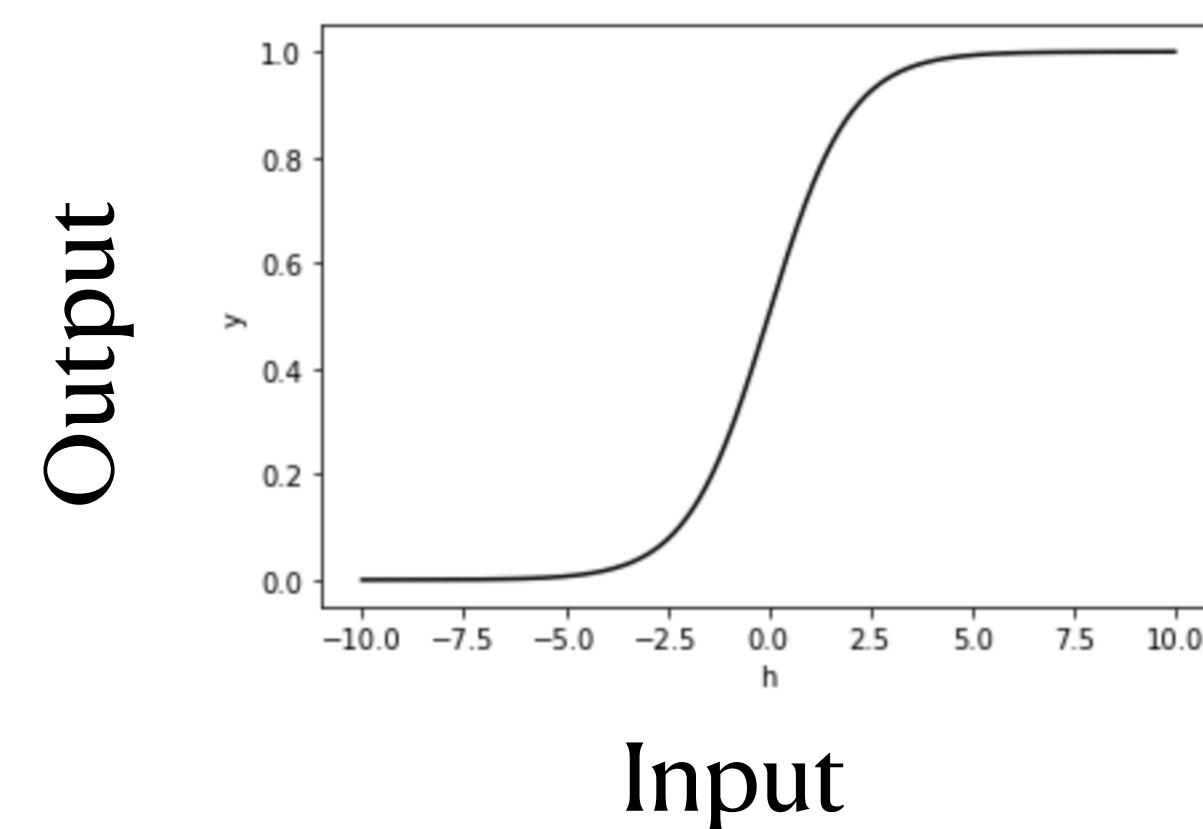


<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

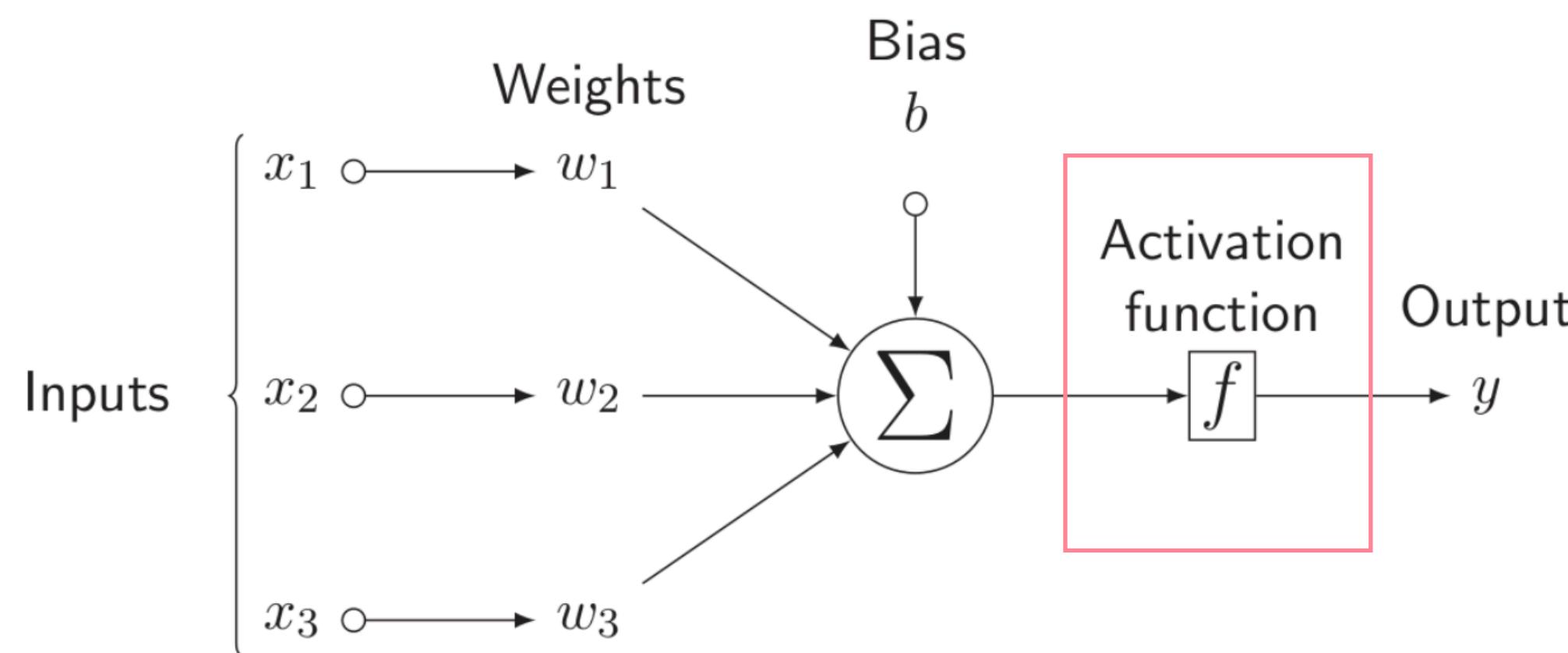
Now we want to consider a simpler task



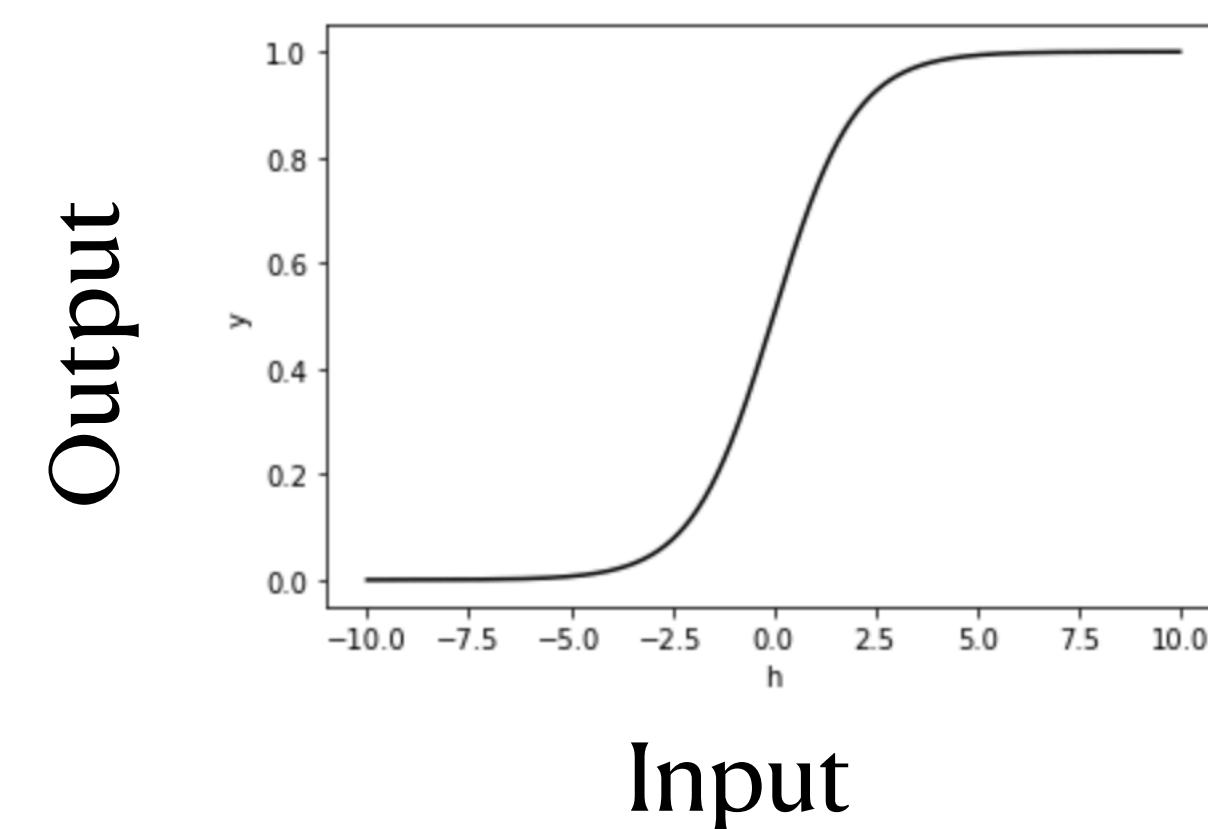
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>



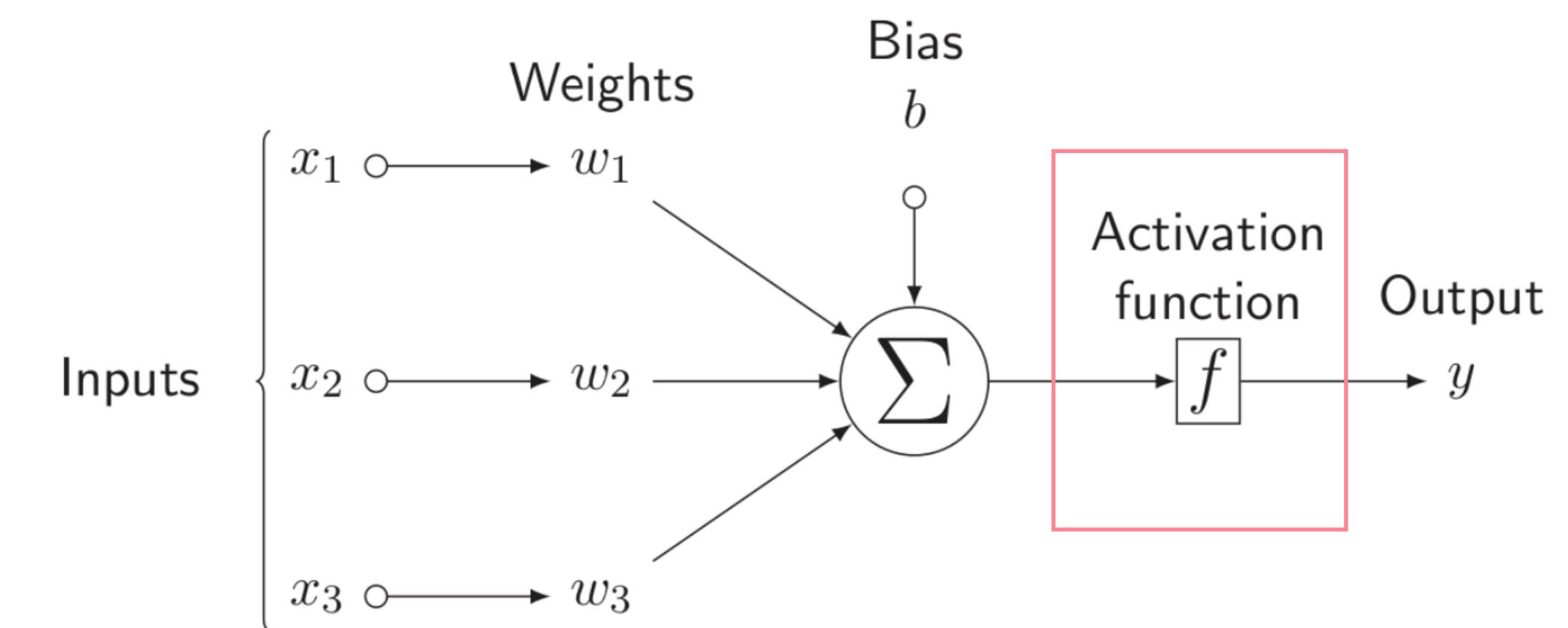
The classification is a bit advanced
because the model is complex



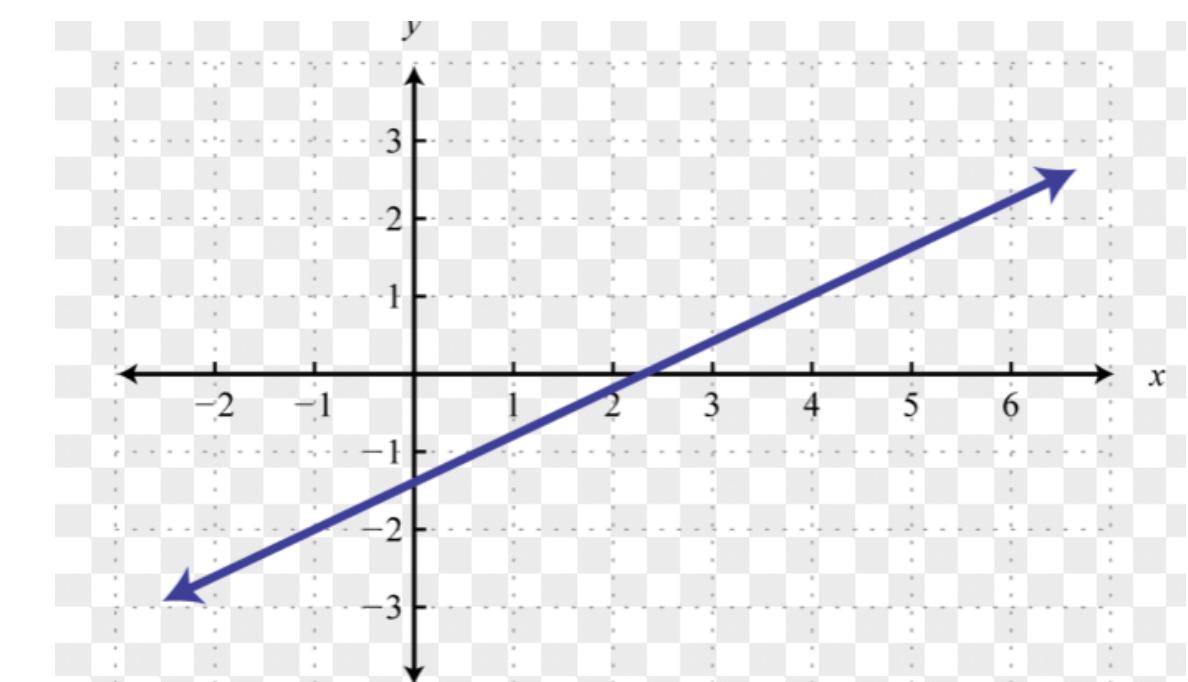
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>



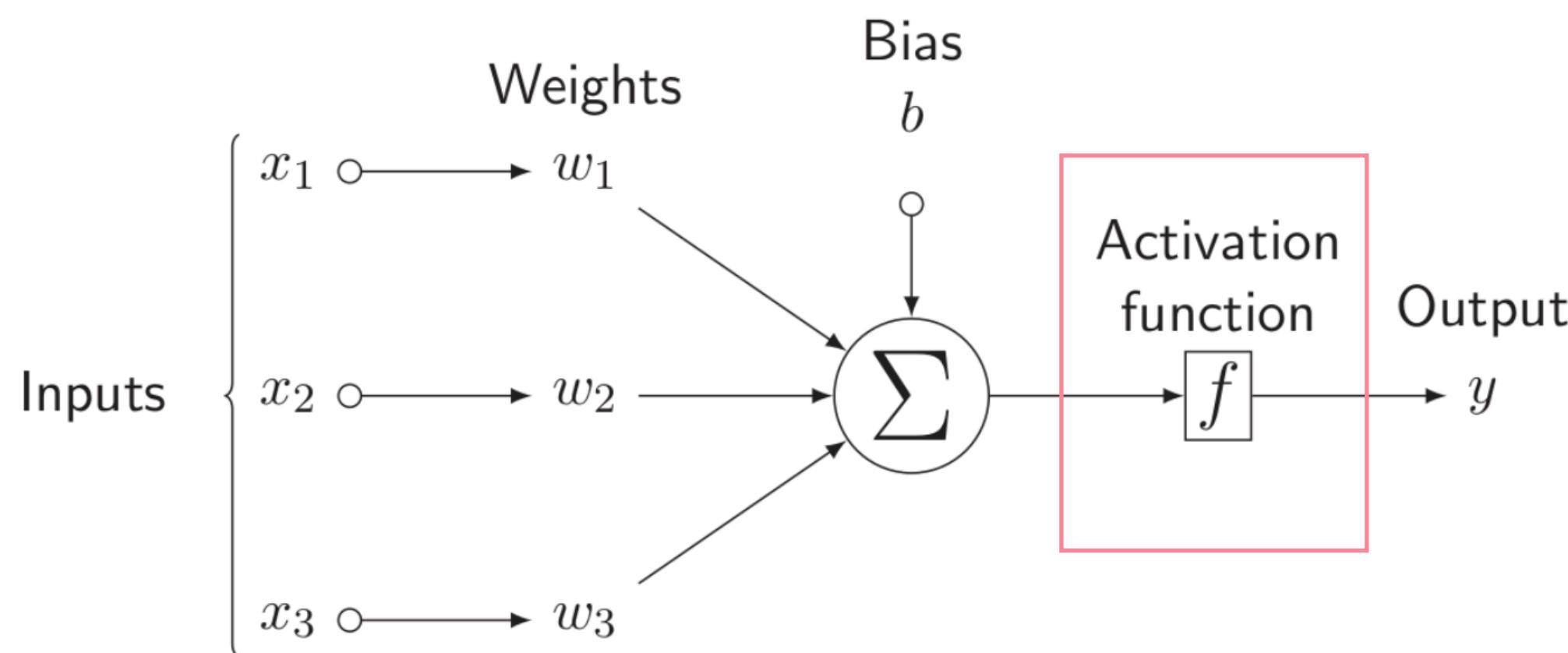
Now we want to consider a simpler task



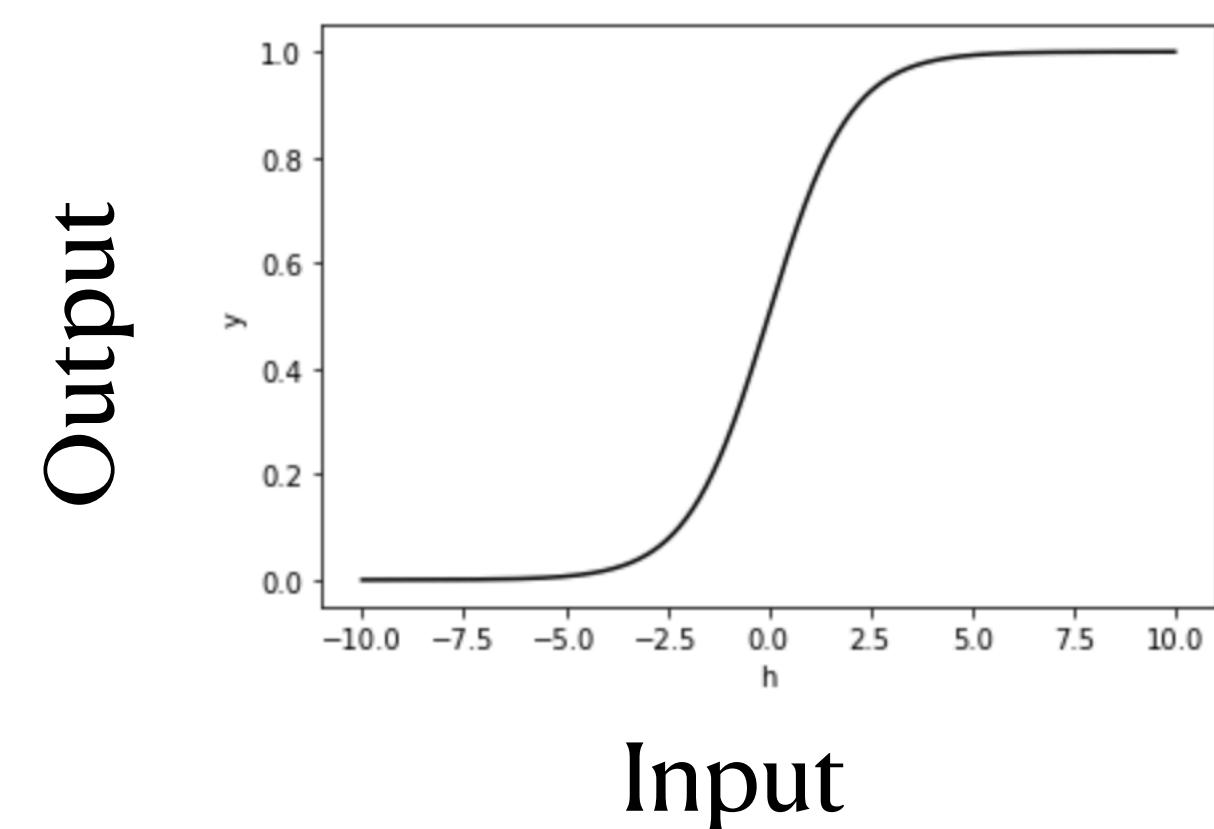
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>



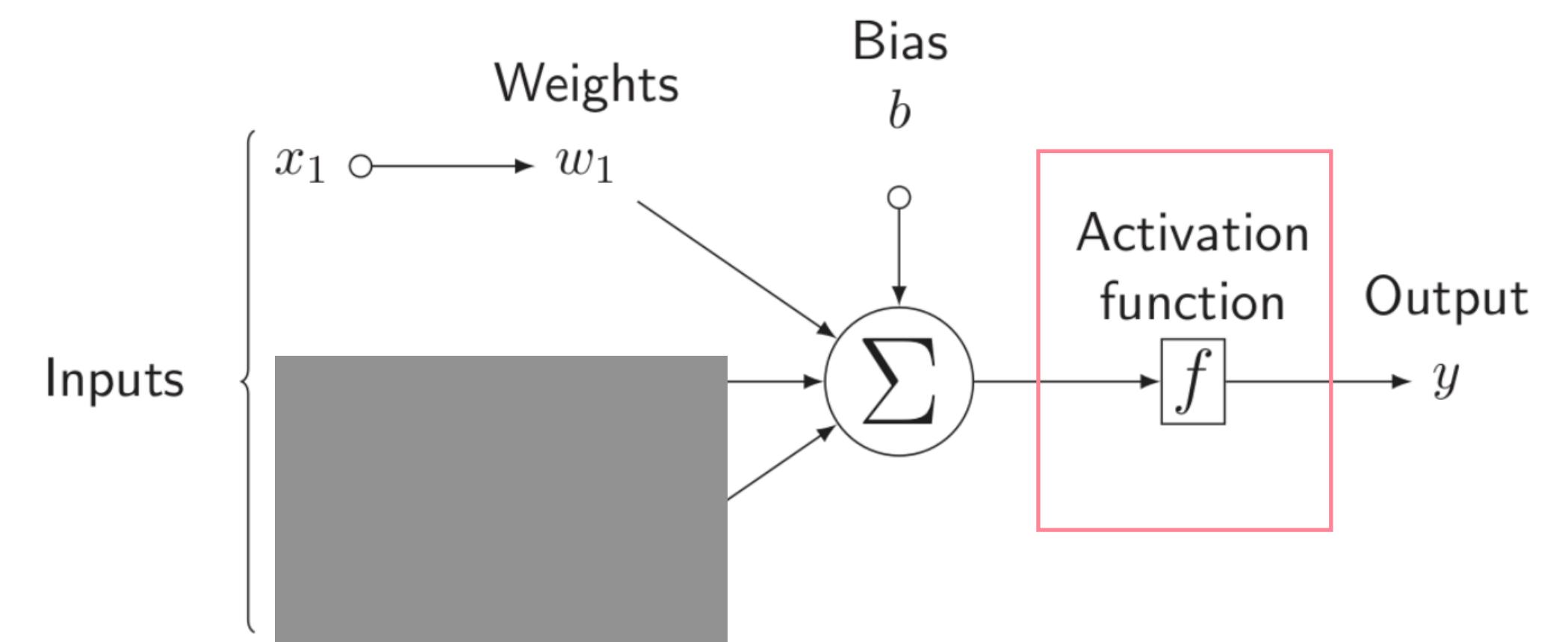
The classification is a bit advanced
because the model is complex



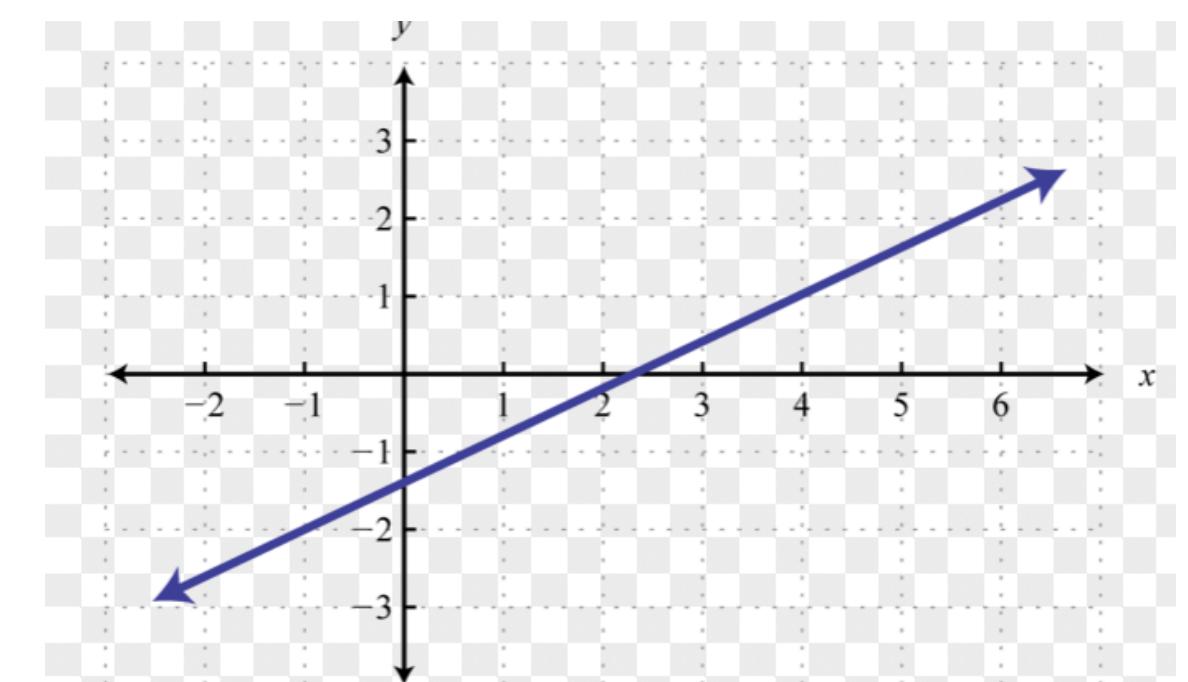
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>



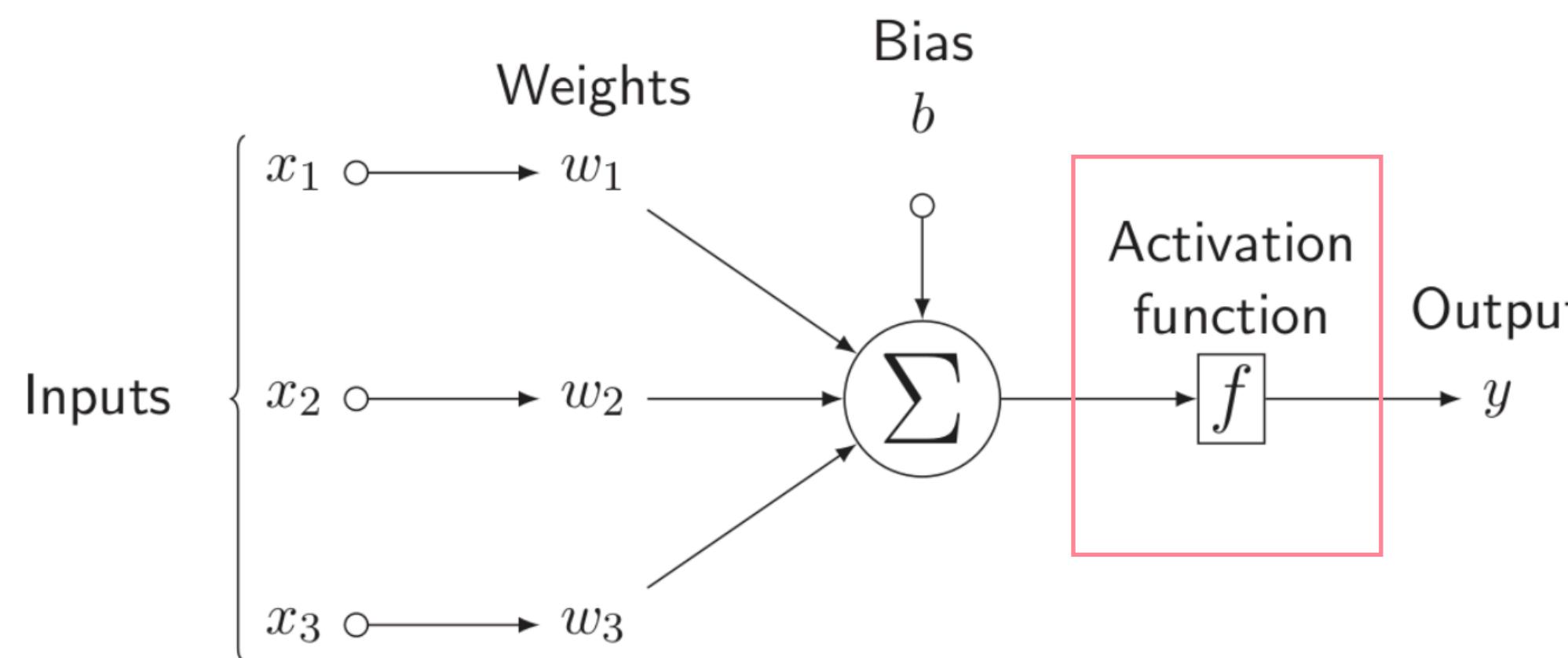
Now we want to consider a simpler task



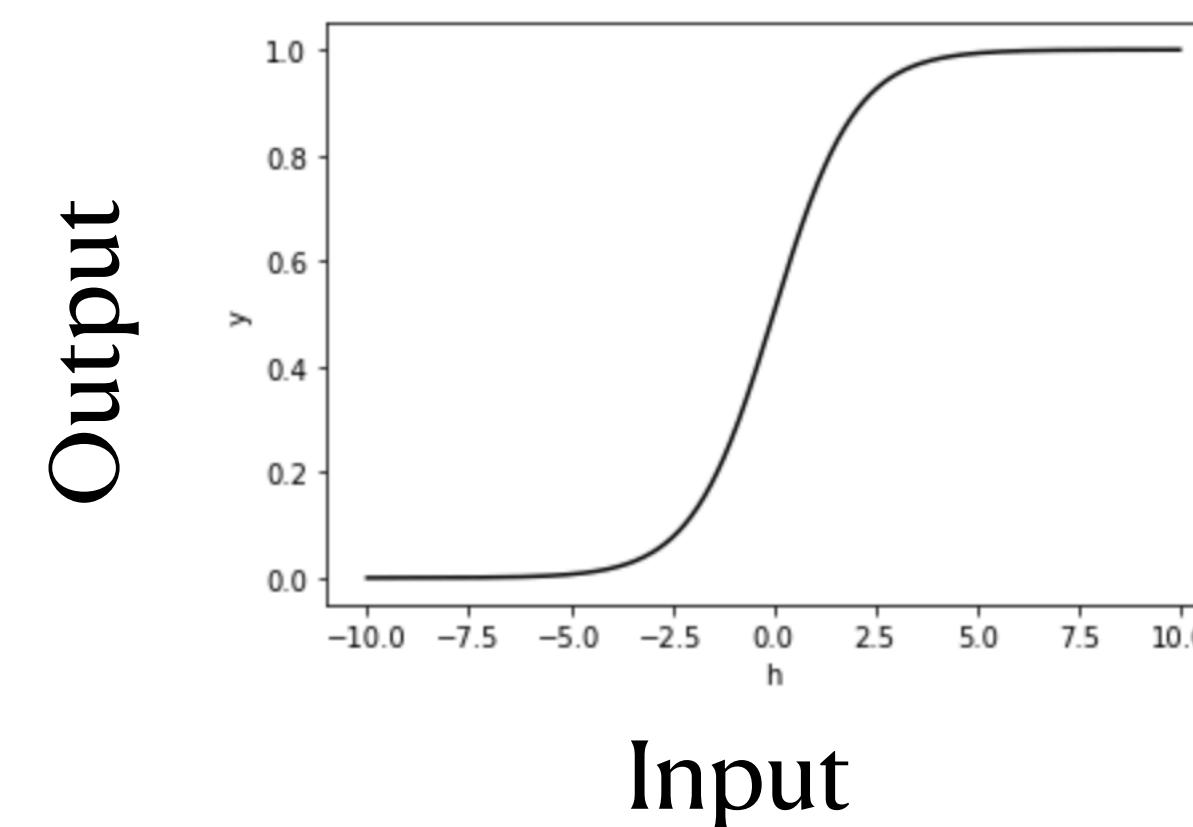
<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>



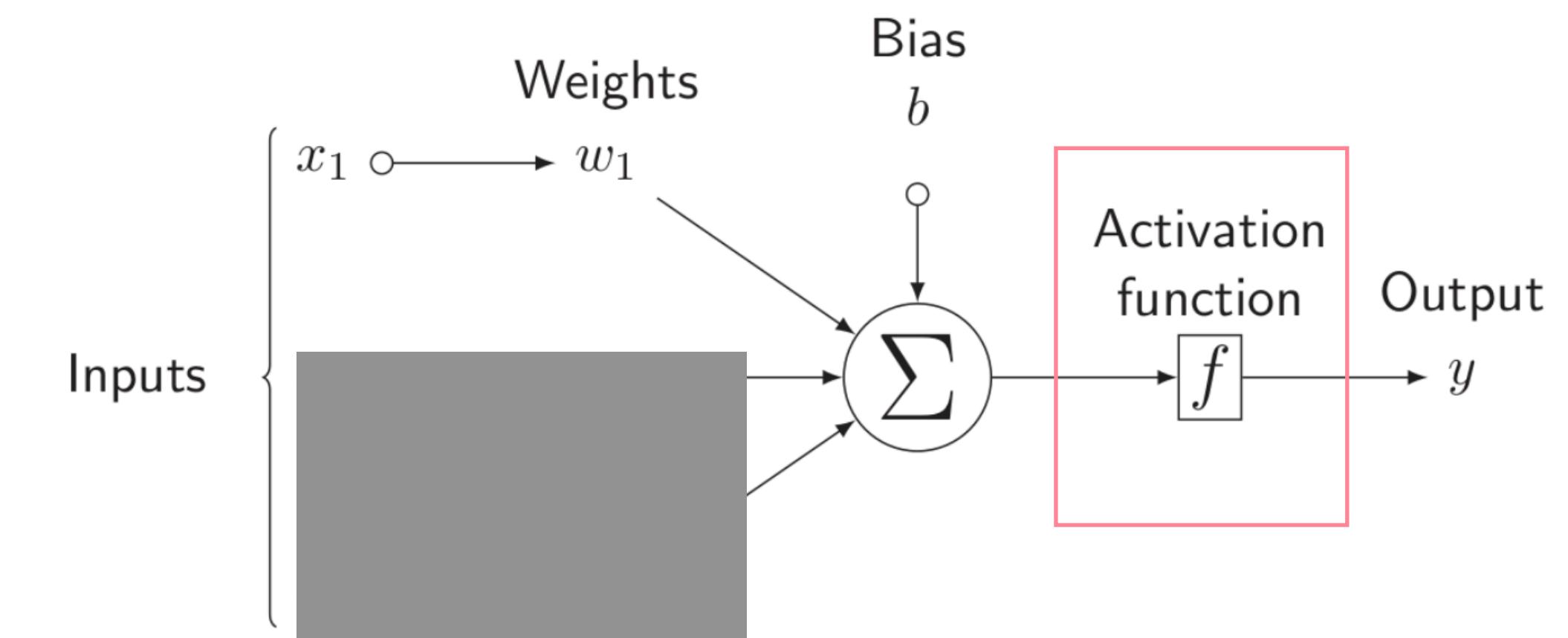
The classification is a bit advanced because the model is complex



<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

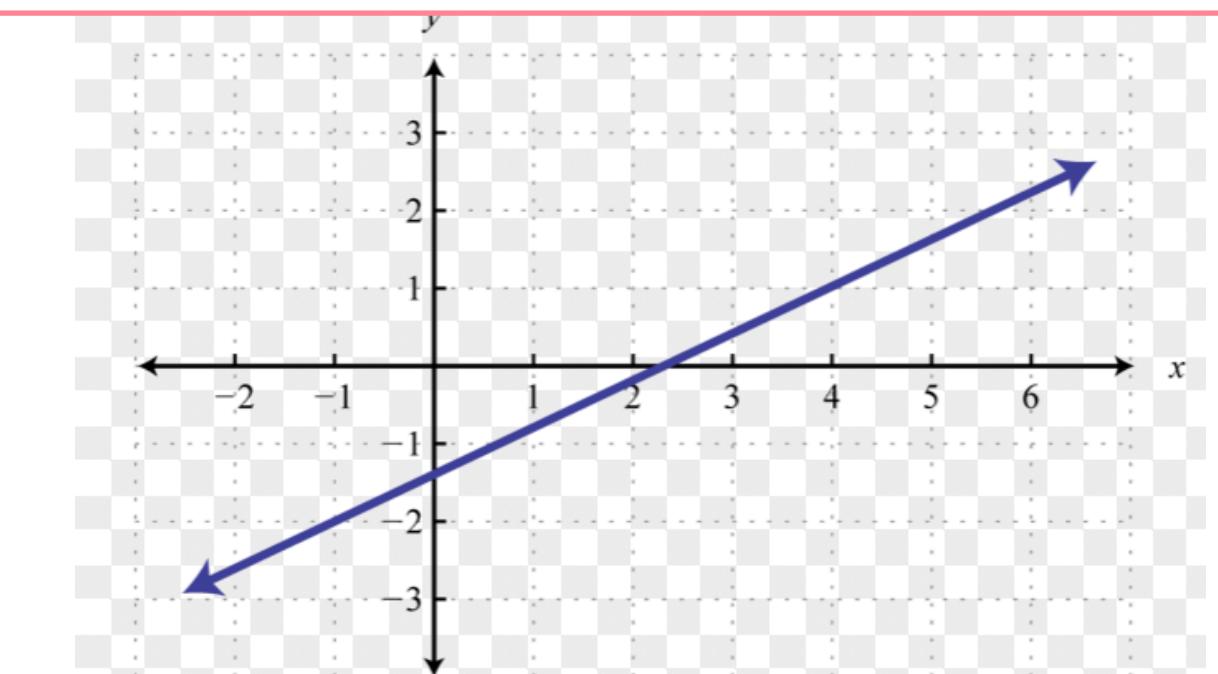


Now we want to consider a simpler task



<https://tex.stackexchange.com/questions/132444/diagram-of-an-artificial-neural-network>

This becomes a linear REGRESSION task!!



A real-world problem

Advertising data set

Reading: You shall read p.59-p.64 first

In [26]: `advertising.head()`

Out [26]:

	Unnamed: 0	TV	radio	newspaper	sales
0	1	230.1	37.8	69.2	22.1
1	2	44.5	39.3	45.1	10.4
2	3	17.2	45.9	69.3	9.3
3	4	151.5	41.3	58.5	18.5
4	5	180.8	10.8	58.4	12.9

We will review simple pandas in Lab video!

A real-world problem

Advertising data set

Reading: You shall read p.59-p.64 first

In [26]: `advertising.head()`

Out [26]:

	Unnamed: 0	TV	radio	newspaper	sales
0	1	230.1	37.8	69.2	22.1
1	2	44.5	39.3	45.1	10.4
2	3	17.2	45.9	69.3	9.3
3	4	151.5	41.3	58.5	18.5
4	5	180.8	10.8	58.4	12.9

We will review simple pandas in Lab video!

A real-world problem

Advertising data set

Reading: You shall read p.59-p.64 first

In [26]: `advertising.head()`

Out [26]:

	Unnamed: 0	TV	radio	newspaper	sales
0	1	230.1	37.8	69.2	22.1
1	2	44.5	39.3	45.1	10.4
2	3	17.2	45.9	69.3	9.3
3	4	151.5	41.3	58.5	18.5
4	5	180.8	10.8	58.4	12.9

Input

We will review simple pandas in Lab video!

A real-world problem

Advertising data set

Reading: You shall read p.59-p.64 first

In [26]: `advertising.head()`

Out [26]:

	Unnamed: 0	TV	radio	newspaper	sales
0	1	230.1	37.8	69.2	22.1
1	2	44.5	39.3	45.1	10.4
2	3	17.2	45.9	69.3	9.3
3	4	151.5	41.3	58.5	18.5
4	5	180.8	10.8	58.4	12.9

Input

We will review simple pandas in Lab video!

A real-world problem

Advertising data set

Reading: You shall read p.59-p.64 first

In [26]: `advertising.head()`

Out [26]:

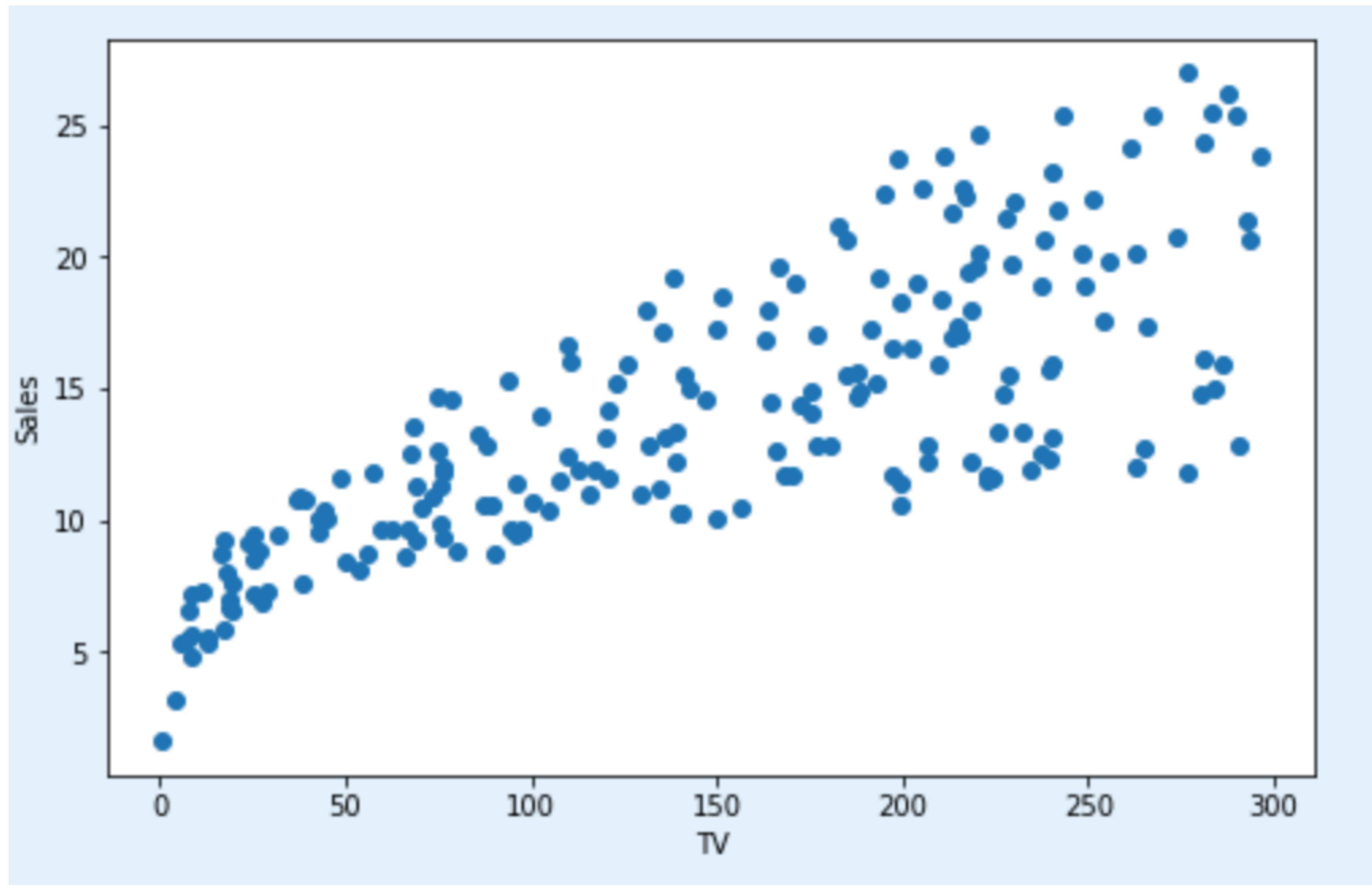
	Unnamed: 0	TV	radio	newspaper	sales
0	1	230.1	37.8	69.2	22.1
1	2	44.5	39.3	45.1	10.4
2	3	17.2	45.9	69.3	9.3
3	4	151.5	41.3	58.5	18.5
4	5	180.8	10.8	58.4	12.9

Input

Output

We will review simple pandas in Lab video!

Question: If the company is going to budget 100 units for the product in TV ad,
what is the expected values in the sales?



Sci-kit learn in python

<https://scikit-learn.org/stable/index.html>

The screenshot shows the official scikit-learn website. At the top, there's a navigation bar with links for 'Install', 'User Guide', 'API', 'Examples', 'More', a search bar, and a 'Go' button. The main title 'scikit-learn' is displayed prominently, followed by the subtitle 'Machine Learning in Python'. Below the title are three buttons: 'Getting Started', 'Release Highlights for 0.23', and 'GitHub'. To the right of the title, there's a list of bullet points highlighting the library's features:

- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

The page is divided into three main sections: 'Classification', 'Regression', and 'Clustering'. Each section contains a brief description, applications, algorithms, and a corresponding visualization.

- Classification:** Describes identifying which category an object belongs to. Applications include spam detection and image recognition. Algorithms listed are SVM, nearest neighbors, random forest, and more. A grid of small plots shows decision boundaries for various classifiers like K-Nearest Neighbors, Linear SVM, and Random Forest.
- Regression:** Describes predicting a continuous-valued attribute associated with an object. Applications include drug response and stock prices. Algorithms listed are SVR, nearest neighbors, random forest, and more. A line plot titled 'Boosted Decision Tree Regression' shows training samples (black dots) and two fitted models (green and red lines) for a target variable over an interval from 0 to 6.
- Clustering:** Describes automatic grouping of similar objects into sets. Applications include customer segmentation and grouping experiment outcomes. Algorithms listed are k-Means, spectral clustering, mean-shift, and more. A scatter plot titled 'K-means clustering on the digits dataset (PCA-reduced data)' shows a 2D PCA-reduced digit dataset with centroids marked by white crosses and colored clusters.

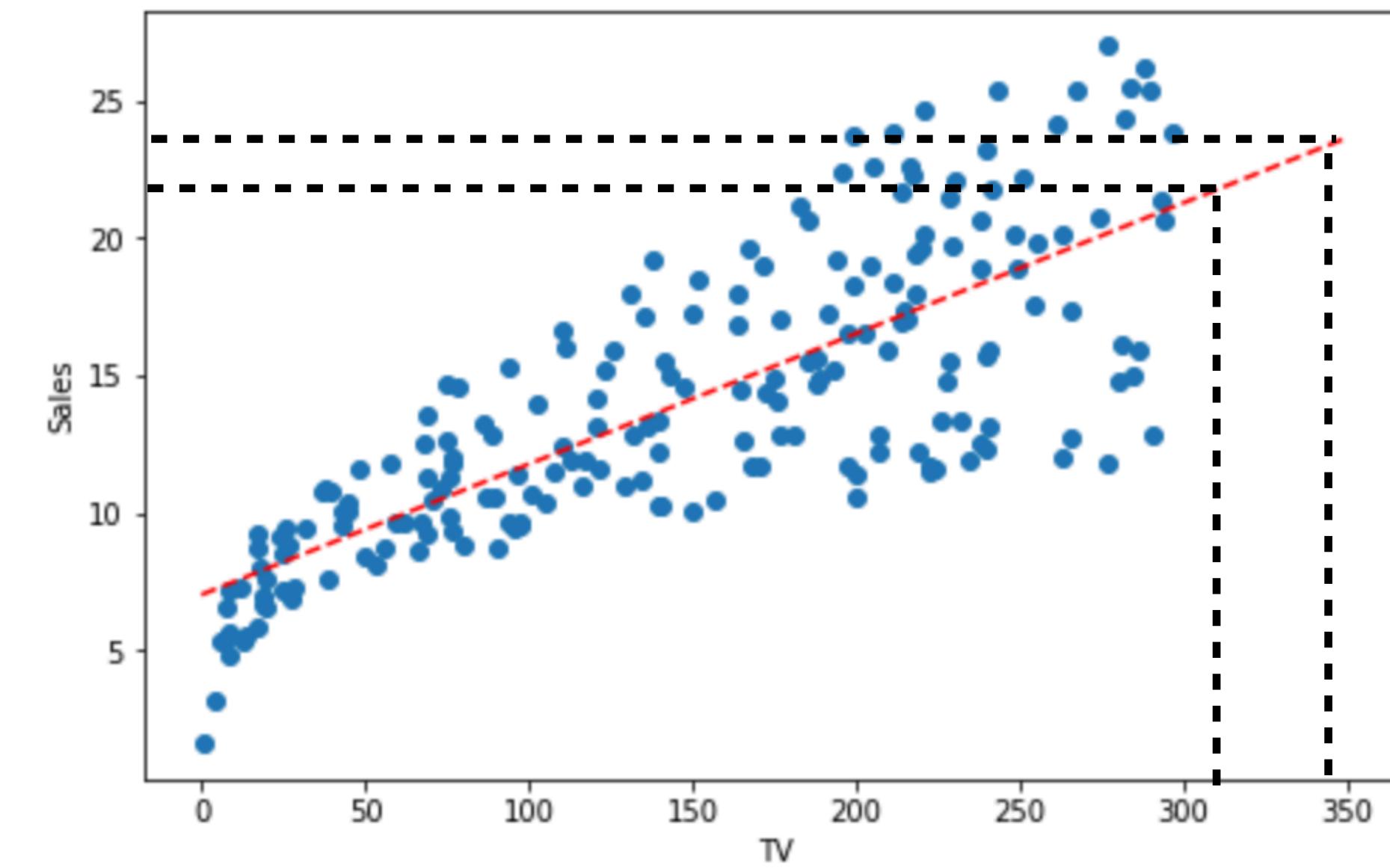
Use sk-learn to solve our problem

```
In [30]: model_full = LinearRegression().fit(X_data, y_data)

In [31]: sales_tv_slope = model_full.coef_
sales_tv_intercep = model_full.intercept_

x_star = [0.0,350.0]

plt.scatter(X_data,y_data)
plt.plot(x_star, reg_lin(x_star,sales_tv_slope,sales_tv_intercep), 'r--')
plt.xlabel('TV')
plt.ylabel('Sales')
```



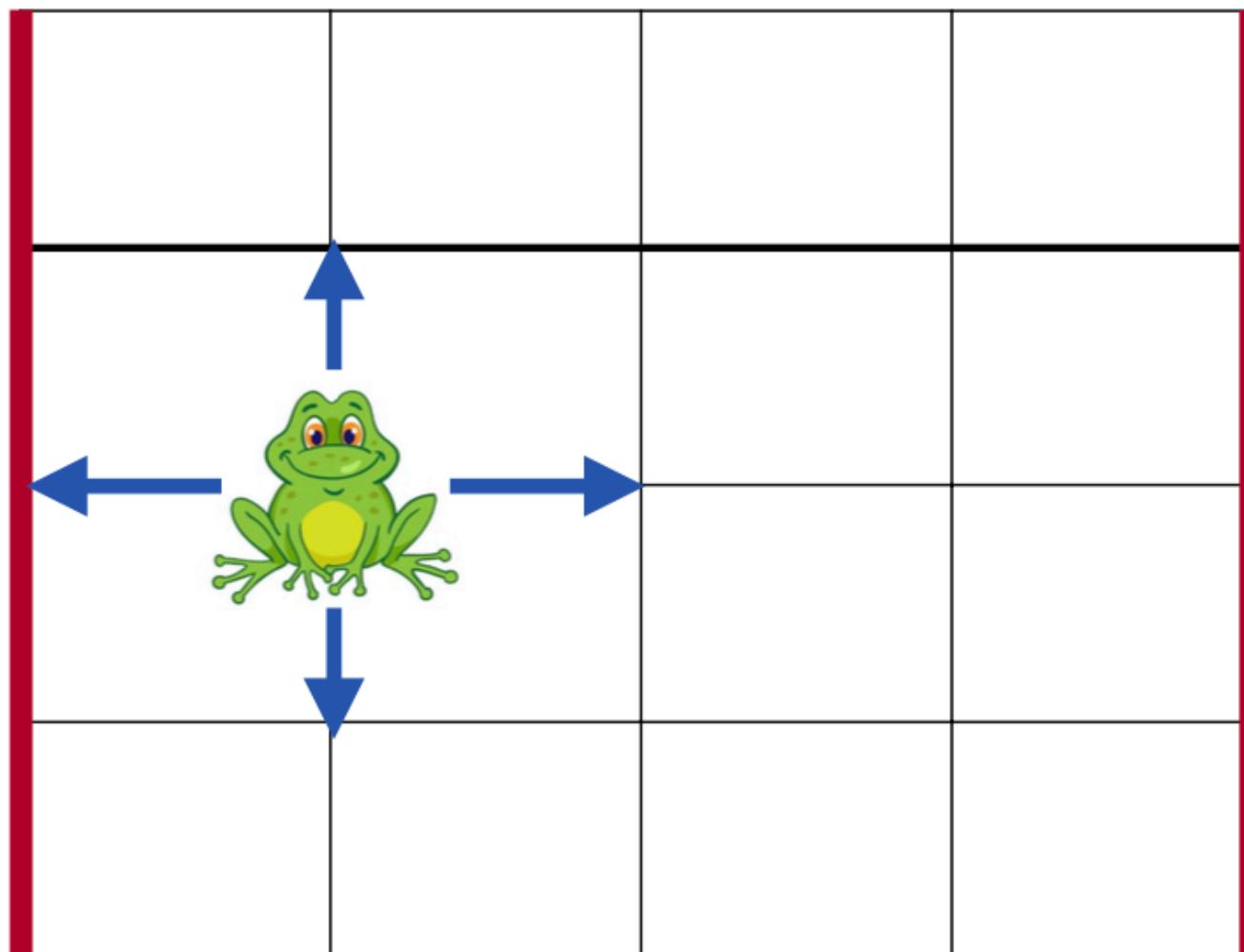
Homework

Simulation on a frog jump

Problem 13

A frog sitting at the point $(1, 2)$ begins a sequence of jumps, where each jump is parallel to one of the coordinate axes and has length 1, and the direction of each jump (up, down, right, or left) is chosen independently at random. The sequence ends when the frog reaches a side of the square with vertices $(0, 0)$, $(0, 4)$, $(4, 4)$, and $(4, 0)$. What is the probability that the sequence of jumps ends on a vertical side of the square?

- (A) $\frac{1}{2}$ (B) $\frac{5}{8}$ (C) $\frac{2}{3}$ (D) $\frac{3}{4}$ (E) $\frac{7}{8}$



https://artofproblemsolving.com/wiki/index.php/2020_AMC_10A_Problems