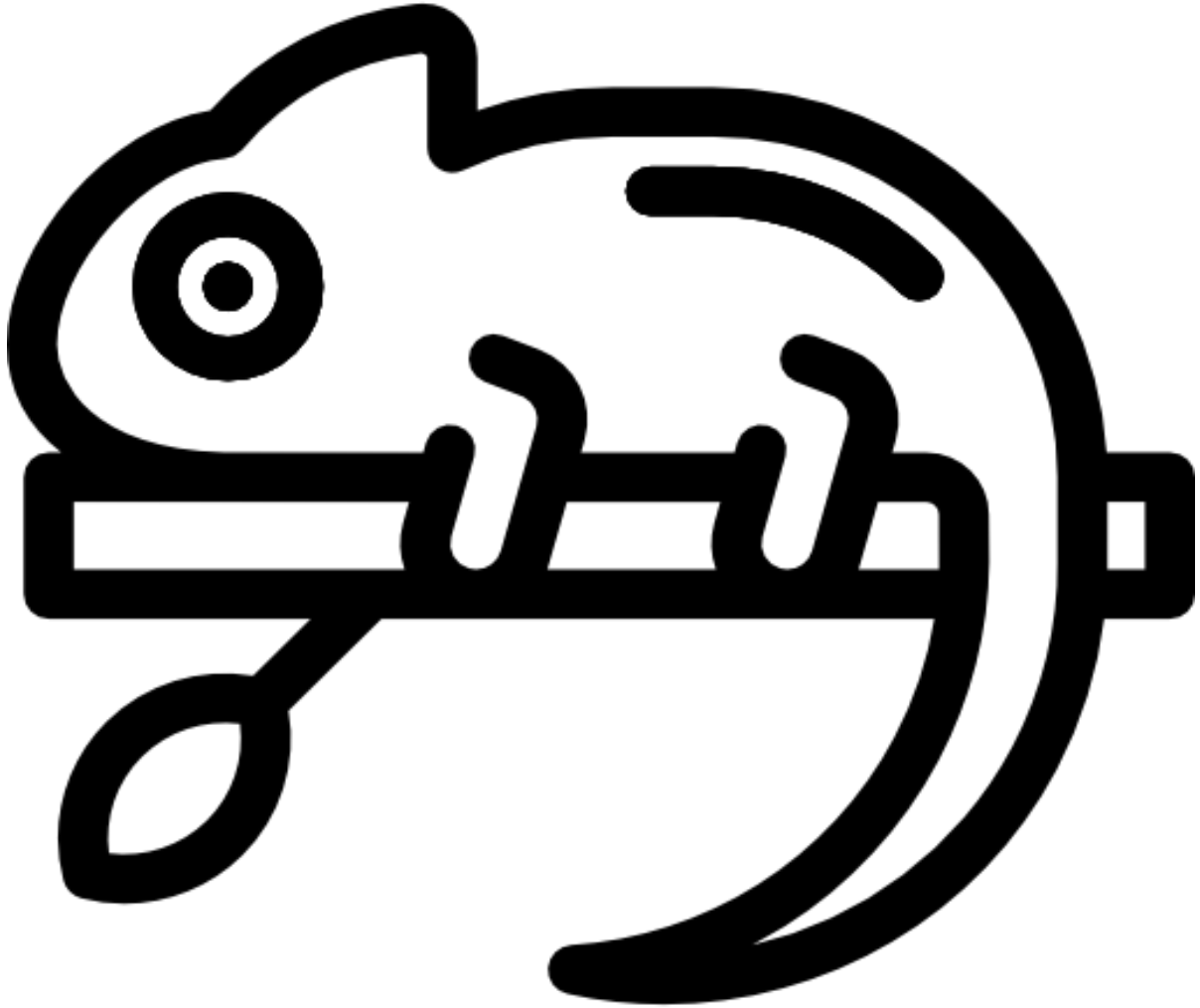
Introduction to Machine Learning with Python



2019 0000 옥현빈
2019 2650 이정훈

Contents

What is Machine Learning?

What is Machine Learning?

Tom Mitchell (1998)

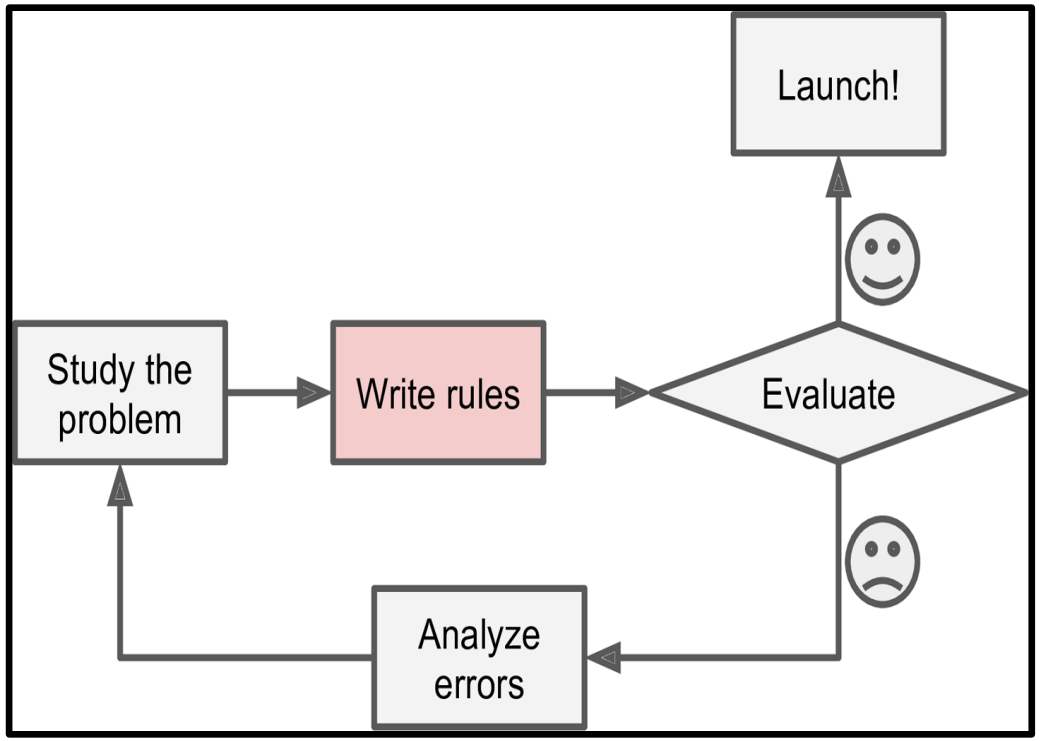
- T = Task
- P = Probability

Well-posed Learning Problem

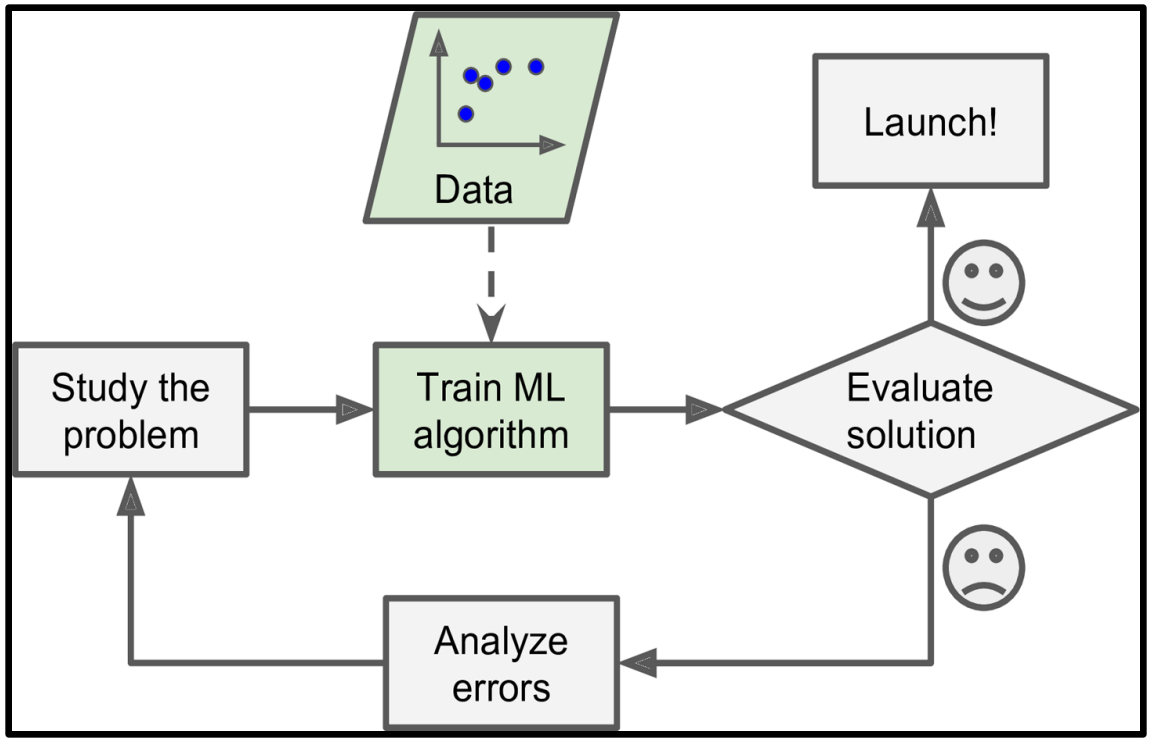
A computer program is said to **learn from experience E**

with respect to some **task T** and some **performance measure P**,
if its performance on T, as measured by P, improves with experience

What is Machine Learning?



Traditional Approach



Machine Learning Approach

Supervised & Unsupervised

Supervised & Unsupervised

Installing Sckit Learn

```
$ pip install numpy scipy matplotlib ipython scikit-learn pandas pillow
```

Numpy

Pandas

Matplotlib

Supervised & Unsupervised

Installing Sckit Learn

$ pip install numpy scipy matplotlib ipython scikit-learn pandas pillow

Numpy **Pandas** **Matplotlib**

```
In [21]: arr4 = np.array([[1,2,3],[4,5,6],[7,8,9],[10,11,12]])
         arr4

Out[21]: array([[ 1,  2,  3],
                [ 4,  5,  6],
                [ 7,  8,  9],
                [10, 11, 12]])
```

Supervised & Unsupervised

Installing Sckit Learn

```
$ pip install numpy scipy matplotlib ipython scikit-learn pandas pillow
```

Numpy

Pandas

Matplotlib

```
import pandas as pd
from IPython.display import display

# create a simple dataset of people
data = {'Name': ["John", "Anna", "Peter", "Linda"],
        'Location' : ["New York", "Paris", "Berlin", "London"],
        'Age' : [24, 13, 53, 33]
       }

data_pandas = pd.DataFrame(data)
# IPython.display allows "pretty printing" of dataframes
# in the Jupyter notebook
display(data_pandas)
```

	Age	Location	Name
0	24	New York	John
1	13	Paris	Anna
2	53	Berlin	Peter
3	33	London	Linda

Supervised & Unsupervised

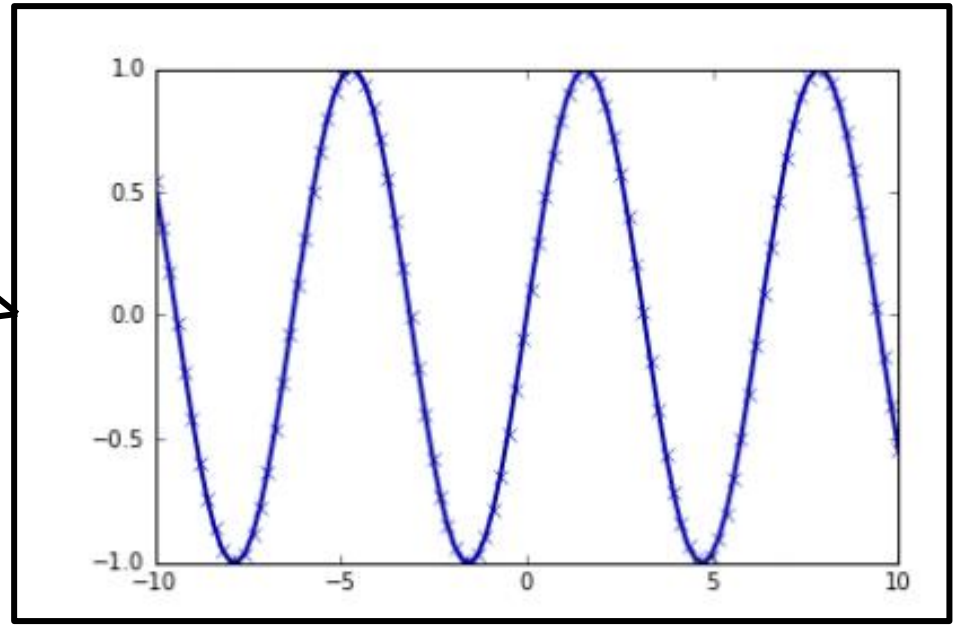
Installing Sckit Learn

$ pip install numpy scipy matplotlib ipython scikit-learn pandas pillow

Numpy **Pandas** **Matplotlib**

```
%matplotlib inline
import matplotlib.pyplot as plt

# Generate a sequence of numbers from -10 to 10 with 100 steps in between
x = np.linspace(-10, 10, 100)
# Create a second array using sine
y = np.sin(x)
# The plot function makes a line chart of one array against another
plt.plot(x, y, marker="x")
```



Supervised & Unsupervised

Supervised Learning

Given input, and we have examples of input / output pairs

Ex) 32 x 44 / 3 x 5 = 1408 / 15 -> 9 x 3 = 27

Classification

1. Binary Classification : Discrete valued output 0 or 1
2. Multiple Class Classification : apple, lemon, grape, etc …

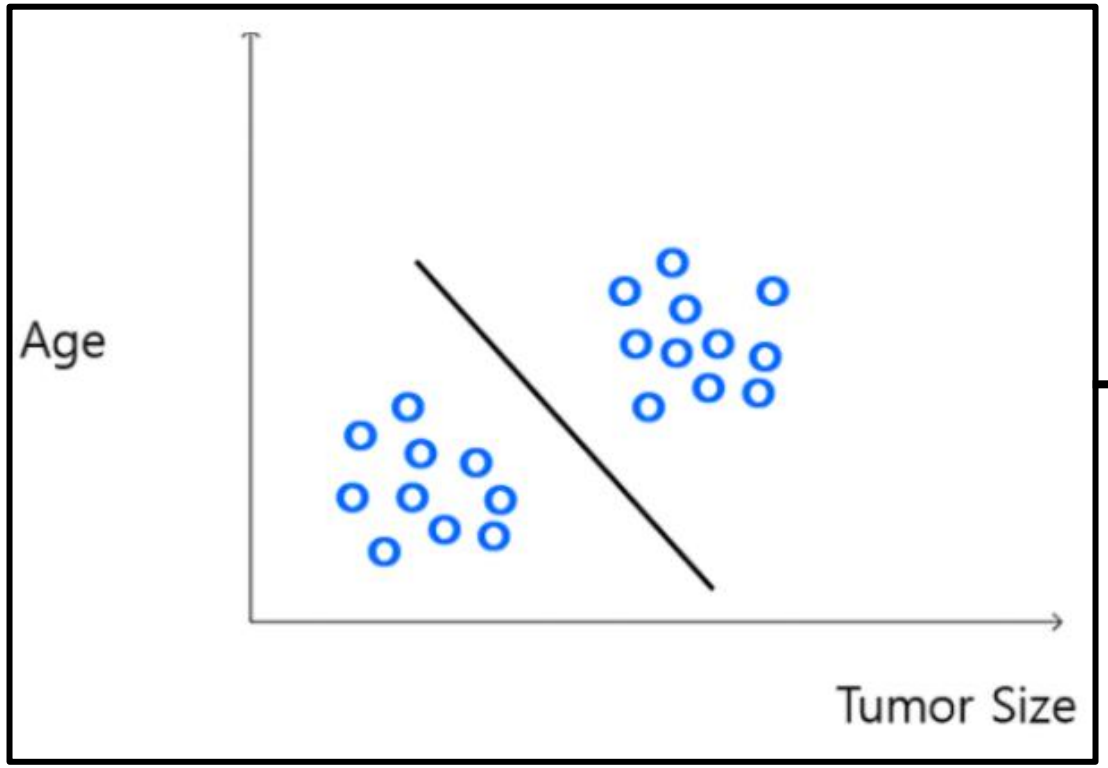
Regression

- Predict continuous valued out put
- Real number or an arbitrary number rather than 0 and 1(like Binary Classification)

Supervised & Unsupervised

Unsupervised Learning

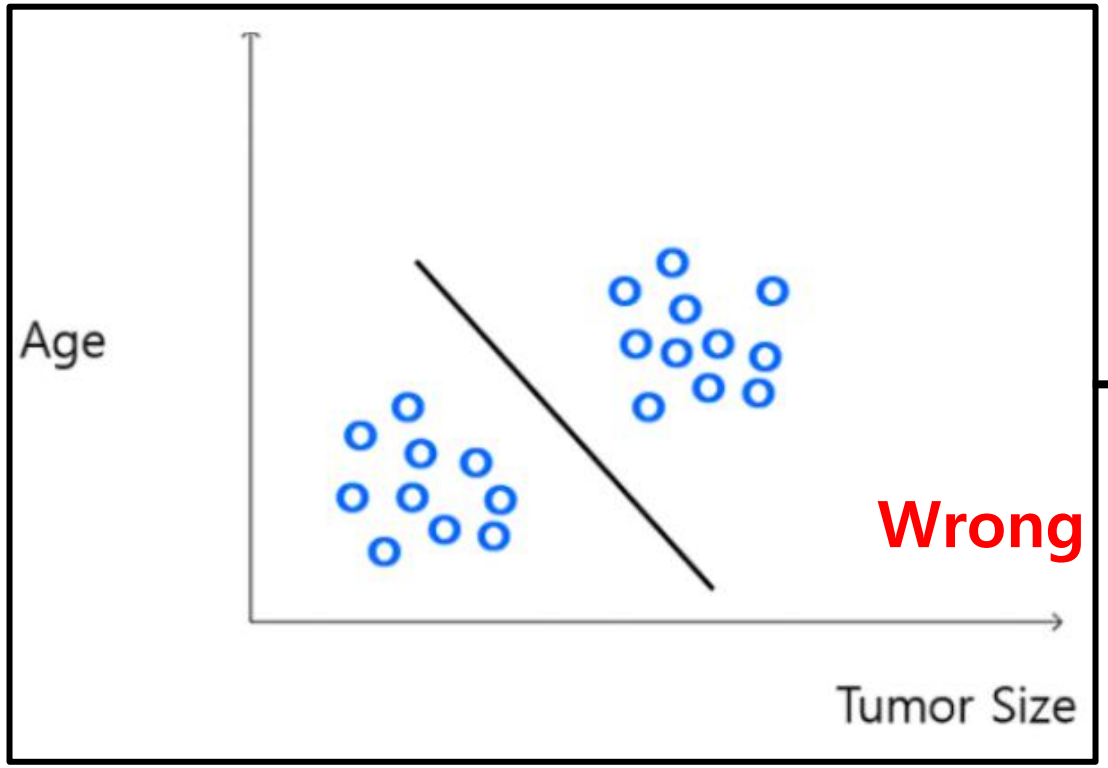
Input has not Right Answer for data set of output Not Given Data set Clustering Algorithm



Supervised & Unsupervised

Unsupervised Learning

Input has not Right Answer for data set of output Not Given Data set Clustering Algorithm

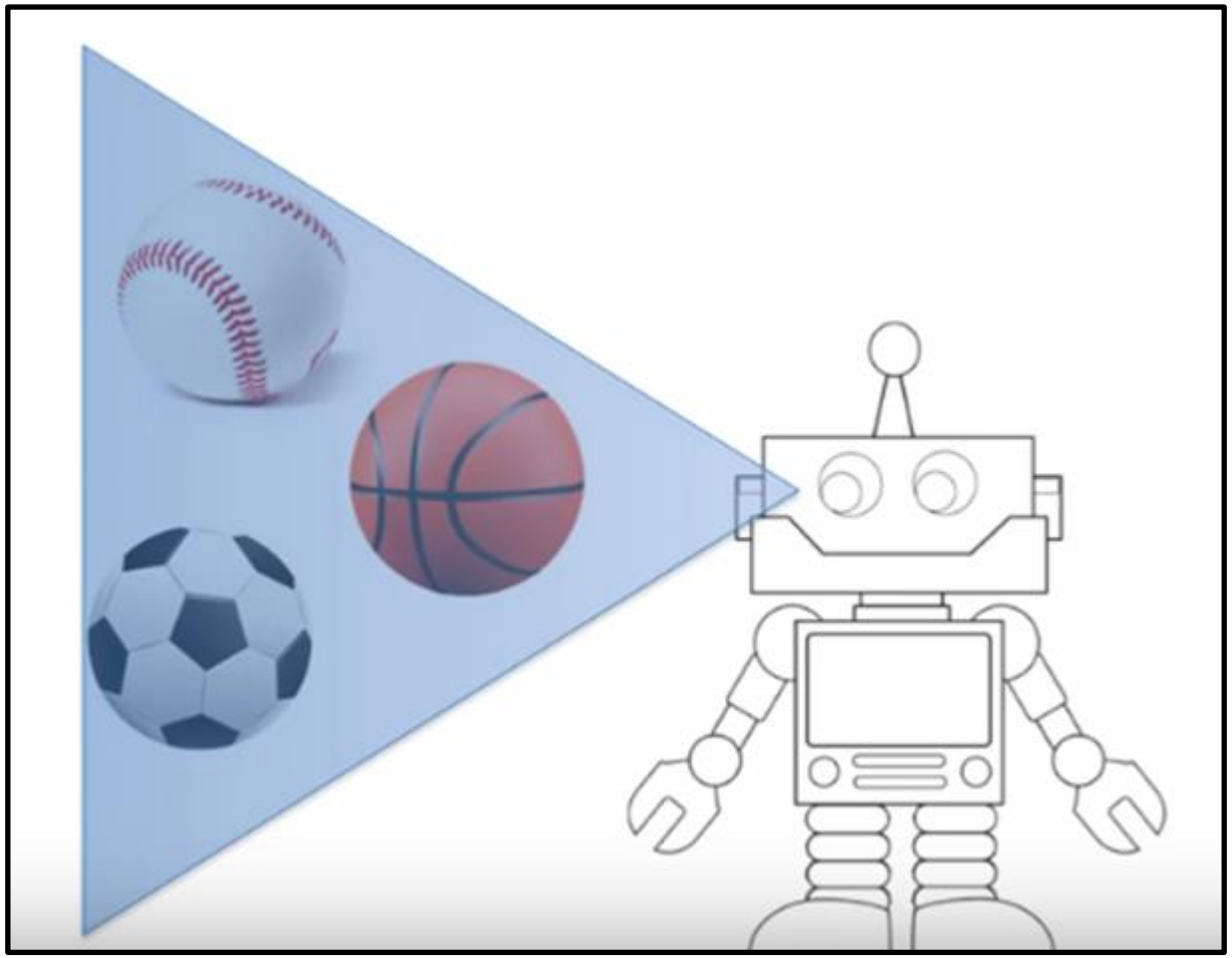


Wrong Prediction Result as Do not Feed Back

Problems

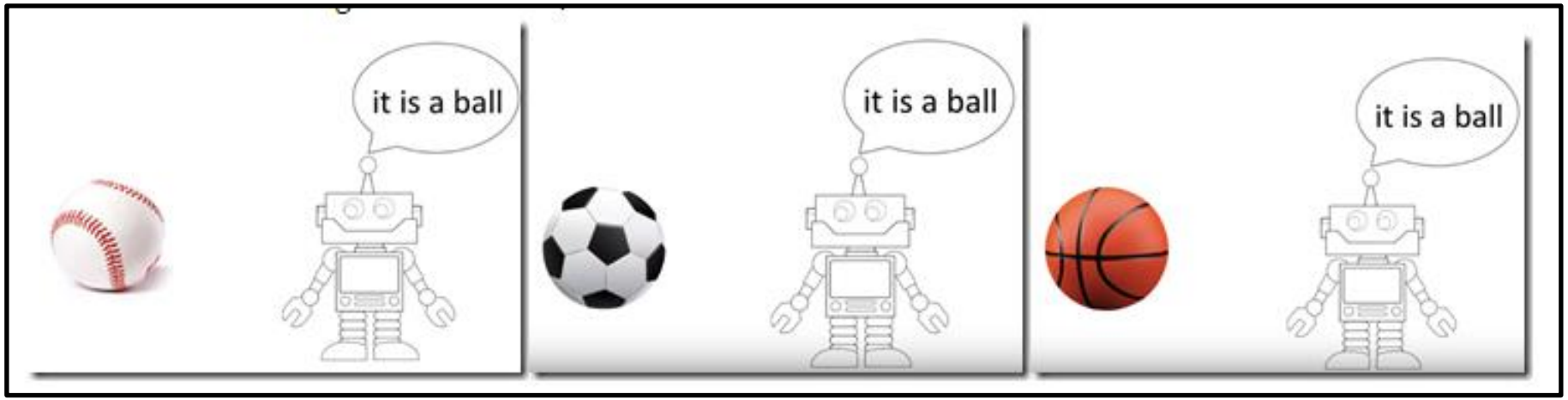
Problems

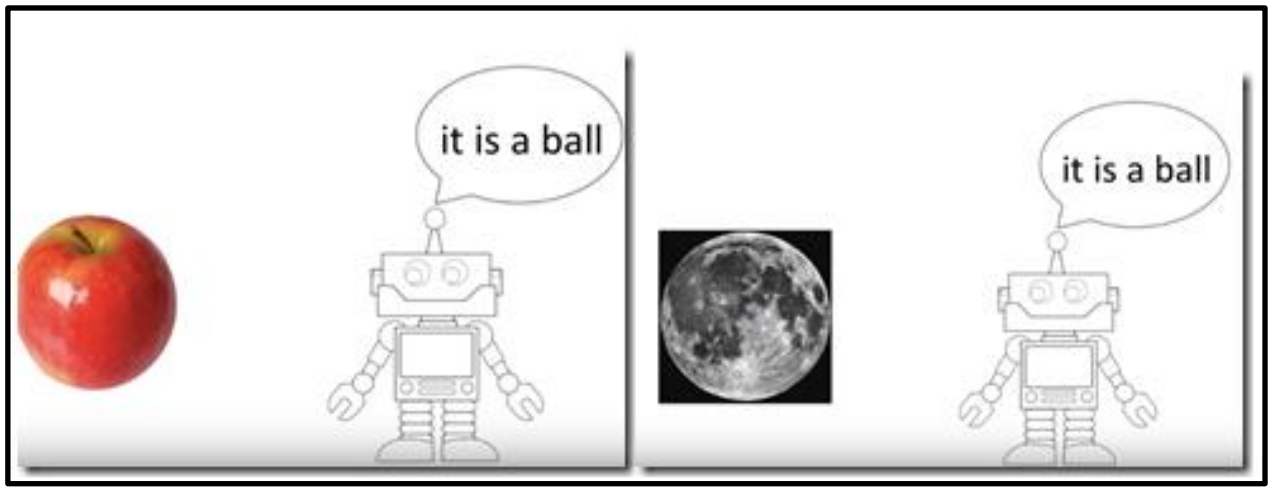
Underfitting



Problems

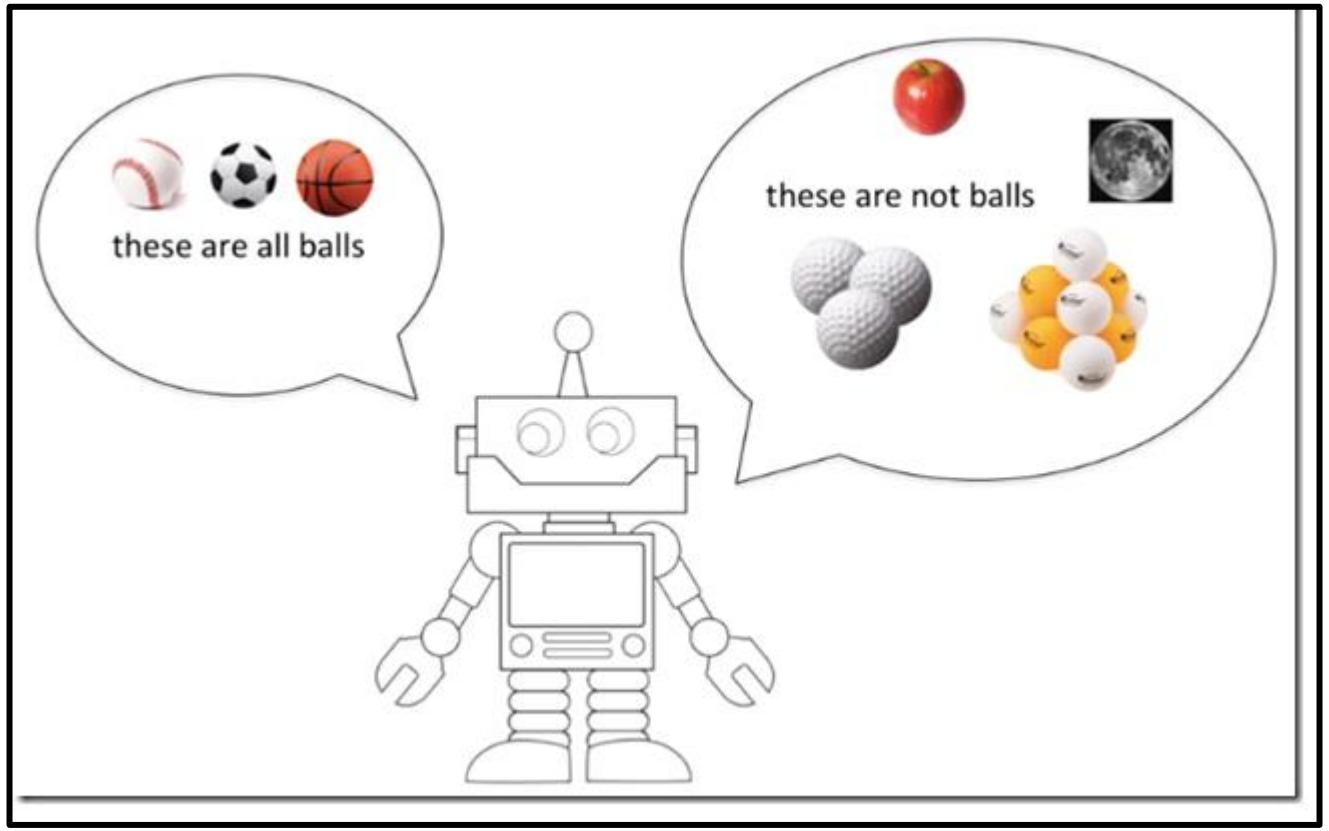
Underfitting





Problems

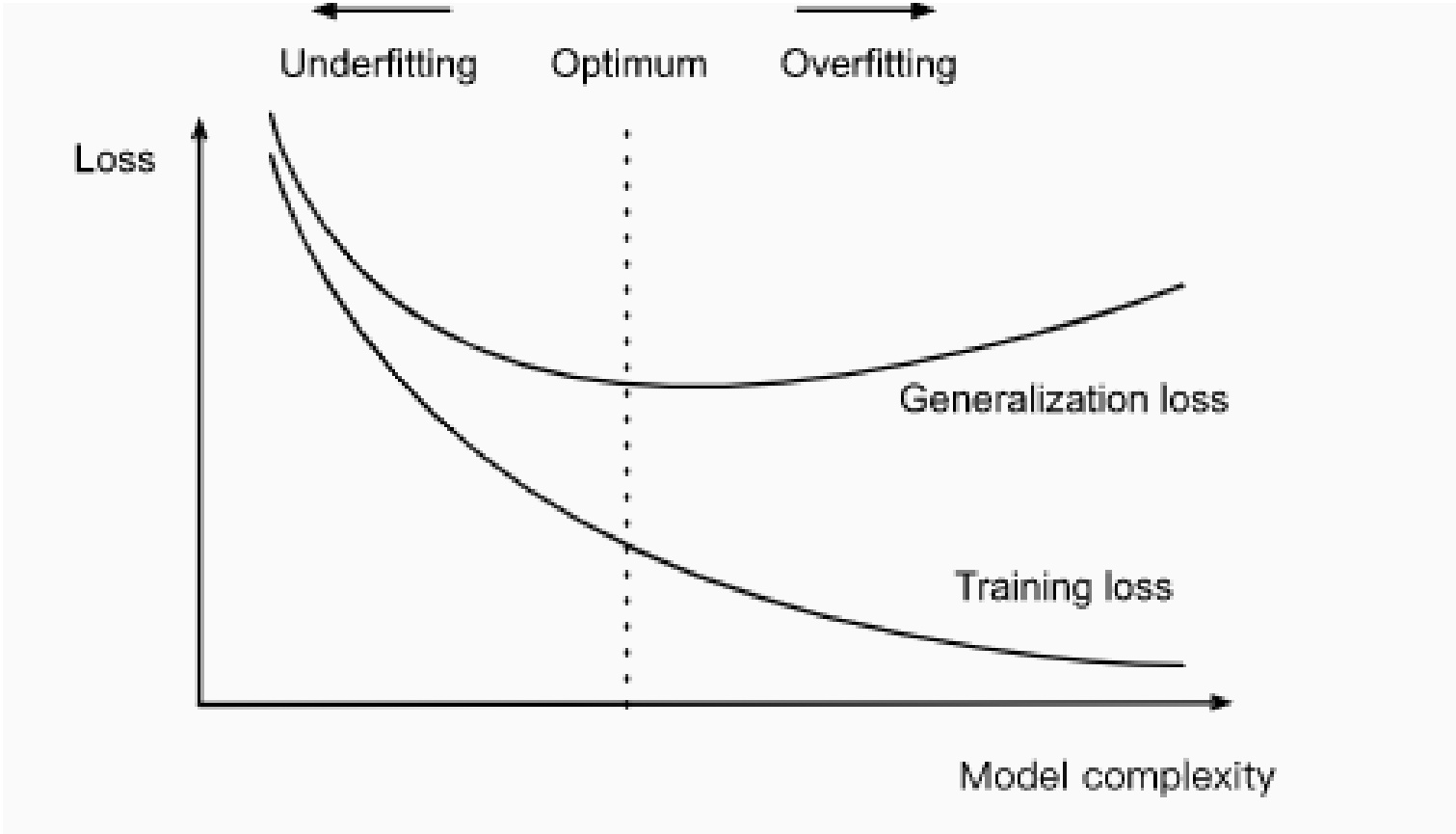
Overfitting



Problems

Solutions

$$\hat{y} = \sum_{i=0}^{d} x^i w_i$$



Problems

Solutions

$$y = 5 + 1.2x - 3.4\frac{x^2}{2!} + 5.6\frac{x^3}{3!} + \epsilon \text{ where } \epsilon \sim \mathcal{N}(0, 0.1)$$

$$(n! = \Gamma(n+1))$$

Impressions

Thank You