



SVM

- 지도학습 분류모델

HA SEUNG HYUN



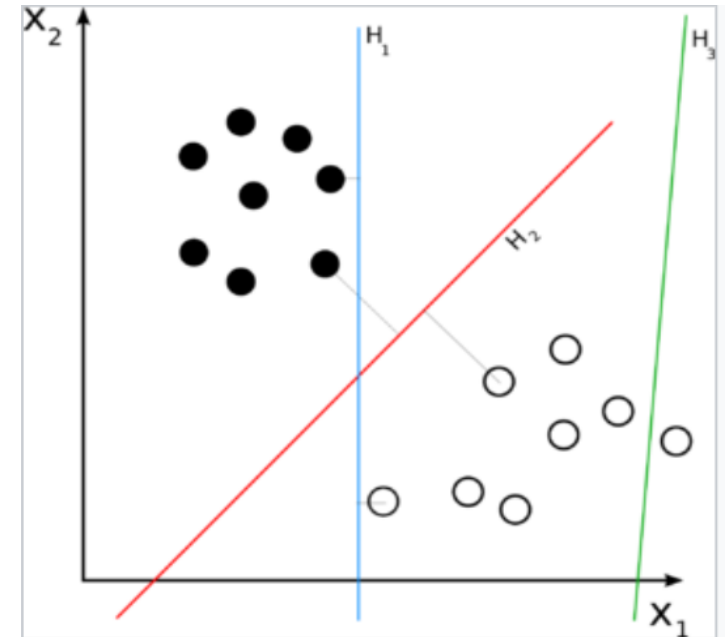
- 1. SVM 개요
- 2. 특징으로 학습하기
- 3. 예측하기
- 4. IRIS 데이터 실습
- 5. Feature와 Label의 연관성



SVM 개요

Classification - SVM

- ▶ SVM (Support Vector Machine)
- ▶ 주로 분류(classification) 및 회귀분석에 쓰이는 지도학습 모델
 - ▶ 비확률적 이진 선형 분류 모델
 - ▶ 두 집단을 분류하는 경계선을 찾는다 생각
- ▶ 데이터를 선형으로 분리하는 최적의 선형 결정 경계를 찾는 알고리즘
- ▶ 마진이란 두 데이터군이 결정경계와 떨어져있는 정도를 말한다.
- ▶ 참고사이트
 - ▶ <https://bskyvision.com/163>
 - ▶ <https://bkshin.tistory.com/entry/머신러닝-2서포트-벡터-머신-SVM>



H3은 두 클래스의 점들을 제대로 분류하고 있지 않다. H1과 H2는 두 클래스의 점들을 분류하는데, H2가 H1보다 더 큰 마진을 갖고 분류하는 것을 확인할 수 있다.

Classification - SVM

▶ Support vector

- ▶ 분류하는 경계선 또는 경계면이 가장 가까운 점과 가장 먼 거리를 가지도록 함
- ▶ **Support vector**란 경계를 결정하는(support) 데이터 점(vector)들을 가리킴

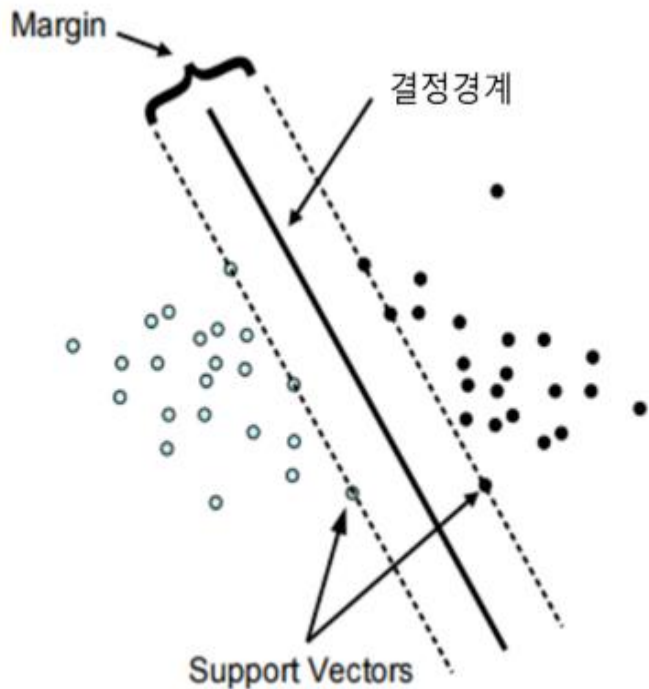


그림4. 마진, 결정 경계, 서포트 벡터.

서포트 벡터들은 두 클래스 사이의 경계에 위치한 데이터 포인트들(그림4에서 점선 위에 있는 데이터들)이다. 많은 데이터가 있지만 그중에 서포트 벡터들이 결정 경계를 만드는데 영향을 준다. 이 데이터들의 위치에 따라 결정 경계의 위치도 달라질 것이다. 즉, 이 데이터들이 결정 경계를 지지(support)하고 있다고 말할 수 있기 때문에, 서포트벡터라고 불리는 것이다.

Classification - SVM

- ▶ SVM의 기본 형태는 두 클래스(집단)를 선형으로 분리시킴
- ▶ 매개변수 cost (비용)
 - ▶ 얼마나 세심하게 경계를 찾을지
 - ▶ cost가 낮으면, outliers를 많이 허용하여 좀 더 일반적인 경계를 찾음
 - ▶ cost가 높으면, 최대한 세심하게 분류하는 경계를 찾음

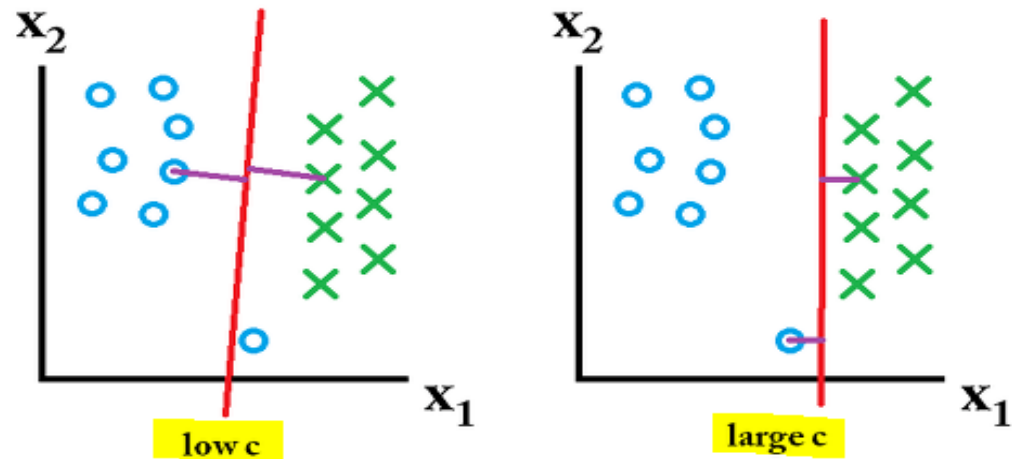


그림6. 매개변수 C의 영향

Classification - SVM

- ▶ SVM의 기본 형태는 두 클래스(집단)를 선형으로 분리시킴
- ▶ But 선형 분리가 안 되는 데이터일 경우..?
 - ▶ 1) 더 높은 차원에서 경계를 찾아 분류함
 - ▶ 2) 이상치(outliers)들을 몇 개 정도 허용함

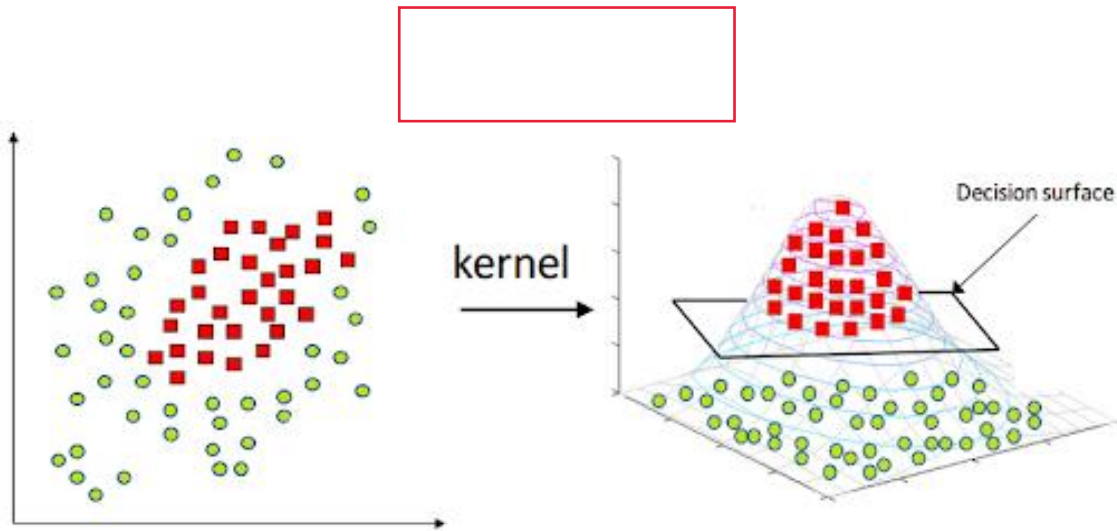


그림5. 이상치(outlier)가 존재하는 경우.

Classification - SVM

서포트 벡터 머신(SVM)의 사용자로서 꼭 알아야할 것들 - 매개변수 C와 gamma

▶ 커널 함수

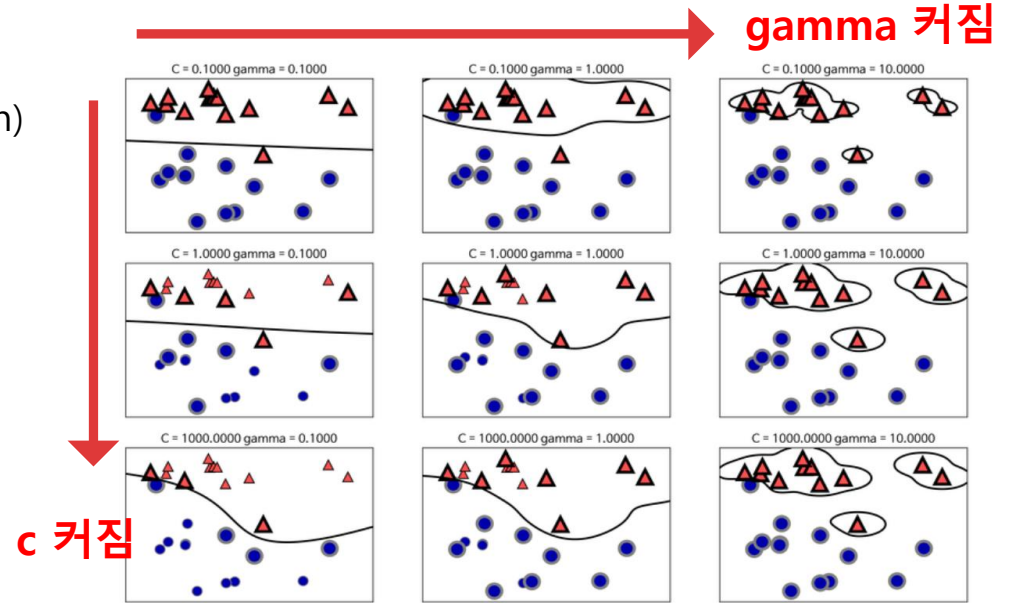
- ▶ 기본적으로는 가우시안 방사 기저 함수(Gaussian Radial Basis Function)
- ▶ 더 높은 차원으로 데이터를 사상시키는(mapping) 함수

▶ 커널 함수의 gamma

- ▶ 각 데이터 점들이 영향을 미치는 거리
- ▶ gamma가 클수록, 거리가 짧아져서 경계가 더 굴곡짐
- ▶ gamma가 작으면, 거리가 길고 더 일반화된 경계를 찾음

▶ 커널 함수의 cost

- ▶ C는 얼마나 많은 데이터 샘플이 다른 클래스에 놓이는 것을 허용하는지를 결정한다



C는 두 데이터를 정확하게 구분하는데 초점을 두고 있고, gamma는 개별 데이터마다 결정선을 지정하는 것에 초점을 둔다. C는 아무리 커져도 결정선이 하나인데 반해서 gamma는 여러 개의 결정선을 만들 수 있다. 두 값 모두 커질수록 알고리즘의 복잡도는 증가한다. 성능을 높이는 것과 overfit을 줄이는 것 사이의 균형을 잘 맞춰야 한다.



인간의 지도 학습 과정- 특징으로 학습하기

지도학습 - 특징들로 학습

Binary Classification



[고양이]

- “야옹”

- 귀가 뽀족하다.

- 입이 짧다.

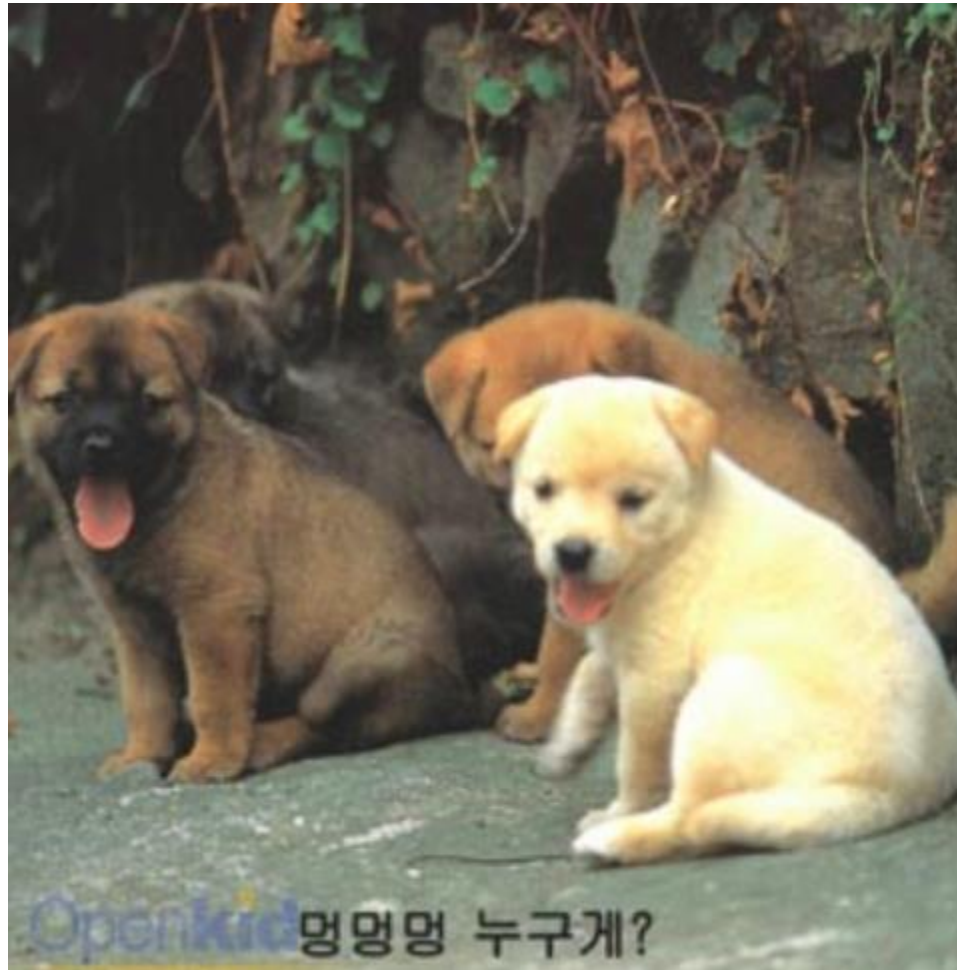
[강아지]

- “멍멍”

- 귀가 크다.

- 입이 길다.

지도학습 - 특징들로 학습



지도학습 - 특징들로 학습

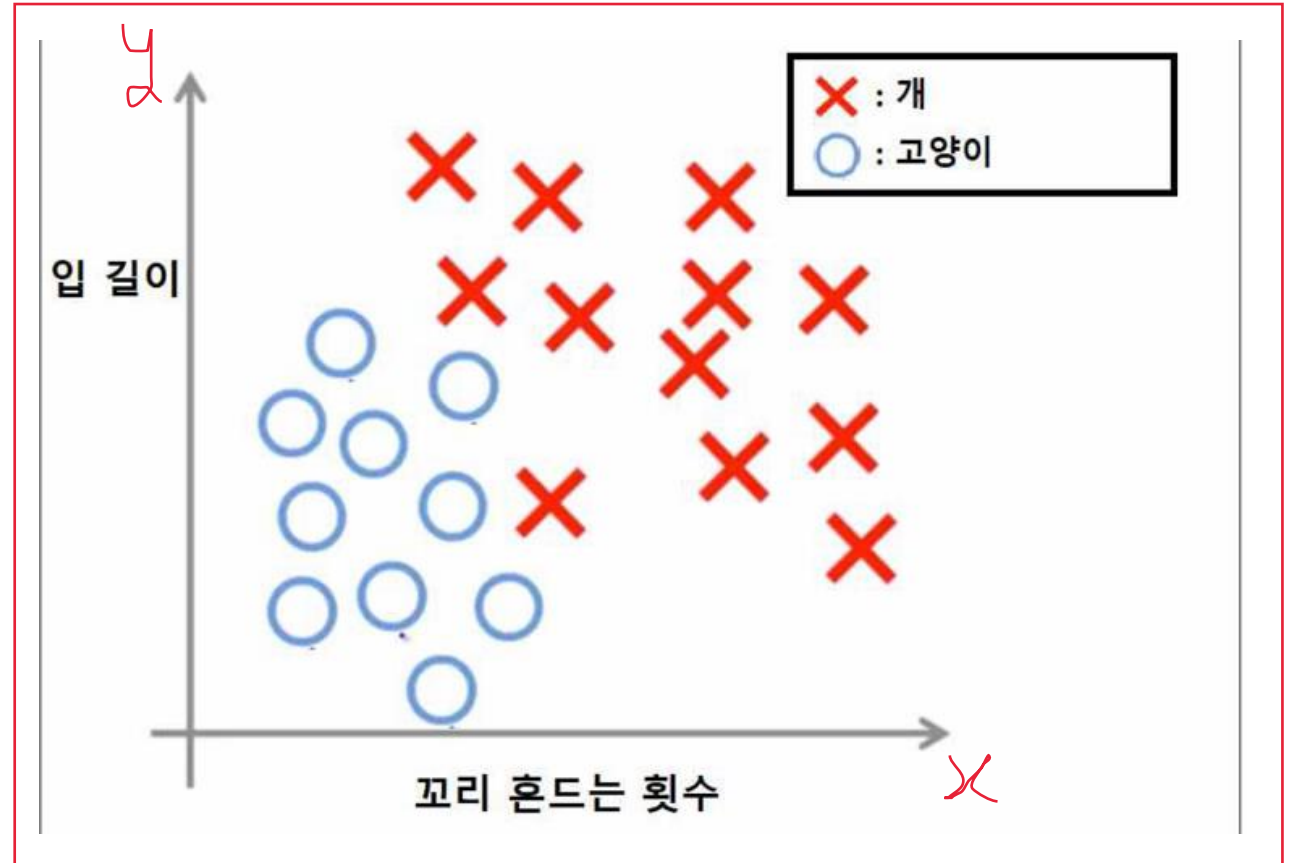
Feature



| 입 길이 1초당 꼬리 흔드는 횟수 | | label 종류 |
|----------------------------|----|-------------|
| 1 | 0 | 고양이 |
| 5 | 11 | 개 |
| 1.1 | 1 | 고양이 |
| 0.9 | 2 | 고양이 |
| 0.8 | 15 | 개 |
| 1.4 | 0 | 고양이 |
| 5.2 | 13 | 개 |
| 1.2 | 1 | 고양이 |

지도학습 - 특징들로 학습

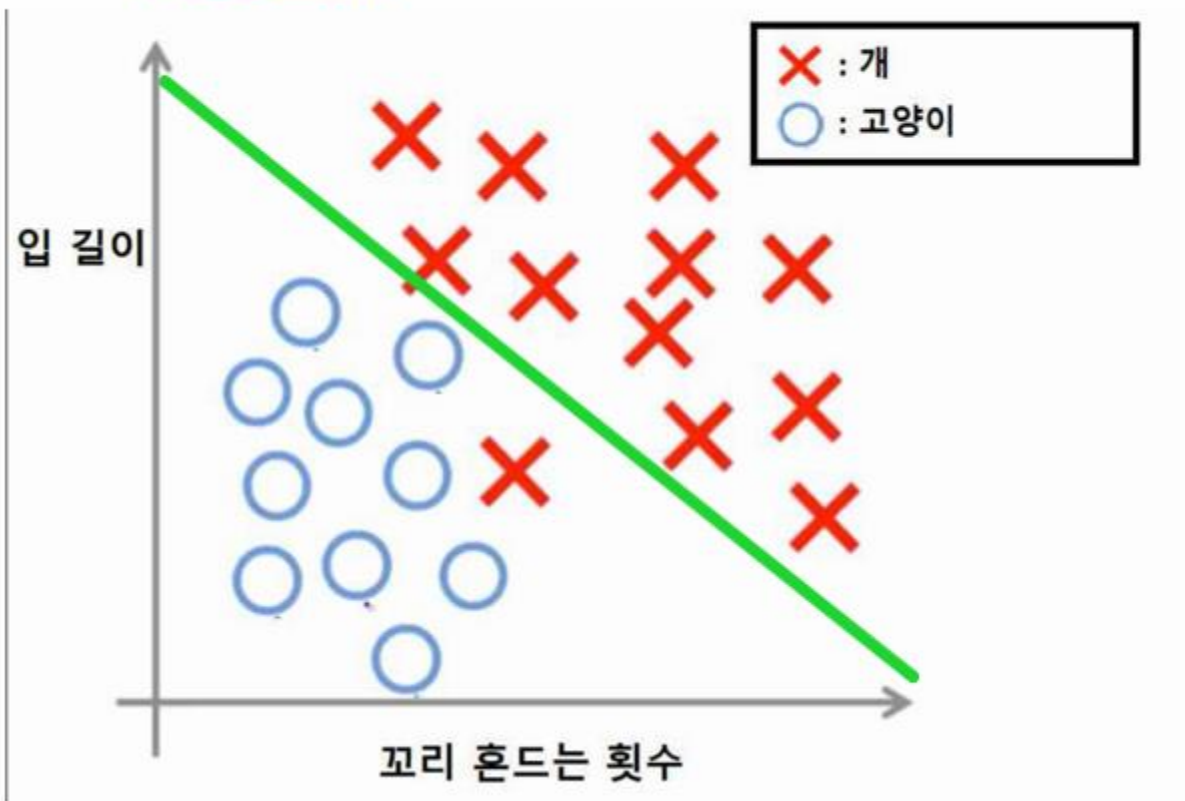
| 입 길이 1초당 꼬리 흔드는 횟수 | | 종류 |
|----------------------------|----|-----|
| 1 | 0 | 고양이 |
| 5 | 11 | 개 |
| 1.1 | 1 | 고양이 |
| 0.9 | 2 | 고양이 |
| 0.8 | 15 | 개 |
| 1.4 | 0 | 고양이 |
| 5.2 | 13 | 개 |
| 1.2 | 1 | 고양이 |



지도학습 - 특징들로 학습

training model

학습완료!!!!

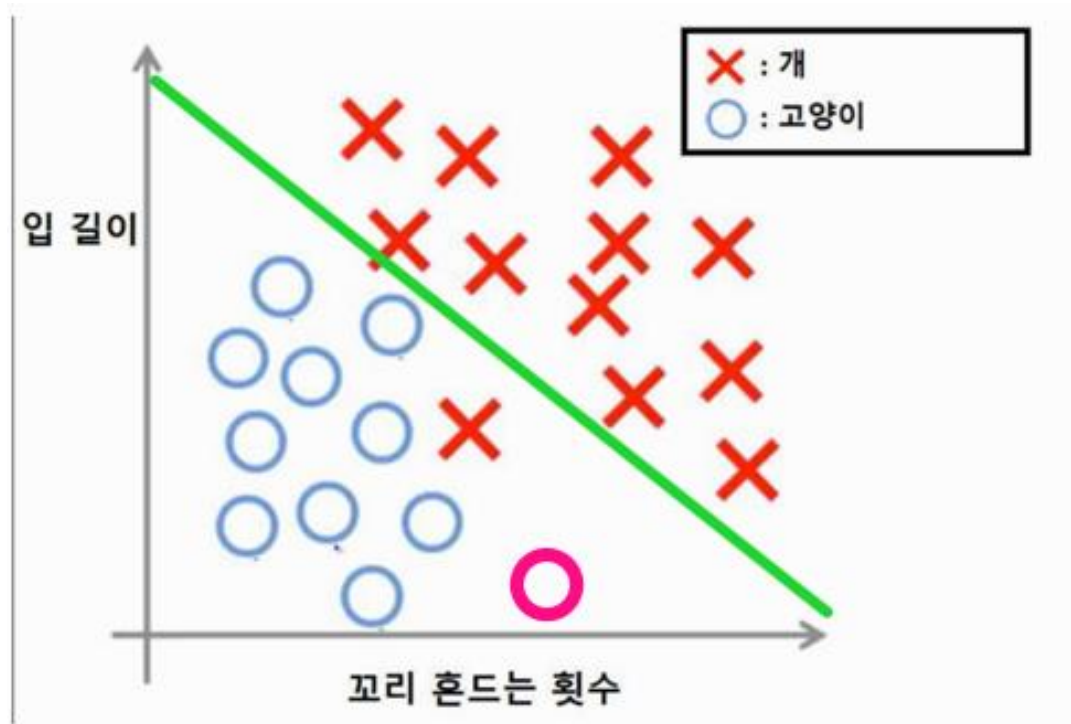


지도학습 - 예측하기

입의 길이가 0.9cm이고,
꼬리 흔들는 횟수가 3회면

강아지?? 고양이???

지도학습 - 예측하기



지도학습 - 실습하기 제공 iris.csv

01_ML_SVM_iris01.ipynb

| | A | B | C | D | E |
|----|--------------|-------------|--------------|-------------|------------|
| 1 | sepal.length | sepal.width | petal.length | petal.width | variety |
| 2 | 5.5 | 3.5 | 1.3 | 0.2 | Setosa |
| 3 | 5.6 | 2.7 | 4.2 | 1.3 | Versicolor |
| 4 | 6.5 | 3.2 | 5.1 | 2 | Virginica |
| 5 | 5.1 | 3.5 | 1.4 | 0.2 | Setosa |
| 6 | 5.1 | 3.5 | 1.4 | 0.3 | Setosa |
| 7 | 6.4 | 2.8 | 5.6 | 2.1 | Virginica |
| 8 | 6.7 | 3 | 5 | 1.7 | Versicolor |
| 9 | 6.1 | 3 | 4.9 | 1.8 | Virginica |
| 10 | 7.9 | 3.8 | 6.4 | 2 | Virginica |

| | | | | | |
|-----|-----|-----|-----|-----|------------|
| 147 | 4.9 | 3.1 | 1.5 | 0.1 | Setosa |
| 148 | 5.5 | 2.6 | 4.4 | 1.2 | Versicolor |
| 149 | 5.6 | 3 | 4.5 | 1.5 | Versicolor |
| 150 | 6.3 | 2.9 | 5.6 | 1.8 | Virginica |
| 151 | 5.1 | 3.8 | 1.9 | 0.4 | Setosa |

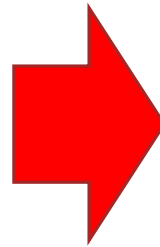
| | A | B | C | D | E |
|-----|--------------|-------------|--------------|-------------|------------|
| 1 | sepal.length | sepal.width | petal.length | petal.width | variety |
| 2 | 5.5 | 3.5 | 1.3 | 0.2 | Setosa |
| 3 | 5.6 | 2.7 | 4.2 | 1.3 | Versicol |
| 4 | 5.5 | 3.2 | 5.1 | 2 | |
| 5 | | 3.5 | 1.4 | | |
| 6 | | 3.5 | 1.4 | | Setosa |
| 7 | 6.4 | | | 2.1 | Virginica |
| 8 | 6.7 | | | 1.7 | Versicolor |
| 9 | 6.1 | | | 1.8 | Virginica |
| 10 | 7.9 | | | 2 | Virginica |
| 147 | | 3.1 | 1.5 | | |
| 148 | 5.5 | 2.6 | 4.4 | 1.2 | |
| 149 | 5.6 | 3 | 4.5 | 1.5 | Versicolor |
| 150 | 6.3 | 2.9 | 5.6 | 1.8 | Virginica |
| 151 | 5.1 | 3.8 | 1.9 | 0.4 | Setosa |

데이터의 일부만
학습용으로 사용

나머지는
검증용으로 사용



지도학습 - 실습하기 제공 iris.csv



| | A | B | C | D | E |
|----|--------------|-------------|--------------|-------------|------------|
| 1 | sepal.length | sepal.width | petal.length | petal.width | variety |
| 2 | 5.5 | 3.5 | 1.3 | 0.2 | Setosa |
| 3 | 5.6 | 2.7 | 4.2 | 1.3 | Versicolor |
| 4 | 6.5 | 3.2 | 5.1 | 2.1 | Virginica |
| 5 | 5.1 | 3.5 | 1.4 | 0.2 | Setosa |
| 6 | 5.1 | 3.5 | 1.4 | 0.3 | Setosa |
| 7 | 6.4 | 2.8 | 5.6 | 2.1 | Virginica |
| 8 | 6.7 | 3 | 5 | 1.7 | Versicolor |
| 9 | 6.1 | 3 | 4.9 | 1.8 | Virginica |
| 10 | 7.9 | 3.8 | 6.4 | 2 | Virginica |

Training Data

train_data | train_label

Validation test

| | | | | | |
|-----|-----|-----|-----|-----|------------|
| 147 | 4.9 | 3.1 | 1.5 | 0.1 | Setosa |
| 148 | 5.5 | 2.6 | 4.4 | 1.3 | Versicolor |
| 149 | 5.6 | 3 | 4.5 | 1.5 | Versicolor |
| 150 | 6.3 | 2.9 | 5.6 | 1.8 | Virginica |
| 151 | 5.1 | 3.8 | 1.9 | 0.4 | Setosa |

Test Data

test_data | test_label

Inference test

Code Processing

1. Module Configuration
2. Data Loader
3. Data분리 - label, data
4. Model Generate
5. Training
6. Predict
7. Accuracy

02_ML_SVM_iris02.ipynb

- Dog 품종 예측하기



지도학습 - Feature와 Label의 연관성

- 데이터 수집하기

| | A | B | C |
|----|----------|------------|--------|
| 1 | 발톱 길이 | 하루에 밥먹는 횟수 | 강아지 품종 |
| 2 | 0.164132 | 5 | 시츄 |
| 3 | 2.910402 | 3 | 치와와 |
| 4 | 3.376386 | 4 | 진돗개 |
| 5 | 3.886729 | 6 | 푸들 |
| 6 | 1.675287 | 1 | 치와와 |
| 7 | 1.437852 | 6 | 허스키 |
| 8 | 0.23381 | 6 | 시츄 |
| 9 | 0.168098 | 2 | 허스키 |
| 10 | 2.508716 | 4 | 치와와 |
| 11 | 1.031392 | 3 | 진돗개 |

컴퓨터 세계
학습 시킬
label

내가 고른
feature

가

지도학습 - Feature와 Label의 연관성

나 ::

그러면,

발톱의 길이가 1.5 cm이고
밥 먹는 횟수가 3회면

무슨 종류의 강아지야???

머신 ::

음...진돗개??

나 ::

공부 안 했구나~~!!

Featurer

.

::
;

Feature Engineering

.

앞에서 살펴보았던

꽃잎의 길이, 꽃잎의 넓이
꽃받침의 길이, 꽃받침의 넓이는

Feature

붓꽃의 품종을 나누는데 있어서
아주 중요한 특징임을 알 수 있다.

Label



타이타닉 생존자 예측하기

- Feature Engineering
- 누락 데이터 처리하기

1. Drop

2.

3. 가 - > binding

1. Data PreProcessing - 문자는 숫자로 매핑

```
sex_mapping = {"male": 0, "female": 1}
```

```
embarked_mapping = {"S": 0, "C": 1, "Q": 2}
```

```
title_mapping = {"Mr": 0, "Miss": 1, "Mrs": 2, "etc": 3}
```

| | Survived | Pclass | Sex | Age | Fare | Embarked | Title | FamilySize |
|---|----------|--------|-----|-----|------|----------|-------|------------|
| 0 | 0 | 3 | 0 | 1.0 | 0.0 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 3.0 | 2.0 | 1 | 2 | 1 |
| 2 | 1 | 3 | 1 | 1.0 | 0.0 | 0 | 1 | 0 |
| 3 | 1 | 1 | 1 | 2.0 | 2.0 | 0 | 2 | 1 |
| 4 | 0 | 3 | 0 | 2.0 | 0.0 | 0 | 0 | 0 |

2. Data PreProcessing - 데이터의 구간화(binning)

```
for dataset in train_test_data:
    dataset.loc[ dataset['Age'] <= 16, 'Age'] = 0
    dataset.loc[(dataset['Age'] > 16) & (dataset['Age'] <= 26), 'Age'] = 1
    dataset.loc[(dataset['Age'] > 26) & (dataset['Age'] <= 36), 'Age'] = 2
    dataset.loc[(dataset['Age'] > 36) & (dataset['Age'] <= 62), 'Age'] = 3
    dataset.loc[ dataset['Age'] > 62, 'Age'] = 4
```

나이를 구간화

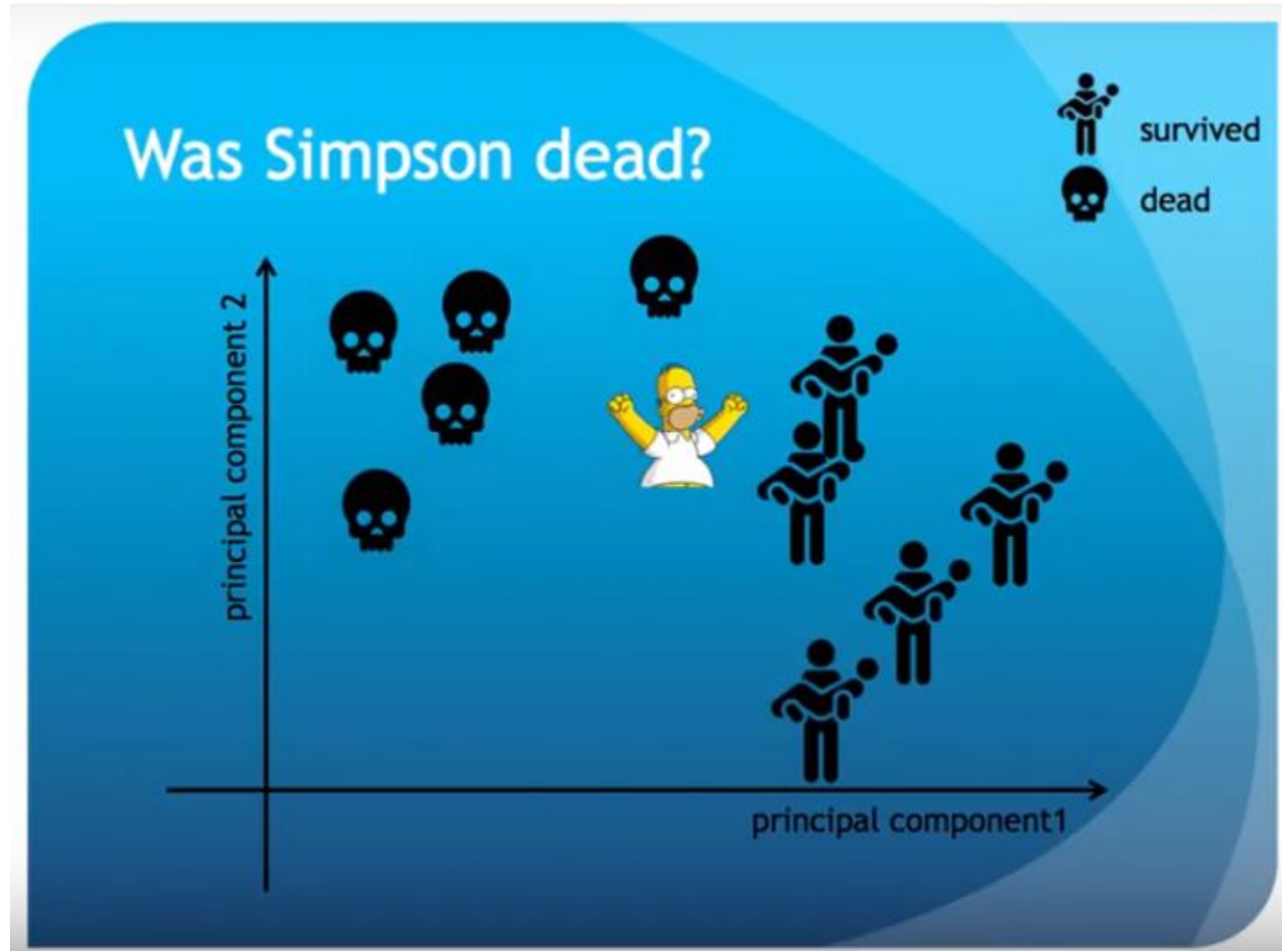
요금을 구간화

```
for dataset in train_test_data:
    dataset.loc[ dataset['Fare'] <= 17, 'Fare'] = 0
    dataset.loc[(dataset['Fare'] > 17) & (dataset['Fare'] <= 30), 'Fare'] = 1
    dataset.loc[(dataset['Fare'] > 30) & (dataset['Fare'] <= 100), 'Fare'] = 2
    dataset.loc[ dataset['Fare'] > 100, 'Fare'] = 3
```

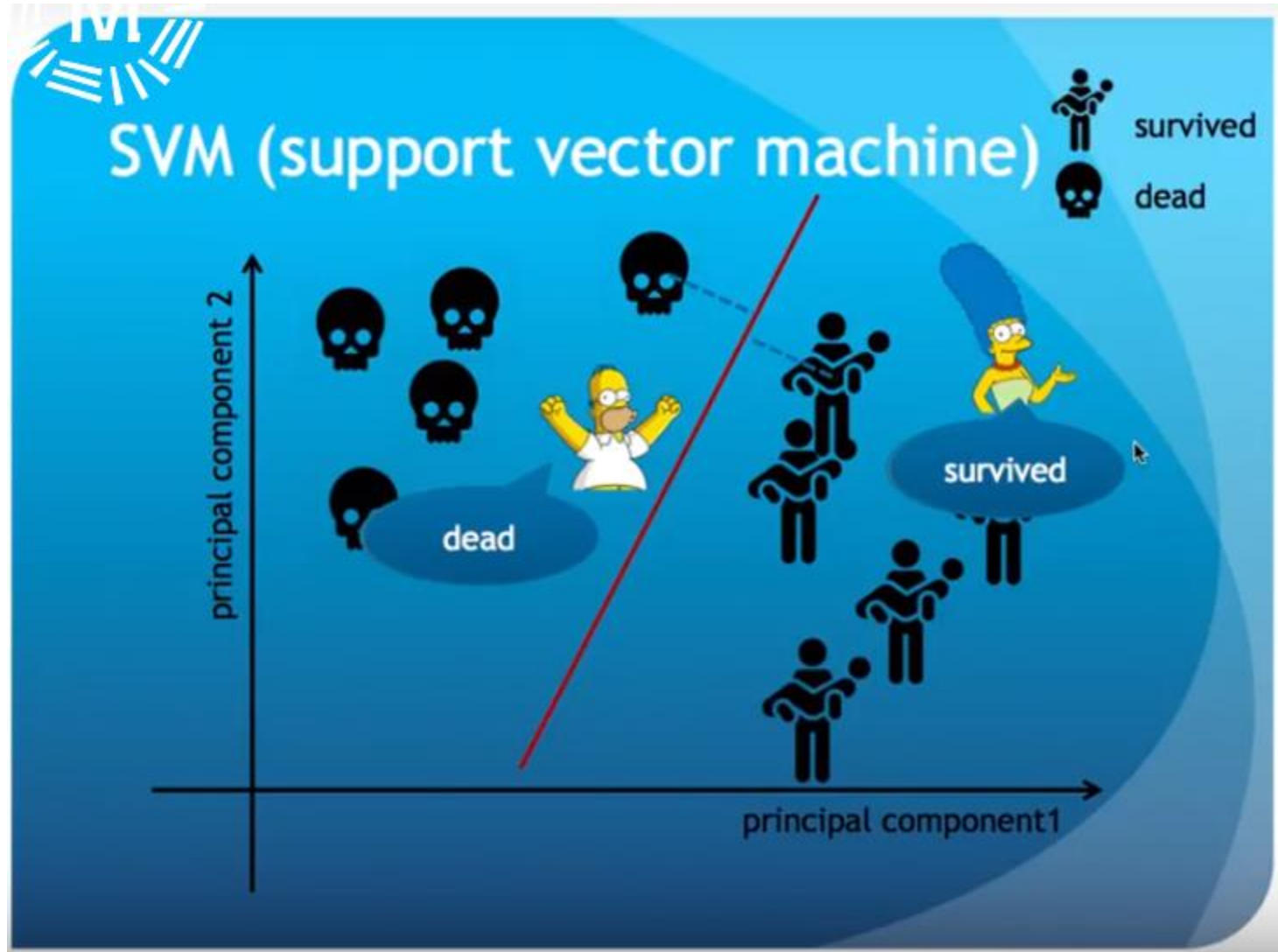
3. Data PreProcessing - 누락데이터 채우기

```
train["Age"].fillna(train.groupby("Title")["Age"].transform("median"), inplace=True)  
test["Age"].fillna(test.groupby("Title")["Age"].transform("median"), inplace=True)
```

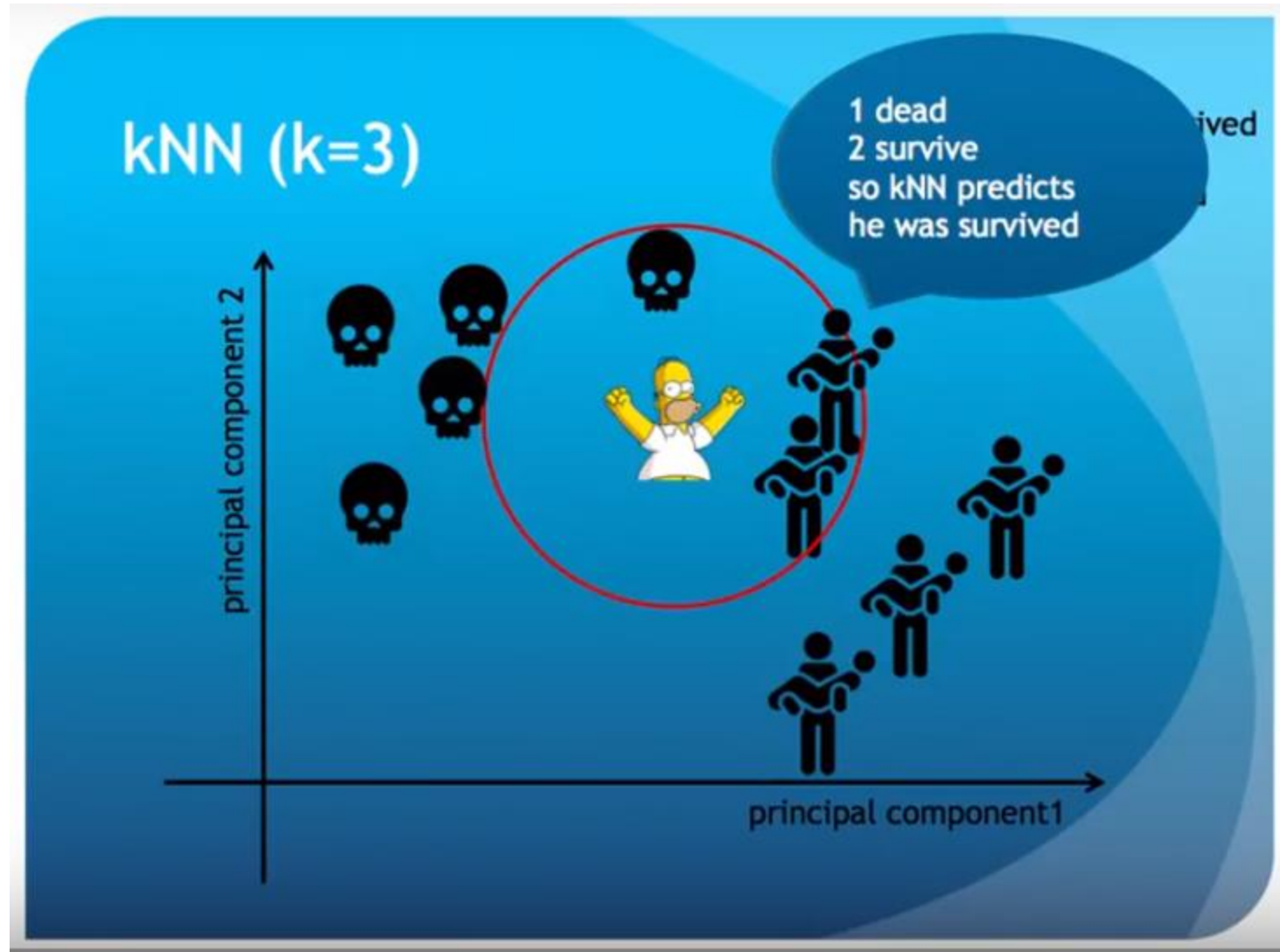
```
train["Fare"].fillna(train.groupby("Pclass")["Fare"].transform("median"), inplace=True)  
test["Fare"].fillna(test.groupby("Pclass")["Fare"].transform("median"), inplace=True)
```



지도학습 - 모델 알고리즘



지도학습 - 모델 알고리즘



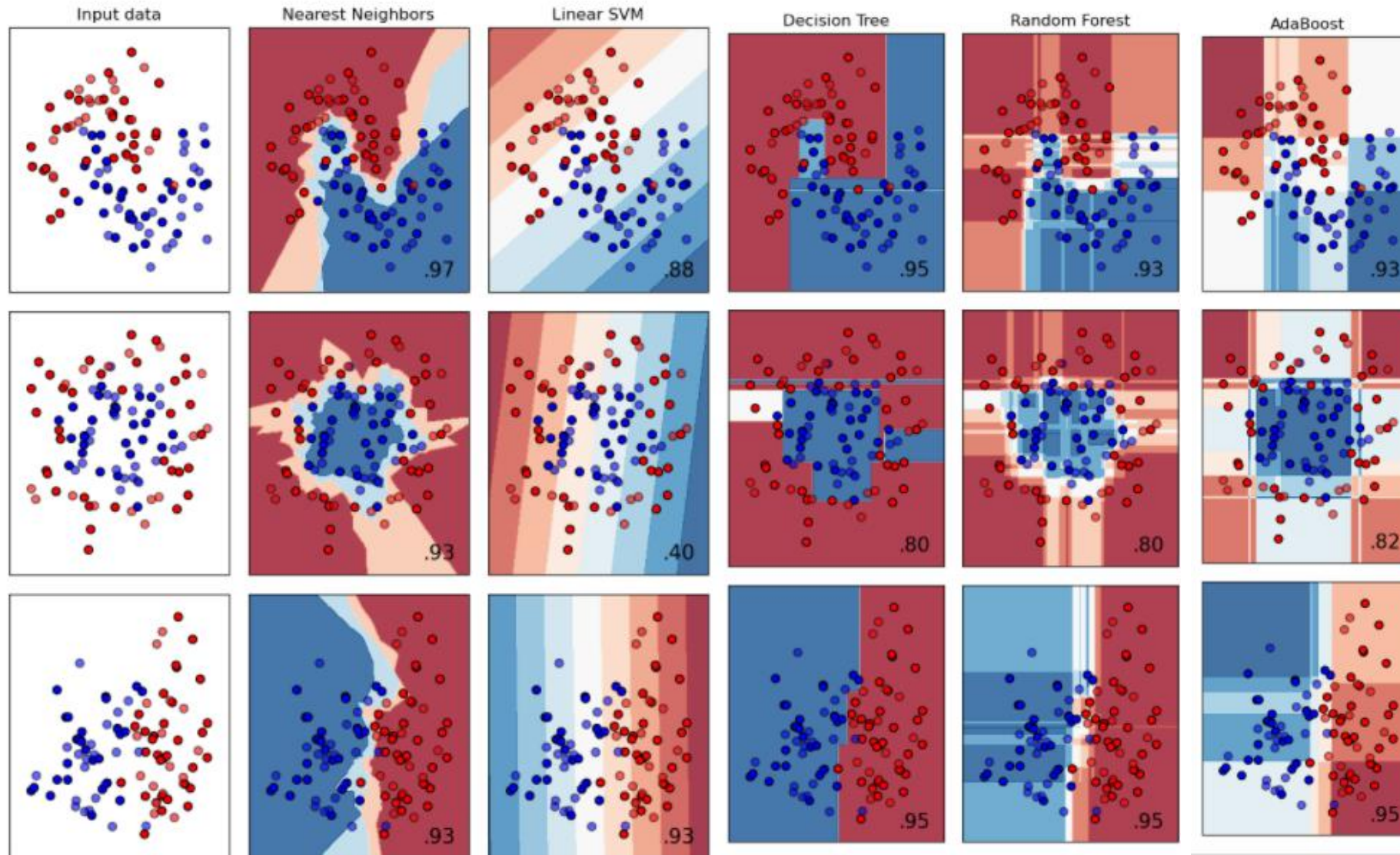
- > overfitting

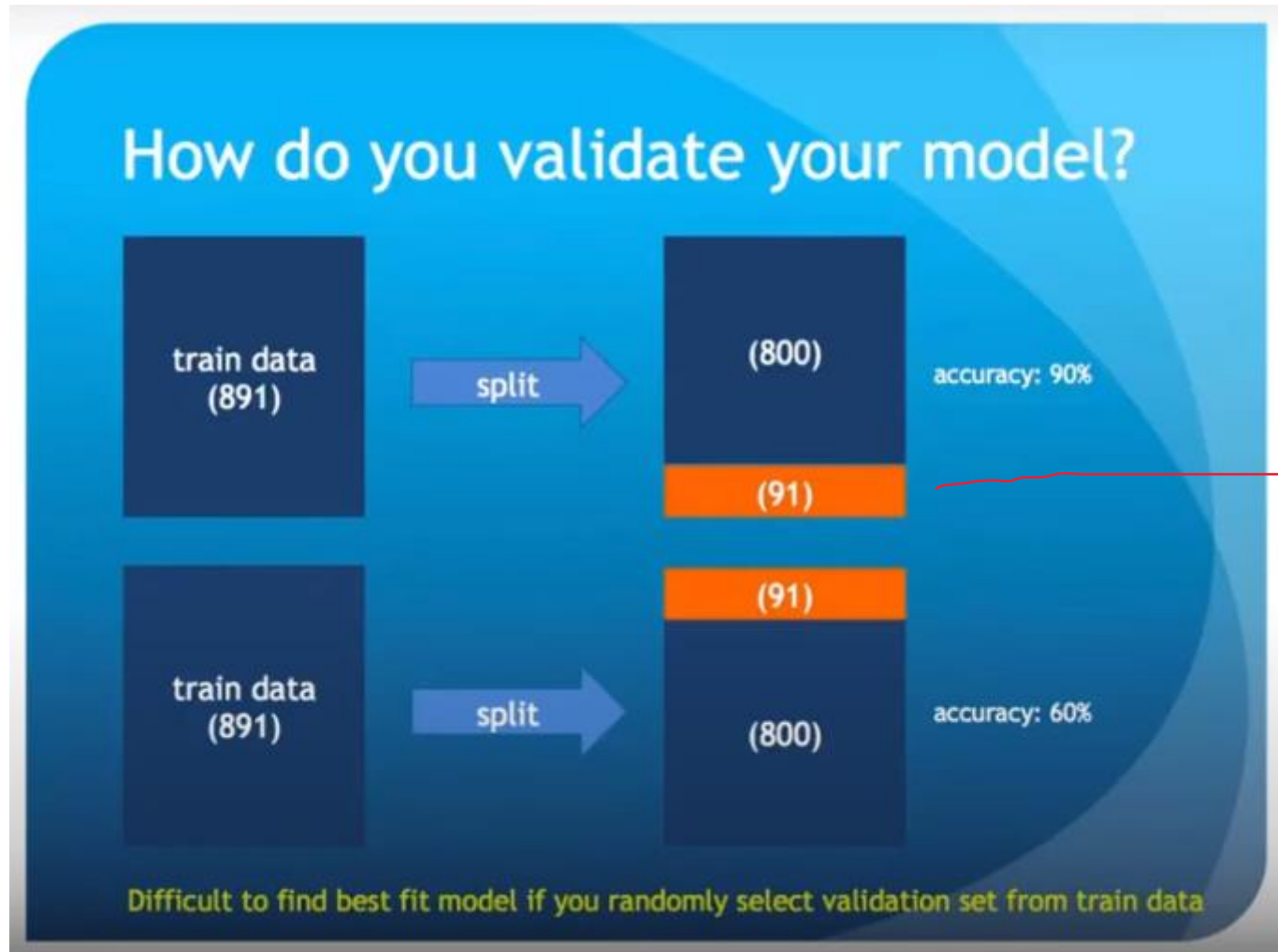


Ensemble - >
overfitting

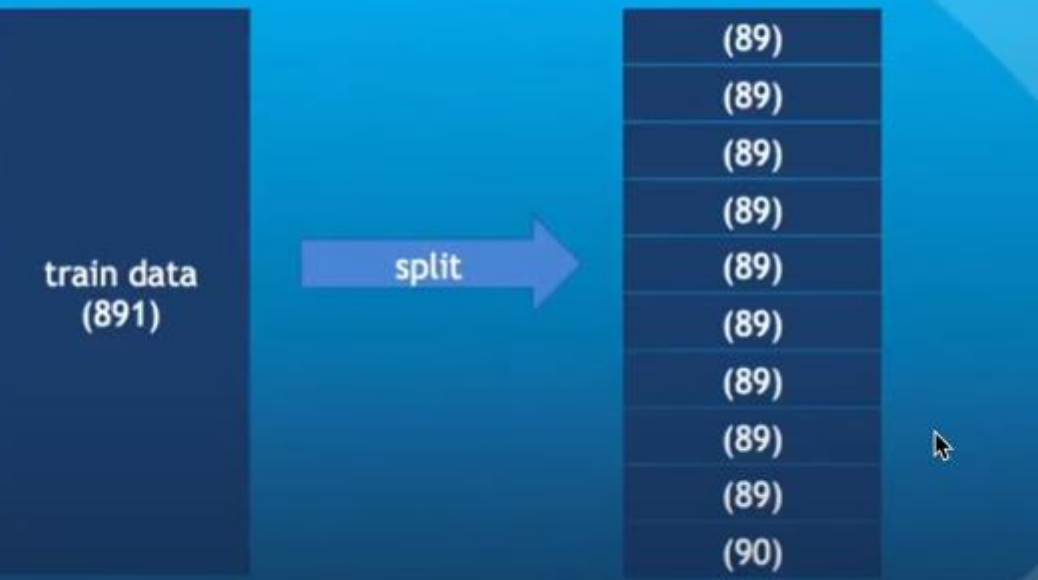


지도 학습 - 모델 알고리즘





k-fold cross validation (1/2)



The diagram illustrates the first step of k-fold cross validation. On the left, a dark blue rectangle represents the initial dataset, labeled "train data (891)". A light blue arrow labeled "split" points from this rectangle to a vertical stack of ten smaller dark blue rectangles on the right. Each of these smaller rectangles represents a fold of the data, with labels (89), (89), (89), (89), (89), (89), (89), (89), (89), and (90) from top to bottom, indicating the size of the training set for each fold.

```
graph LR; A["train data (891)"] -- split --> B["(89)"]; B --> C["(89)"]; C --> D["(89)"]; D --> E["(89)"]; E --> F["(89)"]; F --> G["(89)"]; G --> H["(89)"]; H --> I["(89)"]; I --> J["(90)"];
```

train data (891)

split

(89)
(89)
(89)
(89)
(89)
(89)
(89)
(89)
(89)
(90)

train data
(891)

split

(89)
(89)
(89)
(89)
(89)
(89)
(89)
(89)
(89)
(90)

지도학습 - 학습결과 검증하기

