

5주차(1/3)

기계학습 작업 흐름 1

파이썬으로 배우는 기계학습

한동대학교
김영섭 교수

기계학습 작업 흐름 1

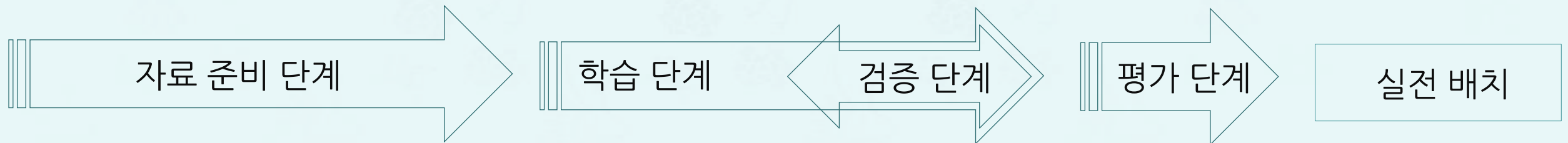
- 학습 목표
 - 기계학습의 전체 과정을 이해하여 단계별 작업에 필요한 흐름을 이해한다.
- 학습 내용
 - 기계학습 작업 과정에 대한 이해
 - 학습 자료 준비
 - 학습 자료 읽기
 - 학습 자료에서 노이즈

1. 기계학습 작업 흐름: 전체 단계

- 작업 흐름(workflow)

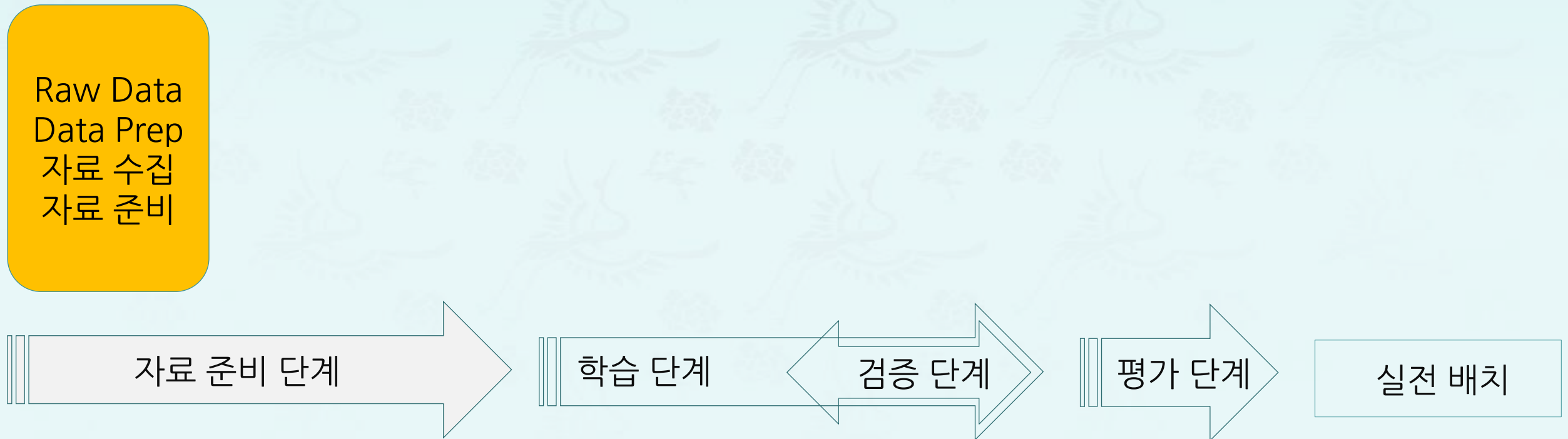
1. 기계학습 작업 흐름: 전체 단계

- 작업 흐름(workflow)



1. 기계학습 작업 흐름: 자료 준비

1. 자료 준비 단계: 자료 수집과 전처리

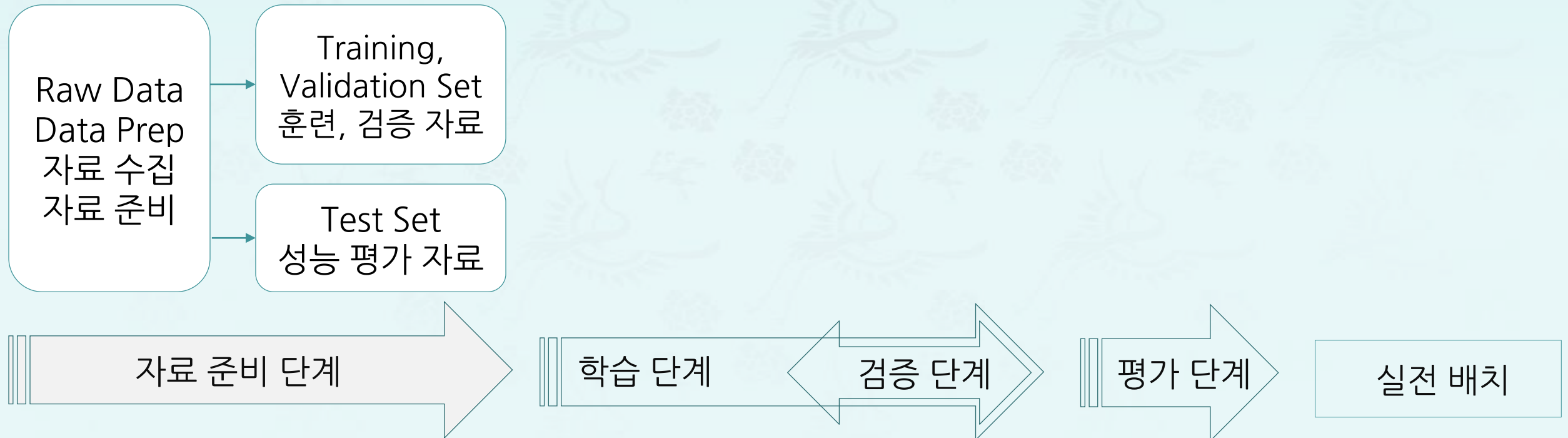


1. 기계학습 작업 흐름: 자료 준비

1. 자료 준비 단계: 수집과 전처리

■ 학습 자료 세트

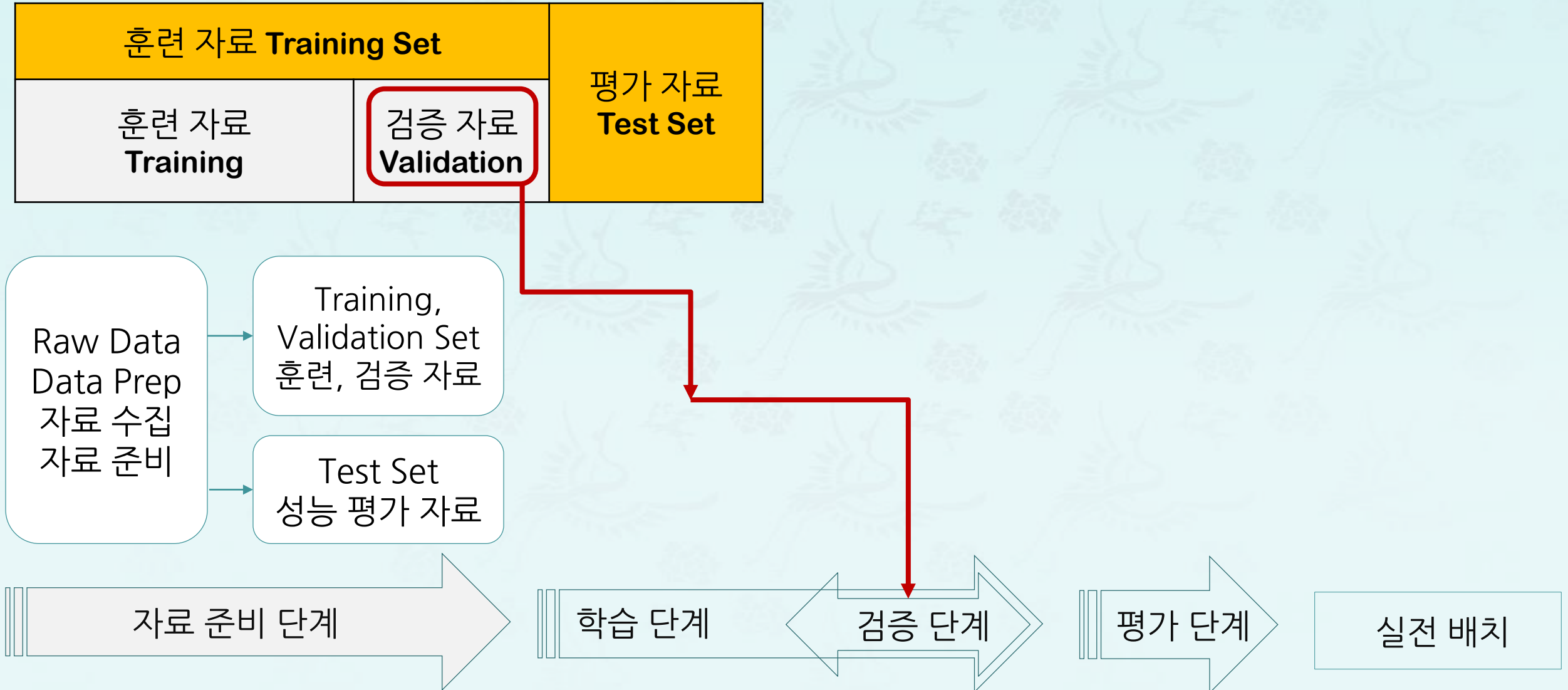
훈련 자료 Training Set		평가 자료 Test Set
훈련 자료 Training	검증 자료 Validation	



1. 기계학습 작업 흐름: 자료 준비

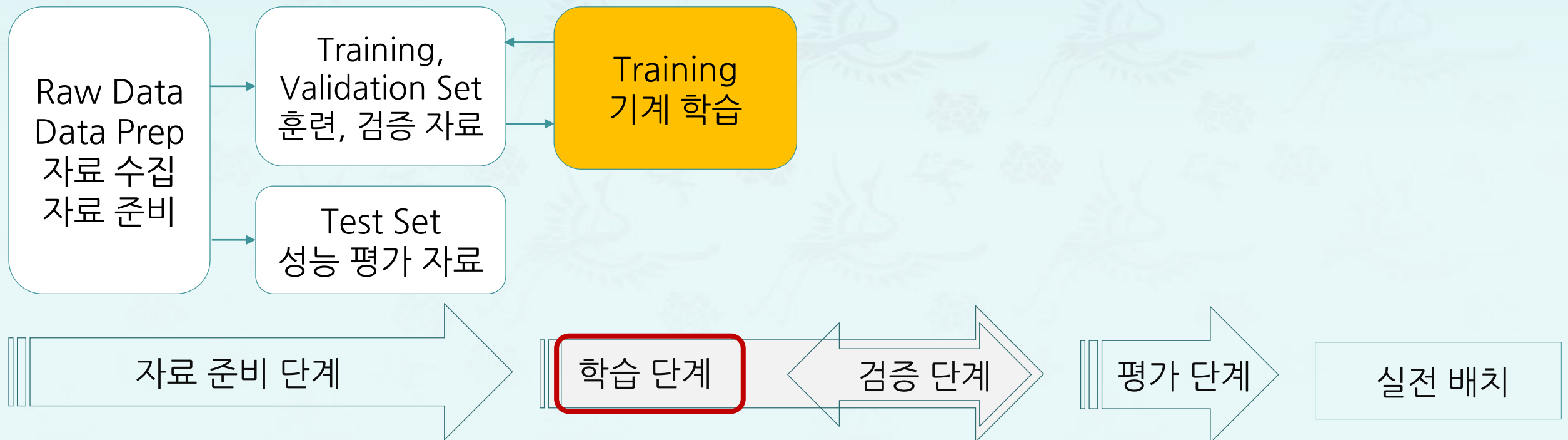
1. 자료 준비 단계: 수집과 전처리

■ 학습 자료 세트



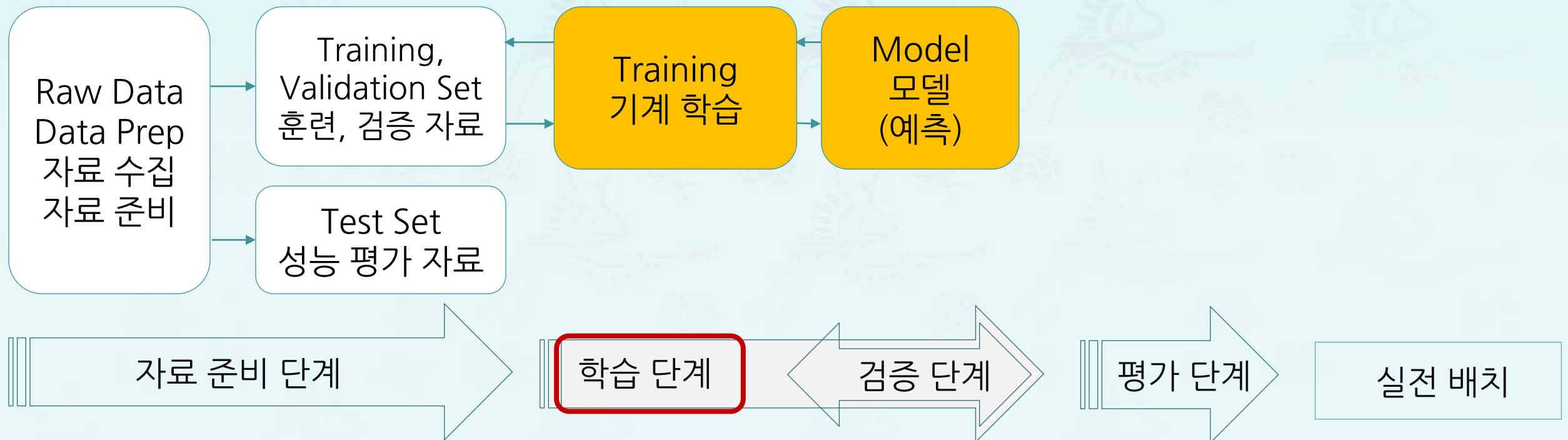
1. 기계학습 작업 흐름: 학습

1. 자료 준비 단계: 수집과 전처리
2. 학습 단계: 모델 훈련과 완성



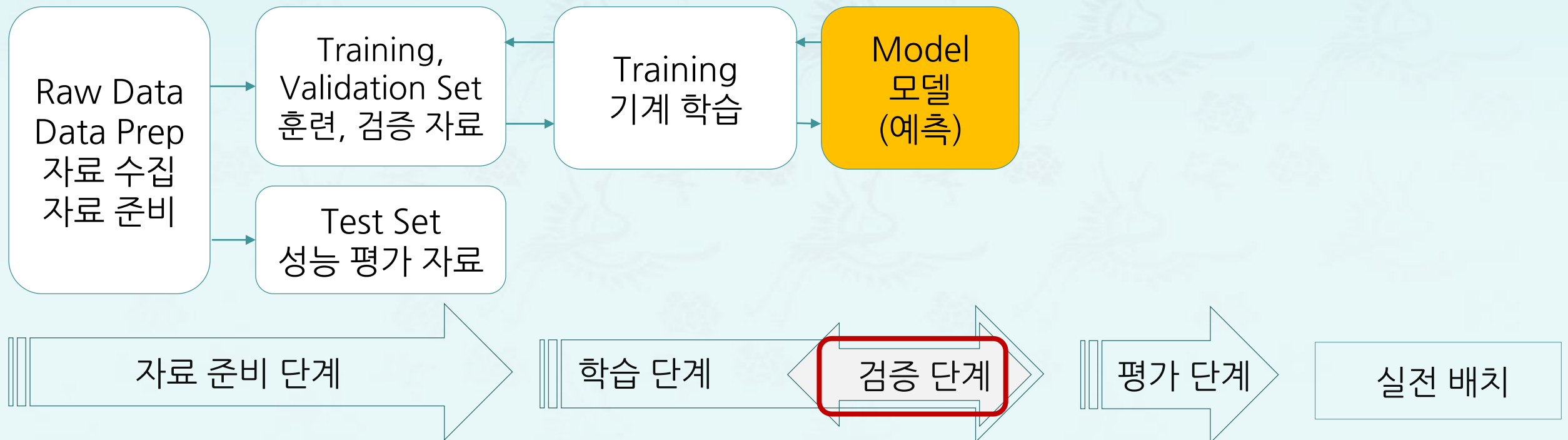
1. 기계학습 작업 흐름: 학습

1. 자료 준비 단계: 수집과 전처리
2. 학습 단계: 모델 훈련과 완성



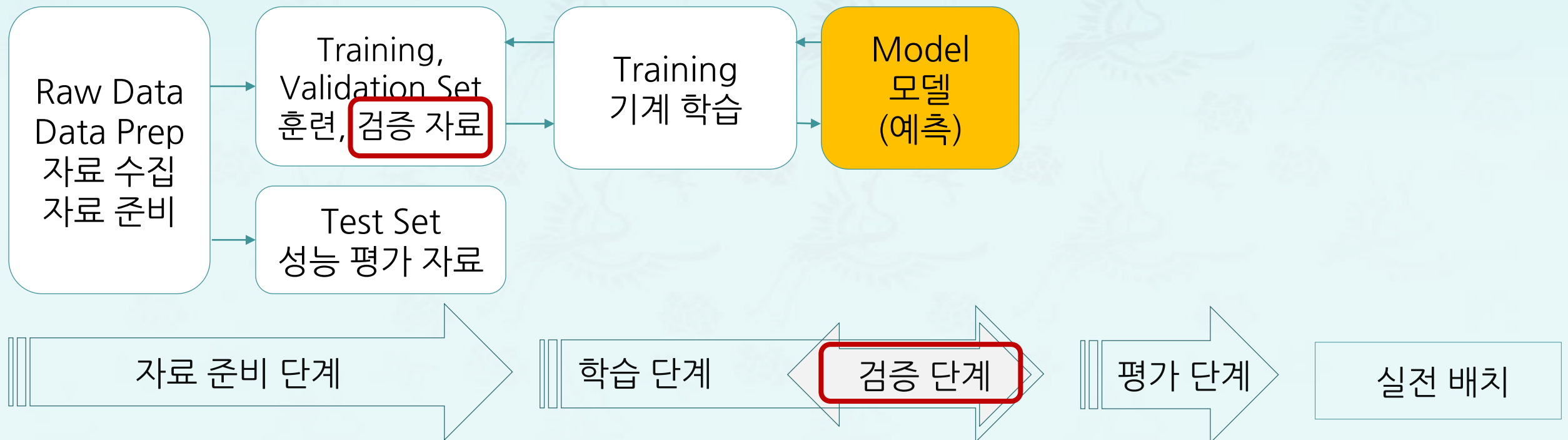
1. 기계학습 작업 흐름: 검증

1. 자료 준비 단계: 수집과 전처리
2. 학습 단계: 모델 훈련과 완성
3. 검증 단계: 하이퍼 파라미터 조정



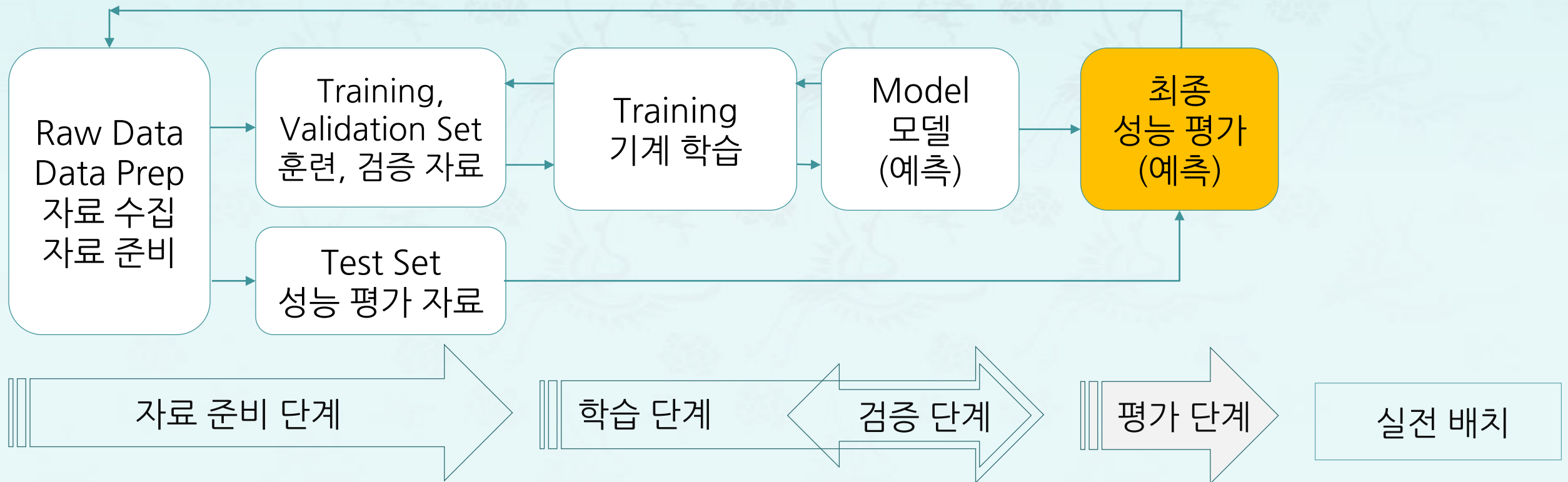
1. 기계학습 작업 흐름: 검증

1. 자료 준비 단계: 수집과 전처리
2. 학습 단계: 모델 훈련과 완성
3. 검증 단계: 하이퍼 파라미터 조정



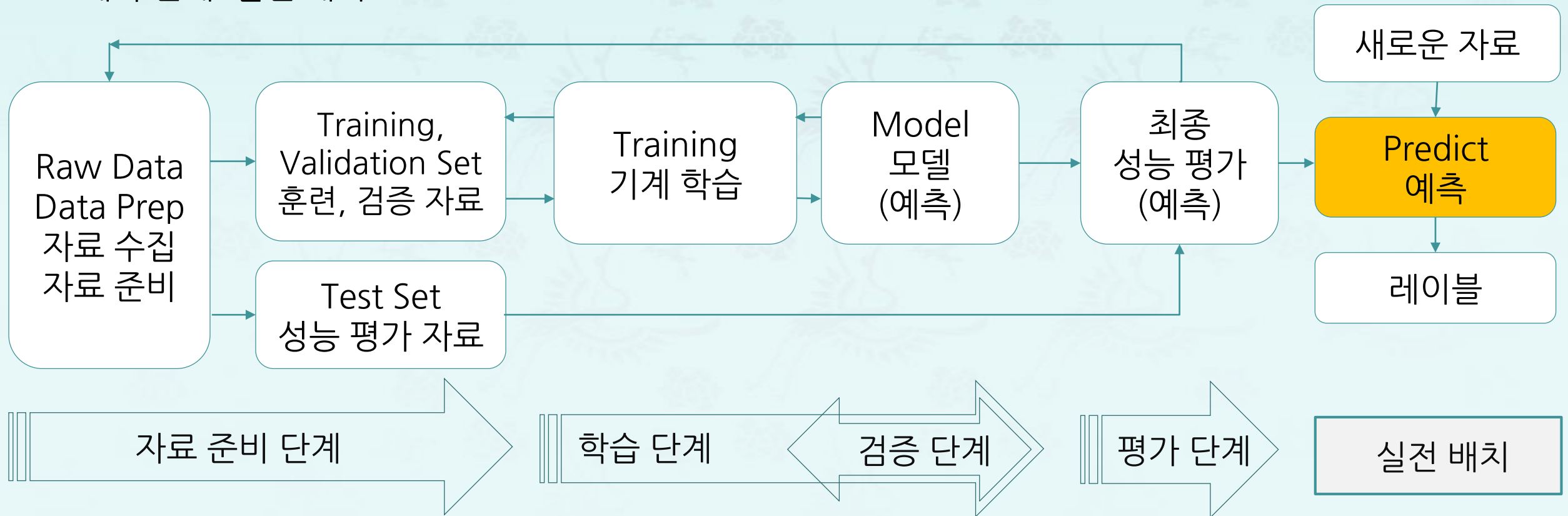
1. 기계학습 작업 흐름: 평가

1. 자료 준비 단계: 수집과 전처리
2. 학습 단계: 모델 훈련과 완성
3. 검증 단계: 하이퍼 파라미터 조정
4. 평가 단계: 최종 실전 배치 여부 결정



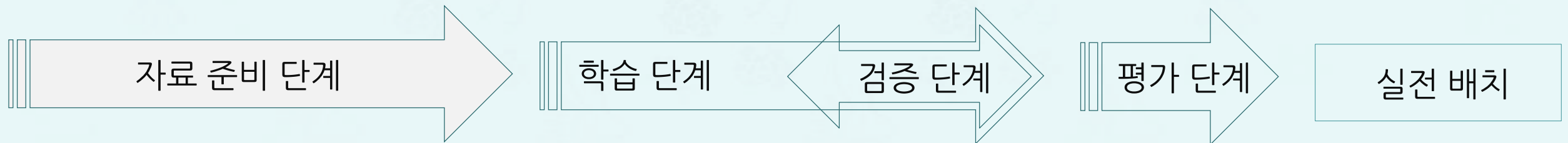
1. 기계학습 작업 흐름: 예측

1. 자료 준비 단계: 수집과 전처리
2. 학습 단계: 모델 훈련과 완성
3. 검증 단계: 하이퍼 파라미터 조정
4. 평가 단계: 최종 실전 배치 여부 결정
5. 예측 단계: 실전 배치



2. 학습 자료의 준비 단계: 자료 수집

- 학습 자료 준비



2. 학습 자료의 준비 단계: 자료 수집

- 학습 자료 준비
 - **joydata.txt**

2. 학습 자료의 준비 단계: 자료 내용

- 학습 자료 준비
 - joydata.txt

```
!cat data/joydata.txt
```

-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

2. 학습 자료의 준비 단계: 표기법

- 학습 자료 준비
 - **joydata.txt**
- 표기법

```
!cat data/joydata.txt
```

-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

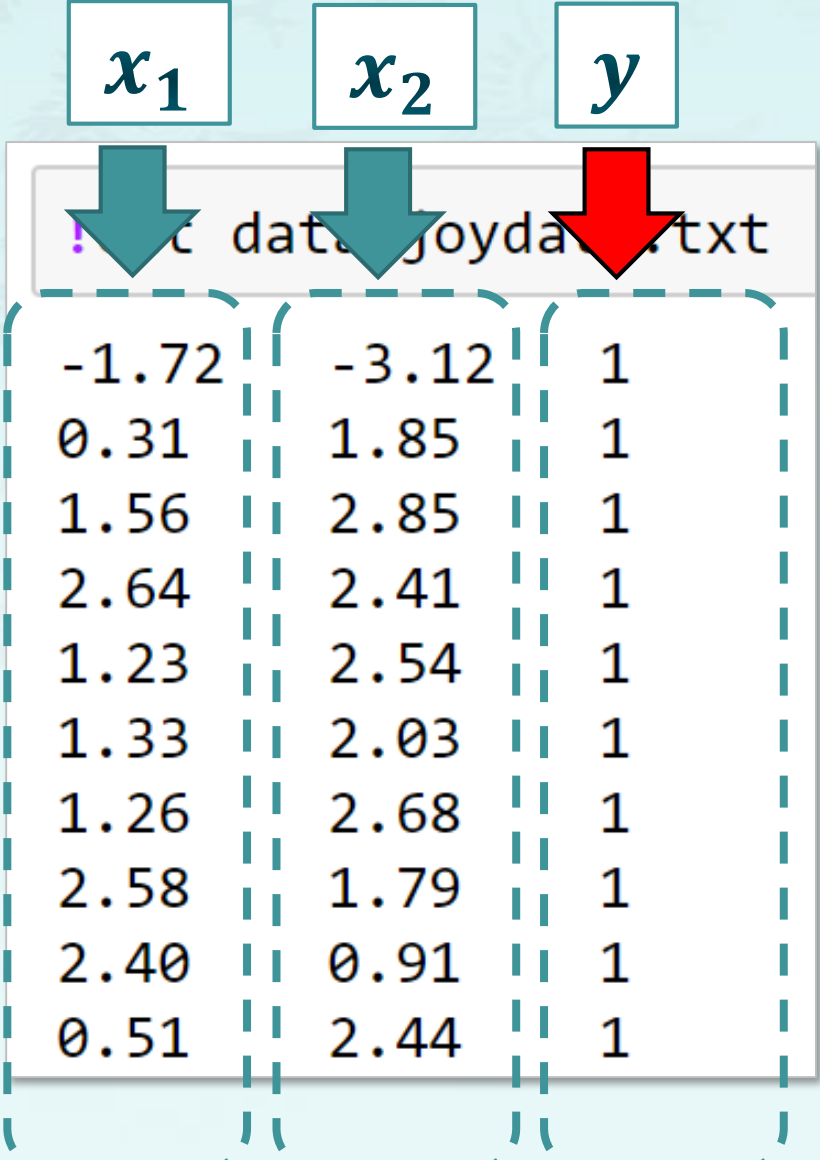
2. 학습 자료의 준비 단계: 표기법

- 학습 자료 준비
 - joydata.txt
- 표기법

특성 feature	클래스 레이블	
↓ ↓ ↓		
: t data joydata.txt		
-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

2. 학습 자료의 준비 단계: 표기법

- 학습 자료 준비
 - joydata.txt
- 표기법



x_1	x_2	y
-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

2. 학습 자료의 준비 단계: 표기법

- 학습 자료 준비
 - joydata.txt
- 표기법

샘플
sample
example

x_1	x_2	y
-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

2. 학습 자료의 준비 단계: 표기법

- 학습 자료 준비
 - joydata.txt
- 표기법

$$x_1^{(1)}, x_2^{(1)}, y^{(1)}$$

$$x_1^{(8)}, x_2^{(8)}, y^{(8)}$$

x_1	x_2	y
-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

2. 학습 자료의 준비 단계: 표기법


- 학습 자료 준비
 - joydata.txt
- 표기법
 - 입력: \mathbf{x} 혹은 \mathbf{X}
 $x^1, x^2, \dots, x^{(i)}, \dots, x^m$
 - 입력의 각 특성: $x_1^i, x_2^i, \dots, x_j^i, \dots, x_n^i$
 - 입력 특성 개수: \mathbf{n}
 - 클래스 레이블: \mathbf{y}
 $y^{(1)}, y^{(2)}, \dots, y^{(i)}, \dots, y^{(m)}$
 - 입력 개수: \mathbf{m}
 - $x_j^{(i)}$: i 번째 샘플의 j 번째 특성 자료
 - 퀴즈: $x_1^{(3)} = ?$

```
!cat data/joydata.txt
```

-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

2. 학습 자료의 준비 단계: 표기법

- 학습 자료 준비
 - joydata.txt
- 표기법
 - 입력: \mathbf{x} 혹은 \mathbf{X}
 $x^1, x^2, \dots, x^{(i)}, \dots, x^m$
 - 입력의 각 특성: $x_1^i, x_2^i, \dots, x_j^i, \dots, x_n^i$
 - 입력 특성 개수: \mathbf{n}
 - 클래스 레이블: \mathbf{y}
 $y^{(1)}, y^{(2)}, \dots, y^{(i)}, \dots, y^{(m)}$
 - 입력 개수: \mathbf{m}
 - $x_j^{(i)}$: i 번째 샘플의 j 번째 특성 자료
 - 퀴즈: $x_1^{(3)} = ?$



```
!cat data/joydata.txt
```

-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

2. 학습 자료의 준비 단계: 표기법

- 학습 자료 준비
 - joydata.txt

- 표기법

- 입력: \mathbf{x} 혹은 \mathbf{X}

➡ $x^1, x^2, \dots, x^{(i)}, \dots, x^m$

- 입력의 각 특성: $x_1^i, x_2^i, \dots, x_j^i, \dots, x_n^i$

- 입력 특성 개수: \mathbf{n}

- 클래스 레이블: \mathbf{y}

➡ $y^{(1)}, y^{(2)}, \dots, y^{(i)}, \dots, y^{(m)}$

- 입력 개수: \mathbf{m}

- $x_j^{(i)}$: i 번째 샘플의 j 번째 특성 자료

- 퀴즈: $x_1^{(3)} = 1.56$

```
!cat data/joydata.txt
```

-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

3. 학습 자료 읽기: 자료구조

- 어떻게 읽을 것인가?

```
!cat data/joydata.txt
```

-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1

3. 학습 자료 읽기: 자료구조

- 어떻게 읽을 것인가?

```
1 data = np.genfromtxt('data/joydata.txt')
2 print(data)
```

```
[[-1.72 -3.12  1.  ]
 [ 0.31  1.85  1.  ]
 [ 1.56  2.85  1.  ]
 [ 2.64  2.41  1.  ]
 [ 1.23  2.54  1.  ]
 [ 1.33  2.03  1.  ]
 [ 1.26  2.68  1.  ]
 [ 2.58  1.79  1.  ]
 [ 2.4   0.91  1.  ]
 [ 0.51  2.44  1.  ]
```

```
!cat data/joydata.txt
```

```
-1.72    -3.12    1
0.31     1.85     1
1.56     2.85     1
2.64     2.41     1
1.23     2.54     1
1.33     2.03     1
1.26     2.68     1
2.58     1.79     1
2.40     0.91     1
0.51     2.44     1
```

3. 학습 자료 읽기: 자료구조

- 기계학습에서 적합한 자료구조

```
1 data = np.genfromtxt('data/joydata.txt')
2 print(data)
```

```
[ [-1.72 -3.12  1.  ]
 [  0.31  1.85  1.  ]
 [  1.56  2.85  1.  ]
 [  2.64  2.41  1.  ]
 [  1.23  2.54  1.  ]
 [  1.33  2.03  1.  ]
 [  1.26  2.68  1.  ]
 [  2.58  1.79  1.  ]
 [  2.4   0.91  1.  ]
 [  0.51  2.44  1.  ]
```

data.shape: (100, 3)

x.shape: (100,2)

y.shape: (100,1)
(100,)

3. 학습 자료 읽기: 자료구조

- 어떻게 **Slicing** 할 것인가?

```
1 data = np.genfromtxt('data/joydata.txt')
2 print(data)
```

```
[ [-1.72 -3.12 1. ]
 [ 0.31  1.85 1. ]
 [ 1.56  2.85 1. ]
 [ 2.64  2.41 1. ]
 [ 1.23  2.54 1. ]
 [ 1.33  2.03 1. ]
 [ 1.26  2.68 1. ]
 [ 2.58  1.79 1. ]
 [ 2.4   0.91 1. ]
 [ 0.51  2.44 1. ]
```

data.shape: (100, 3)

x.shape: (100,2)

y.shape: (100,1)
(100,)

3. 학습 자료 읽기: 자료구조

- Slicing 방법

```
1 data = np.genfromtxt('data/joydata.txt')
2 print(data)
```

```
[ [-1.72 -3.12 1. ]
 [ 0.31  1.85 1. ]
 [ 1.56  2.85 1. ]
 [ 2.64  2.41 1. ]
 [ 1.23  2.54 1. ]
 [ 1.33  2.03 1. ]
 [ 1.26  2.68 1. ]
 [ 2.58  1.79 1. ]
 [ 2.4   0.91 1. ]
 [ 0.51  2.44 1. ]
```

data.shape: (100, 3)

x.shape: (100,2)

y.shape: (100,1)
(100,)

data.shape = (100,3)

x = data[A,B]

y = data[C,D]

3. 학습 자료 읽기: 자료구조

- Slicing 방법

```
1 data = np.genfromtxt('data/joydata.txt')
2 print(data)
```

```
[ [-1.72 -3.12 1. ]
 [ 0.31  1.85 1. ]
 [ 1.56  2.85 1. ]
 [ 2.64  2.41 1. ]
 [ 1.23  2.54 1. ]
 [ 1.33  2.03 1. ]
 [ 1.26  2.68 1. ]
 [ 2.58  1.79 1. ]
 [ 2.4   0.91 1. ]
 [ 0.51  2.44 1. ]
```

data.shape: (100, 3)

x.shape: (100,2)

y.shape: (100,1)
(100,)

data.shape = (100,3)

x = data[:, B]

y = data[:, D]

3. 학습 자료 읽기: 자료구조

- Slicing 방법

```
1 data = np.genfromtxt('data/joydata.txt')
2 print(data)
```

```
[ [-1.72 -3.12 1. ]
 [ 0.31  1.85 1. ]
 [ 1.56  2.85 1. ]
 [ 2.64  2.41 1. ]
 [ 1.23  2.54 1. ]
 [ 1.33  2.03 1. ]
 [ 1.26  2.68 1. ]
 [ 2.58  1.79 1. ]
 [ 2.4   0.91 1. ]
 [ 0.51  2.44 1. ]
```

data.shape: (100, 3)

x.shape: (100,2)

y.shape: (100,1)
(100,)

data.shape = (100,3)

x = data[:, :2]

y = data[:, 2]

3. 학습 자료 읽기: 자료구조

- Slicing 방법

```
1 data = np.genfromtxt('data/joydata.txt')
2 print(data)
```

```
[ [-1.72 -3.12 1. ]
 [ 0.31  1.85 1. ]
 [ 1.56  2.85 1. ]
 [ 2.64  2.41 1. ]
 [ 1.23  2.54 1. ]
 [ 1.33  2.03 1. ]
 [ 1.26  2.68 1. ]
 [ 2.58  1.79 1. ]
 [ 2.4   0.91 1. ]
 [ 0.51  2.44 1. ]
```

data.shape: (100, 3)

x.shape: (100,2)

y.shape: (100,1)
(100,)

data.shape = (100,3)

x = data[:, :2]

y = data[:, 2]

4. 학습 자료 다루기: 자료구조로 읽기 코드

- Slicing 코드

```
1 data = np.genfromtxt('data/joydata.txt')
2 x, y = data[:, :2], data[:, 2]
3 y = y.astype(np.int)
4 print(x[:5])
5 print(y[:5])
```

```
[[ -1.72  -3.12]
 [  0.31   1.85]
 [  1.56   2.85]
 [  2.64   2.41]
 [  1.23   2.54]]
[1 1 1 1 1]
```

4. 학습 자료 다루기: 시각화 코드

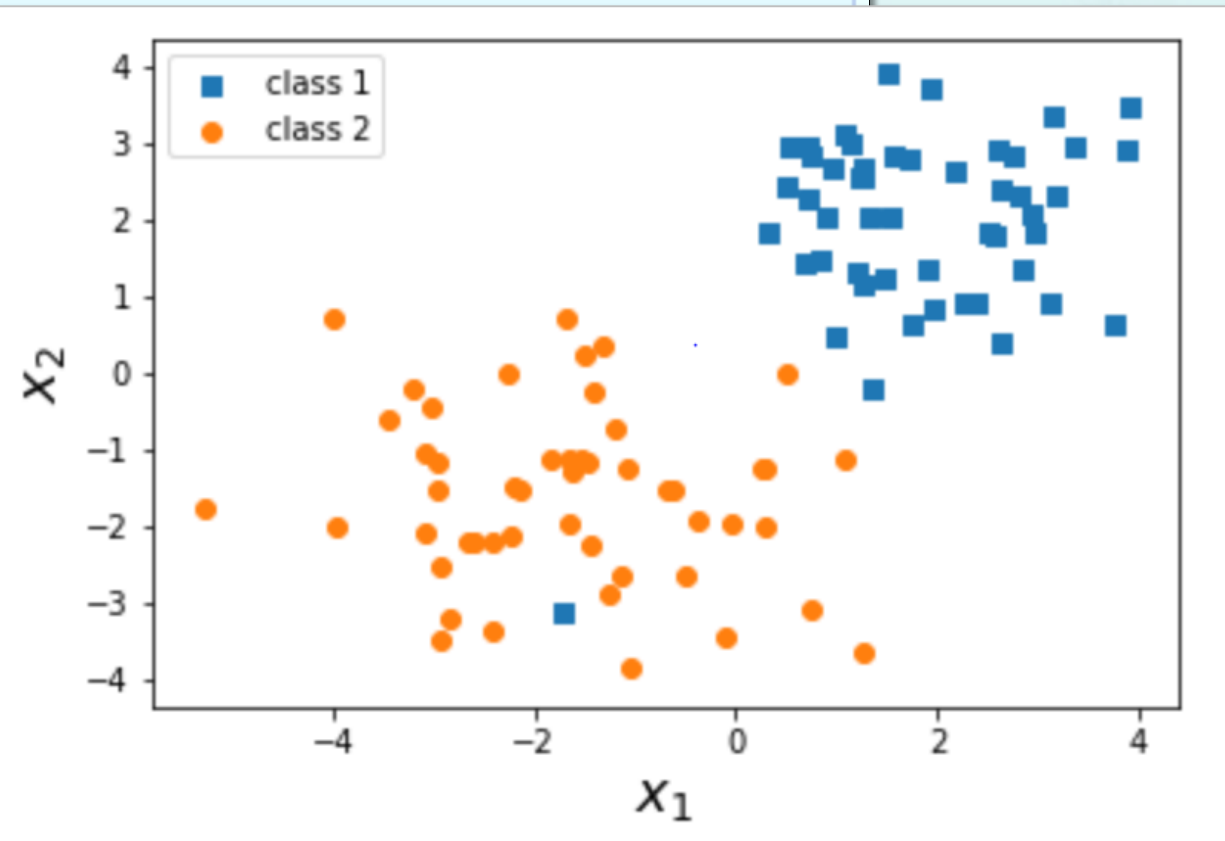
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
data = np.genfromtxt('data/joydata.txt')
x, y = data[:, :2], data[:, 2]
y = y.astype(np.int)

plt.scatter(x[y==1, 0], x[y==1, 1], label='class 1', marker='s')
plt.scatter(x[y==0, 0], x[y==0, 1], label='class 2', marker='o')
plt.xlabel('$x_1$', fontsize=18)
plt.ylabel('$x_2$', fontsize=18)
plt.legend()
plt.show()
```

4. 학습 자료 다루기: 노이즈

```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
data = np.genfromtxt('data/joydata.txt')
x, y = data[:, :2], data[:, 2]
y = y.astype(np.int)

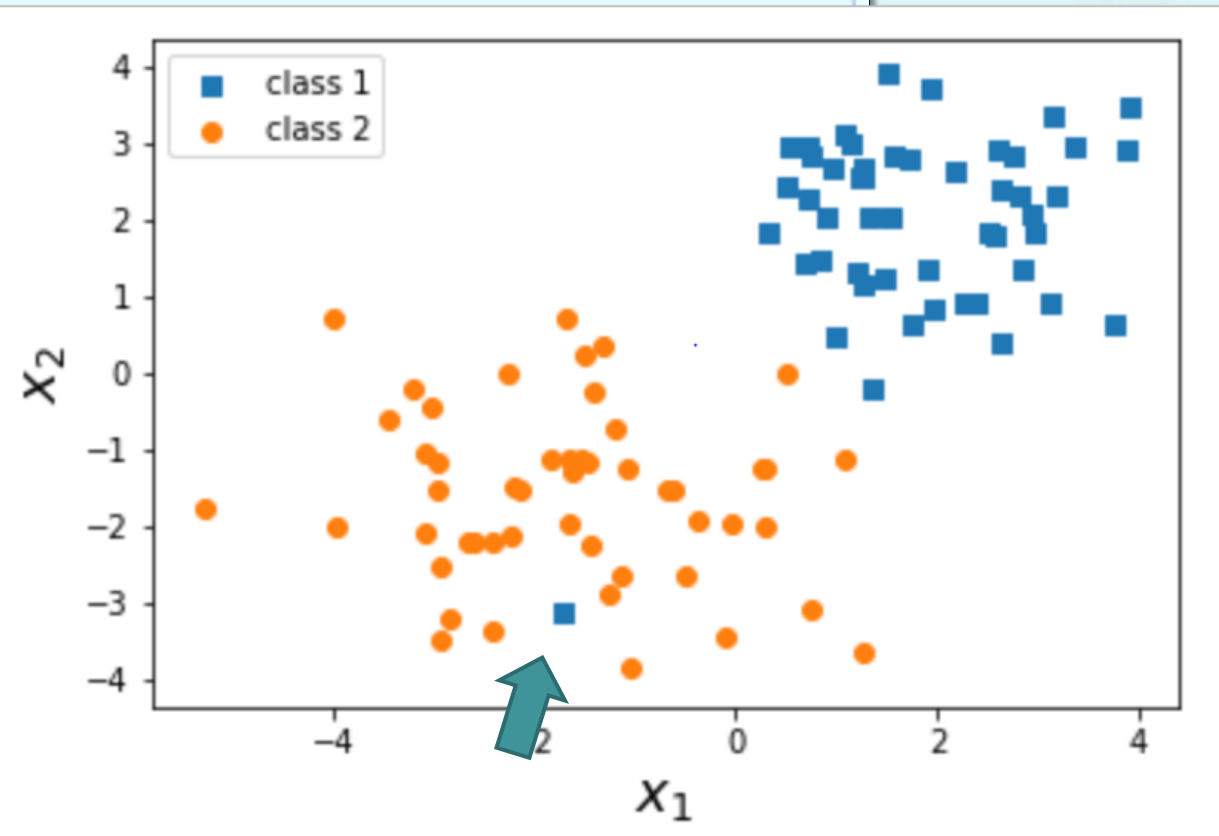
plt.scatter(x[y==1, 0], x[y==1, 1], label='class 1')
plt.scatter(x[y==0, 0], x[y==0, 1], label='class 2')
plt.xlabel('$x_1$', fontsize=18)
plt.ylabel('$x_2$', fontsize=18)
plt.legend()
plt.show()
```



4. 학습 자료 다루기: 노이즈

```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
data = np.genfromtxt('data/joydata.txt')
x, y = data[:, :2], data[:, 2]
y = y.astype(np.int)

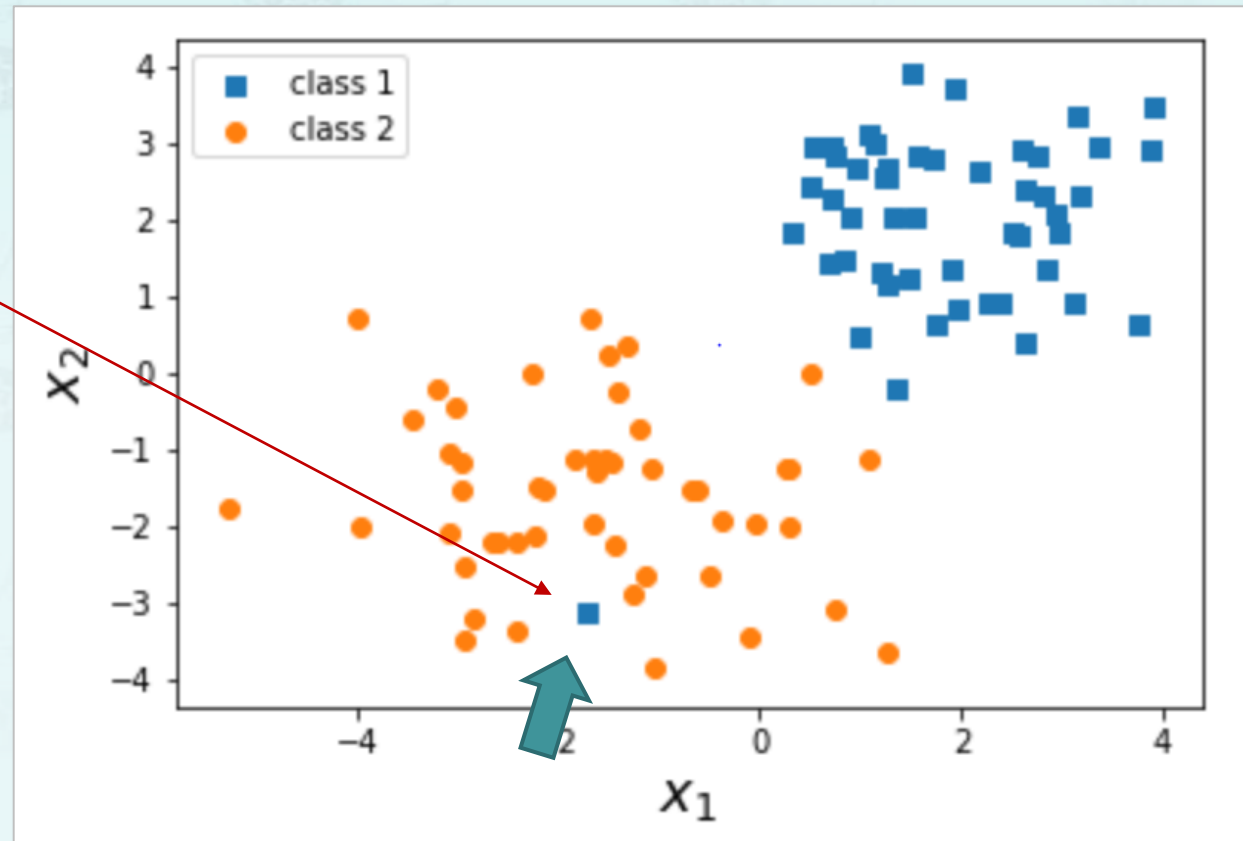
plt.scatter(x[y==1, 0], x[y==1, 1], label='class 1')
plt.scatter(x[y==0, 0], x[y==0, 1], label='class 2')
plt.xlabel('$x_1$', fontsize=18)
plt.ylabel('$x_2$', fontsize=18)
plt.legend()
plt.show()
```



4. 학습 자료 다루기: 노이즈

```
!cat data/joydata.txt
```

-1.72	-3.12	1
0.31	1.85	1
1.56	2.85	1
2.64	2.41	1
1.23	2.54	1
1.33	2.03	1
1.26	2.68	1
2.58	1.79	1
2.40	0.91	1
0.51	2.44	1



기계학습 작업 흐름 1

- 학습 정리
 - 기계학습의 전반적인 작업의 흐름을 이해하기
 - 자료 준비와 전처리 과정 이해하기
- 차시 예고
 - **5-2 기계학습 작업 흐름 2**

5주차(1/3)

기계학습 작업 흐름

파이썬으로 배우는 기계학습

한동대학교
김영섭 교수

여러분 곁에 항상 열려 있는 K-MOOC 강의실에서 만나 뵙기를 바랍니다.