인공지능의 이해

Understanding Artificial Intelligence

유준수 (Joon Soo Yoo, Ph.D.)

Fall 2026

Defense & Advanced Security Lab (DAS Lab)
Department of Advanced Defense Technology and Industry
Jeonbuk National University

Introduction to Machine Learning

From Data to Prediction

Week 2

Learning Objectives

이번 Week에서는 Machine Learning의 가장 기본적인 구조를 학습한다.

Learning Goals

- Rule-based Program과 Machine Learning의 차이를 이해한다.
- Data, Feature, Label의 의미를 이해한다.
- Training과 Model의 의미를 이해한다.
- 새로운 Data에 대해 Prediction이 만들어지는 과정을 이해한다.
- Training Data와 Test Data를 구분할 수 있다.
- Regression과 Classification의 차이를 이해한다.
- Prediction Error와 Model Evaluation의 필요성을 이해한다.
- Overfitting의 기본 개념을 이해한다.

1. From Rules to Learning

지난 Week에서는 사람이 직접 Rule을 만들었다.

Source Code

```
if temperature >= 30:
    print("Hot")
else:
    print("Normal")
```

여기서 중요한 질문은 다음과 같았다.

Who decided “30”?

정답은 Human이다.

핵심

Rule-based Program에서는 사람이 문제를 해결하는 Rule을 직접 정의한다.

1-1. Can a Computer Learn the Rule?

이번에는 사람이 Rule을 직접 만들지 않는다고 생각해보자.

대신 Computer에게 많은 Example을 제공한다.

Study Hours	Exam Score
1	52
2	58
3	65
4	73
5	81

Computer가 이 Data에서 Pattern을 찾도록 한다.

$$\text{Data} \rightarrow \text{Learning} \rightarrow \text{Pattern}$$

핵심

Machine Learning에서는 사람이 모든 Rule을 직접 작성하는 대신, Computer가 Data에서 Pattern을 학습한다.

1-2. What is Machine Learning?

Machine Learning의 기본 구조는 다음과 같다.

Training Data

$\Downarrow$

Learning Algorithm

$\Downarrow$

Model

New Data $\rightarrow$ Model $\rightarrow$ Prediction

두 단계가 있다는 점이 중요하다.

- Training: Data로부터 Model을 만든다.
- Prediction: 만들어진 Model을 새로운 Data에 사용한다.

핵심

Machine Learning은 Data로부터 Model을 학습하고, 그 Model을 이용하여 새로운 Data에 대한 Prediction을 만든다.

2. Everything Starts with Data

Machine Learning은 Data에서 시작한다.

예를 들어 학생의 시험 점수를 예측하고 싶다고 하자.

Study Hours	Exam Score
1	52
2	58
3	65
4	73
5	81

우리는 이 Data에서 다음 관계를 찾고 싶다.

$$\text{Study Hours} \longrightarrow \text{Exam Score}$$

핵심

Machine Learning은 Data에 존재하는 관계와 Pattern을 찾는 과정에서 시작한다.

2-1. Feature and Label

Machine Learning에서는 Data의 역할을 구분한다.

Study Hours	Exam Score
1	52
2	58
3	65
4	73
5	81

여기서,

- Study Hours: Feature
- Exam Score: Label / Target

일반적으로 다음 기호를 많이 사용한다.

Feature: X

Label / Target: y

핵심

Feature는 Prediction에 사용하는 입력 정보이고, Label은 우리가 예측하고 싶은 정답이다.

2-2. More Than One Feature

실제 Machine Learning에서는 여러 Feature를 함께 사용할 수 있다.

예를 들어 집값을 예측한다고 하자.

Size	Age	Distance	Price
84	5	2.1	420
59	12	5.3	270
102	3	1.5	550

- Size: Feature
- Age: Feature
- Distance: Feature
- Price: Label / Target

Features $\longrightarrow$ Model $\longrightarrow$ Price

핵심

하나의 Prediction을 위해 여러 Feature를 동시에 사용할 수 있다.

3. What Does Training Mean?

Training은 Data를 이용하여 Model을 만드는 과정이다.

Training Data

⇓

Learning Algorithm

⇓

Find a Pattern

⇓

Model

Study Hours와 Exam Score 사이에 다음과 같은 Pattern이 있다고 생각해보자.

More Study Hours

⇓

Higher Exam Score

핵심

Training은 Data에 잘 맞는 Pattern 또는 관계를 Model이 찾도록 만드는 과정이다.

3-1. What is a Model?

Model은 Data에서 학습한 Pattern을 표현한 것이다.

예를 들어 다음과 같은 관계를 학습했다고 생각해보자.

$$\text{Score} \approx 7 \times \text{Study Hours} + 45$$

그러면 Study Hours가 6시간인 새로운 학생에 대해

$$7 \times 6 + 45 = 87$$

을 Prediction할 수 있다.

Study Hours = 6

⇓

Model

⇓

Predicted Score = 87

핵심

Model은 Training Data에서 학습한 Pattern을 이용하여 새로운 Data에 대한 Prediction을 만든다.

3-2. (참고) Linear Model

가장 간단한 Model 중 하나는 Linear Model이다.

하나의 Feature x를 사용하는 경우 다음과 같이 표현할 수 있다.

$$\hat{y} = wx + b$$

각 기호의 의미는 다음과 같다.

Symbol	Meaning
x	Input Feature
$\hat{y}$	Predicted Value
w	Weight / Slope
b	Bias / Intercept

예를 들어,

$$\hat{y} = 7x + 45$$

라면 $w = 7$, $b = 45$이다.

참고

Training은 Data에 잘 맞는 w와 b를 찾는 과정으로 생각할 수 있다.

4. From Training to Prediction

Training이 끝나면 Model을 새로운 Data에 사용할 수 있다.

Training Data

⇓

Training

⇓

Model

New Data

⇓

Model

⇓

Prediction

예를 들어 Training Data에는 없었던 6시간 공부한 학생의 점수를 Prediction할 수 있다.

핵심

Machine Learning의 목적은 Training Data를 외우는 것이 아니라, 새로운 Data에 대해 좋은 Prediction을 만드는 것이다.

4-1. Generalization

Machine Learning에서 매우 중요한 것은 새로운 Data에서도 잘 동작하는 것이다.

이를 Generalization이라고 한다.

Seen During Training

⇓

Model Learns Pattern

New, Unseen Data

⇓

Good Prediction

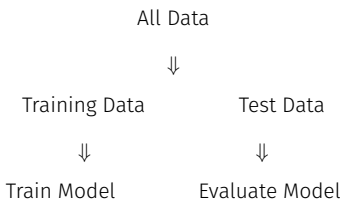
Training Data에서만 잘 동작하는 것은 충분하지 않다.

핵심

좋은 Model은 Training Data뿐 아니라 처음 보는 Data에서도 잘 동작해야 한다.

5. Training Data and Test Data

Model을 제대로 평가하려면 Training에 사용하지 않은 Data가 필요하다.



- Training Data: Model이 학습할 때 사용
- Test Data: 학습된 Model의 성능을 확인할 때 사용

핵심

Test Data는 Model이 Training 과정에서 미리 보지 않은 Data여야 한다.

5-1. Why Do We Need Test Data?

시험 문제를 미리 알고 답만 외운 학생을 생각해보자.

Known Questions

⇓

Memorize Answers

⇓

100 Points

하지만 새로운 문제가 나오면?

New Questions

⇓

Cannot Solve

⇓

Low Score

Machine Learning도 같은 문제가 발생할 수 있다.

핵심

Training Data의 성능만으로는 Model이 새로운 Data에서도 잘 동작하는지 알 수 없다.

5-2. Train-Test Split

실제 Machine Learning에서는 Data를 Training과 Test로 나누어 사용한다.

예를 들어 100개의 Data가 있다면,

100 Samples

⇓

80 Training Samples　　20 Test Samples

이를 흔히

80% Training / 20% Test

와 같이 표현한다.

항상 80:20을 사용해야 하는 것은 아니다.

핵심

Train-Test Split은 Model이 보지 않은 Data에서 얼마나 잘 동작하는지 확인하기 위한 방법이다.

6. Two Common Machine Learning Problems

Machine Learning에는 다양한 문제가 있지만, 먼저 두 가지 대표적인 문제를 살펴본다.

Regression

숫자 값을 Prediction한다.

예:

- Exam Score
- House Price
- Temperature
- Energy Consumption

Classification

Category를 Prediction한다.

예:

- Spam / Normal
- Cat / Dog
- Normal / Attack
- Disease / Normal

핵심

Regression은 숫자를 예측하고, Classification은 Category를 예측한다.

6-1. Regression

Regression은 연속적인 숫자 값을 Prediction하는 문제이다.

예를 들어 Study Hours로 Exam Score를 예측한다고 하자.

Study Hours

⇓

Regression Model

⇓

Exam Score

예를 들어,

6 Hours → Model → 87 Points

다른 예:

- House Information → House Price
- Weather Data → Temperature
- Sensor Data → Remaining Battery Time

핵심

Regression의 Output은 연속적인 Numerical Value이다.

6-2. Classification

Classification은 Data가 어떤 Category에 속하는지 Prediction하는 문제이다.

예를 들어 Email을 분류한다고 하자.

Email

⇓

Classification Model

⇓

Spam / Normal

다른 예:

- Image → Cat / Dog
- Transaction → Fraud / Normal
- Sensor Data → Normal / Attack
- Medical Image → Disease / Normal

핵심

Classification의 Output은 미리 정의된 Category 또는 Class이다.

6-3. Regression vs. Classification

Regression과 Classification을 비교해보자.

Problem	Input	Output
Exam Score	Study Hours	87.3
House Price	House Information	420 million
Spam Detection	Email	Spam
Image Recognition	Image	Cat
Attack Detection	Sensor Data	Attack

가장 먼저 물어볼 질문은 다음과 같다.

What do we want to predict?

- Number → Regression
- Category → Classification

핵심

Machine Learning 문제를 정의할 때 먼저 Prediction하려는 Output이 무엇인지 생각해야 한다.

7. Predictions Are Not Always Correct

Machine Learning Model의 Prediction은 항상 정확하지 않다.

예를 들어 실제 Exam Score가 80점인데 Model이 76점을 Prediction했다고 하자.

Actual Value

80

⇕

Prediction

76

Prediction과 실제 값 사이에는 차이가 존재한다.

Prediction Error = 4

핵심

Model의 성능을 평가하려면 Prediction과 실제 Answer가 얼마나 다른지 확인해야 한다.

7-1. Why Do We Need Evaluation?

두 개의 Model이 있다고 생각해보자.

실제 Score는 80점이다.

Model	Prediction	Error
Model A	78	2
Model B	65	15

어떤 Model이 더 좋은가?

이 경우 Model A의 Prediction이 실제 값에 더 가깝다.

하지만 실제로는 하나의 Data만 보고 Model을 평가하지 않는다.

핵심

여러 Test Data에 대한 Prediction Error를 계산하여 Model의 성능을 평가한다.

7-2. Evaluating Multiple Predictions

여러 Test Data에 대해 Prediction을 했다고 하자.

Actual	Prediction	Difference
50	52	2
60	57	3
70	74	4
80	79	1
90	85	5

Model이 어떤 Data에서는 조금 틀리고, 어떤 Data에서는 많이 틀릴 수 있다.

따라서 여러 Error를 하나의 숫자로 요약하는 방법이 필요하다.

핵심

Evaluation Metric은 여러 Prediction의 성능을 하나의 숫자로 요약하는 방법이다.

7-3. Mean Absolute Error (MAE)

Regression에서 사용할 수 있는 간단한 Evaluation Metric이 MAE이다.

MAE는 Prediction이 실제 값에서 평균적으로 얼마나 떨어져 있는지를 나타낸다.

앞의 Error가 다음과 같았다.

$$2, \quad 3, \quad 4, \quad 1, \quad 5$$

평균을 계산하면,

$$\frac{2+3+4+1+5}{5} = 3$$

이다.

따라서 MAE는 3이다.

핵심

MAE가 작을수록 Prediction이 실제 값에 평균적으로 더 가깝다.

7-4. (참고) MAE Formula

MAE는 Mean Absolute Error의 약자이다.

수식으로는 다음과 같이 표현한다.

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$$

각 기호의 의미는 다음과 같다.

Symbol	Meaning
n	Number of Data Samples
y_i	Actual Value
$\hat{y}_i$	Predicted Value
$\lvert y_i - \hat{y}_i \rvert$	Absolute Prediction Error

Absolute Value를 사용하는 이유는 양수와 음수 Error가 서로 상쇄되지 않도록 하기 위해서이다.

참고

MAE는 각 Prediction의 Absolute Error를 계산한 뒤 그 값들의 평균을 구한다.

7-5. (참고) Why Absolute Value?

Absolute Value가 없다면 어떤 문제가 생길까?

두 Prediction을 생각해보자.

Actual	Prediction	$y - \hat{y}$
80	75	+5
80	85	-5

그냥 평균을 계산하면,

$$\frac{5 + (-5)}{2} = 0$$

이다.

하지만 두 Prediction 모두 5점씩 틀렸다.

Absolute Value를 사용하면,

$$\frac{|5| + |-5|}{2} = 5$$

가 된다.

참고

Prediction Error의 방향보다 Error의 크기를 평가하기 위해 Absolute Value를 사용한다.

8. How Do We Evaluate Classification?

Classification에서는 Error의 의미가 조금 다르다.

예를 들어 실제 Answer와 Prediction을 비교한다.

Actual	Prediction	Result
Normal	Normal	Correct
Spam	Spam	Correct
Normal	Spam	Wrong
Spam	Spam	Correct

가장 간단한 방법은 전체 중 몇 개를 맞혔는지 계산하는 것이다.

$$\text{Correct Predictions} \div \text{Total Predictions}$$

핵심

Classification에서는 Accuracy와 같은 별도의 Evaluation Metric을 사용한다.

8-1. Accuracy

Accuracy는 전체 Prediction 중 정답을 맞힌 비율이다.

예를 들어 100개의 Email을 분류했다고 하자.

100 Emails

⇓

90 Correct

10 Wrong

그러면 Accuracy는

$$\frac{90}{100} = 0.9 = 90\%$$

이다.

핵심

Accuracy는 Classification Model의 성능을 가장 직관적으로 표현하는 방법 중 하나이다.

8-2. Is Accuracy Always Enough?

Accuracy가 높으면 항상 좋은 Model일까?

100개의 Data가 있다고 생각해보자.

- Normal: 99
- Attack: 1

Model이 모든 Data를 Normal이라고 Prediction하면?

99 Correct

1 Wrong

Accuracy = 99%

하지만 Attack은 하나도 찾지 못했다.

중요

높은 Accuracy가 항상 좋은 Model을 의미하지는 않는다. Data의 특성과 문제의 목적을 함께 고려해야 한다.

8-3. Other Classification Metrics

Classification에서는 Accuracy 이외에도 다양한 Evaluation Metric을 사용한다.

대표적인 Metric은 다음과 같다.

- Precision
- Recall
- F1 Score
- ROC Curve
- AUC

예를 들어 Attack Detection에서는

How many attacks did we detect?

How many alarms were actually attacks?

같은 질문이 중요할 수 있다.

Next

Classification Evaluation은 다음 Week에서 Confusion Matrix와 함께 자세히 살펴본다.

9. What is Overfitting?

Model이 Training Data에서는 매우 잘 동작하지만, 새로운 Data에서는 잘 동작하지 않을 수 있다.

이를 Overfitting이라고 한다.

Training Data

Very Good Performance

⇓

New Test Data

Poor Performance

핵심

Overfitting은 Model이 Training Data에 지나치게 맞춰져 새로운 Data에 Generalize하지 못하는 현상이다.

9-1. Memorization vs. Understanding

시험을 준비하는 두 학생을 생각해보자.

Student A

기출문제의 답을 그대로 외운다.

- 기출문제: 100점
- 새로운 문제: 40점

Memorization

Student B

문제를 해결하는 원리를 이해한다.

- 기출문제: 90점
- 새로운 문제: 85점

Understanding

Machine Learning에서도 비슷한 문제가 발생한다.

핵심

Training Data를 단순히 잘 맞추는 것보다, 새로운 Data에서도 잘 동작하는 것이 중요하다.

9-2. Training Performance vs. Test Performance

Model의 성능은 Training Data와 Test Data에서 각각 확인할 수 있다.

Model	Training Error	Test Error
Model A	Small	Small
Model B	Very Small	Large

Model B는 Training Data를 매우 잘 맞추지만, 새로운 Test Data에서는 큰 Error를 보인다.

Very Low Training Error

+

High Test Error

⇓

Possible Overfitting

핵심

Training 성능만 보는 것이 아니라 Test Data에서의 성능을 함께 확인해야 한다.

9-3. What is Underfitting?

반대의 문제도 발생할 수 있다.

Model이 너무 단순하여 Data의 중요한 Pattern조차 제대로 학습하지 못할 수 있다.

Training Data

⇓

Model Cannot Learn the Pattern

⇓

Poor Training Performance

+

Poor Test Performance

이를 Underfitting이라고 한다.

핵심

Underfitting은 Model이 너무 단순하거나 충분히 학습하지 못해 Training Data의 Pattern도 제대로 설명하지 못하는 현상이다.

9-4. Underfitting vs. Overfitting

Model은 너무 단순해도, 너무 Training Data에 맞춰져도 문제가 된다.

Case	Training	Test
Underfitting	Poor	Poor
Good Fit	Good	Good
Overfitting	Very Good	Poor

우리가 원하는 것은

Good Performance on New Data

이다.

핵심

Machine Learning의 목표는 Training Data를 완벽하게 외우는 것이 아니라, 새로운 Data에서도 잘 동작하는 Model을 만드는 것이다.

9-5. (참고) Model Complexity

Overfitting과 Underfitting은 Model Complexity와 관련될 수 있다.

Too Simple

⇓

Underfitting

Appropriate Complexity

⇓

Good Generalization

Too Complex

⇓

Possible Overfitting

복잡한 Model이 항상 더 좋은 것은 아니다.

참고

Model은 Training Data의 중요한 Pattern을 학습하면서도 불필요한 Noise까지 외우지 않아야 한다.

10. The Machine Learning Workflow

지금까지 배운 내용을 하나의 과정으로 연결해보자.

Data

⇓

Features / Labels

⇓

Train-Test Split

⇓

Training

⇓

Model

⇓

Prediction

⇓

Evaluation

이 구조는 앞으로 다양한 Machine Learning Model을 배울 때 계속 반복된다.

핵심

Model이 달라져도 Machine Learning의 기본 Workflow는 크게 달라지지 않는다.

10-1. Step 1: What Do We Want to Predict?

Machine Learning을 시작하기 전에 먼저 문제를 정의해야 한다.

예를 들어,

Problem	Prediction Target
Exam Score Prediction	Score
House Price Prediction	Price
Spam Detection	Spam / Normal
Attack Detection	Attack / Normal

가장 먼저 생각할 질문은

What do we want to predict?

이다.

핵심

Machine Learning은 Algorithm을 먼저 선택하는 것이 아니라, 해결하려는 Problem을 먼저 정의하는 것에서 시작한다.

10-2. Step 2: Prepare the Data

문제를 정의했다면 Model이 사용할 Data를 준비한다.

예를 들어 Exam Score Prediction에서는

Feature	Label
Study Hours	Exam Score

그리고 Data를 나눈다.

All Data

⇓

Training Data　　Test Data

핵심

어떤 Data를 Feature와 Label로 사용할지 결정하고, Training과 Test Data를 준비한다.

10-3. Step 3: Train the Model

Training Data를 이용하여 Model을 학습한다.

Training Features X

+

Training Labels y

⇓

Learning Algorithm

⇓

Trained Model

Python에서는 이후 다음과 같은 코드를 사용하게 된다.

Source Code

```
model.fit(X_train, y_train)
```

핵심

*fit()*은 Training Data를 이용하여 Model을 학습시키는 과정이다.

10-4. Step 4: Make Predictions

Training이 끝난 Model에 새로운 Data를 입력한다.

New Feature

⇓

Trained Model

⇓

Prediction

Python에서는 다음과 같은 코드를 사용한다.

Source Code

```
prediction = model.predict(X_test)
```

핵심

*predict()*는 학습된 Model을 이용하여 새로운 Data에 대한 Prediction을 만든다.

10-5. Step 5: Evaluate the Model

마지막으로 Prediction과 실제 Answer를 비교한다.

Prediction

vs.

Actual Answer

⇓

Evaluation Metric

⇓

Model Performance

문제에 따라 서로 다른 Metric을 사용할 수 있다.

- Regression: MAE 등
- Classification: Accuracy, Precision, Recall, F1 등

핵심

Machine Learning은 Model을 Training하는 것으로 끝나지 않는다. 새로운 Data에서 Model을 평가하는 과정이 반드시 필요하다.

11. One Complete Example

Exam Score Prediction 문제를 처음부터 끝까지 다시 살펴보자.

Study Hours and Exam Scores

⇓

Feature: Study Hours

Label: Exam Score

⇓

Train-Test Split

⇓

Regression Model Training

⇓

Predict Test Scores

⇓

Compare with Actual Scores

⇓

Calculate Error

핵심

Data에서 시작하여 Training, Prediction, Evaluation까지 하나의 Machine Learning Pipeline을 구성한다.

11-1. What Does the Computer Actually Learn?

Machine Learning을 “Computer가 스스로 생각한다”고 이해할 필요는 없다.

이번 Regression Example에서는 Computer가 Data에 잘 맞는 관계를 찾는다.

Training Data

⇓

Find Parameters

⇓

Model

⇓

Prediction

예를 들어 Linear Model에서는 직선의 기울기와 위치를 결정하는 값을 찾는다.

핵심

Learning은 마법이 아니라, Data에 잘 맞는 Model의 Parameter를 찾는 과정으로 볼 수 있다.

11-2. (참고) Learning as Optimization

조금 더 수학적으로 보면 Training은 좋은 Parameter를 찾는 Optimization 문제이다.

Model의 Prediction을

$$\hat{y} = f(x; \theta)$$

라고 하자.

여기서 θ는 Model의 Parameter이다.

Prediction과 실제 값의 차이를 Loss Function L로 측정할 수 있다.

$$L(y, \hat{y})$$

Training의 목표는 전체 Loss가 작아지도록 Parameter θ를 찾는 것이다.

$$\theta^* = \arg\min_{\theta} \sum_{i=1}^{n} L(y_i, f(x_i; \theta))$$

참고

Machine Learning의 Training은 Prediction Error를 작게 만드는 Parameter를 찾는 Optimization 문제로 표현할 수 있다.

11-3. (참고) Linear Regression and Squared Error

Linear Regression에서는 Model을 다음과 같이 둘 수 있다.

$$\hat{y}_i = wx_i + b$$

실제 값 y_i와 Prediction $\hat{y}_i$ 사이의 Error를 제곱하여 측정할 수 있다.

$$(y_i - \hat{y}_i)^2$$

전체 Data에 대해서는

$$\sum_{i=1}^{n} (y_i - (wx_i + b))^2$$

를 계산할 수 있다.

Linear Regression은 이 값이 작아지는 w와 b를 찾는다.

참고

Linear Regression은 Data와 직선 사이의 Squared Error가 작아지도록 직선을 찾는 방법으로 이해할 수 있다.

12. What We Learned

이번 Week에서는 Machine Learning의 기본 구조를 살펴보았다.

- Machine Learning은 Data에서 Pattern을 학습한다.
- Feature는 Model의 Input이다.
- Label은 Prediction하고 싶은 Target이다.
- Training은 Data를 이용하여 Model을 만드는 과정이다.
- Prediction은 학습된 Model을 새로운 Data에 사용하는 과정이다.
- Training Data와 Test Data는 역할이 다르다.
- Regression은 Number를 Prediction한다.
- Classification은 Category를 Prediction한다.
- Evaluation Metric으로 Model의 성능을 측정한다.
- 새로운 Data에서 잘 동작하는 것이 중요하다.

핵심

Machine Learning의 목적은 Training Data를 외우는 것이 아니라, Data의 Pattern을 학습하여 새로운 Data에 대해 좋은 Prediction을 만드는 것이다.

12-1. The Core Machine Learning Pipeline

앞으로 계속 사용하게 될 Machine Learning의 기본 Pipeline을 기억하자.

Data

⇓

Feature X / Label y

⇓

Train-Test Split

⇓

Training

⇓

Model

⇓

Prediction

⇓

Evaluation

핵심

앞으로 Model과 Problem이 달라져도 이 기본적인 Machine Learning Pipeline은 계속 반복된다.

12-2. Three Important Questions

Machine Learning 문제를 보면 먼저 세 가지 질문을 생각해보자.

1. **What is the Input?**
   어떤 Feature를 Model에 입력할 것인가?
2. **What do we want to predict?**
   어떤 Label 또는 Target을 Prediction할 것인가?
3. **How do we evaluate the prediction?**
   Model이 잘 동작하는지 어떻게 측정할 것인가?

핵심

Input, Prediction Target, Evaluation을 정의하는 것이 Machine Learning 문제를 이해하는 출발점이다.

Thank You

Questions?