Chapter. 26

진짜 문제를 해결해보자 (4) Bosch Production Line Performance

# 문제 소개

FAST CAMPUS
ONLINE
데이터 탐색과 전처리 I

강사. 안길승

# I 대회 소개

- 출처: 캐글
  - 문제 제공자: Bosch
  - https://www.kaggle.com/c/bosch-production-line-performance

- 문제 개요: 제조 공정 데이터를 기반으로 불량 여부 예측

- 공정 패턴에 따른 결측이 반드시 존재하는 상황이어서, 특징 추출이 매우 중요한 상황임

# I 사용 데이터

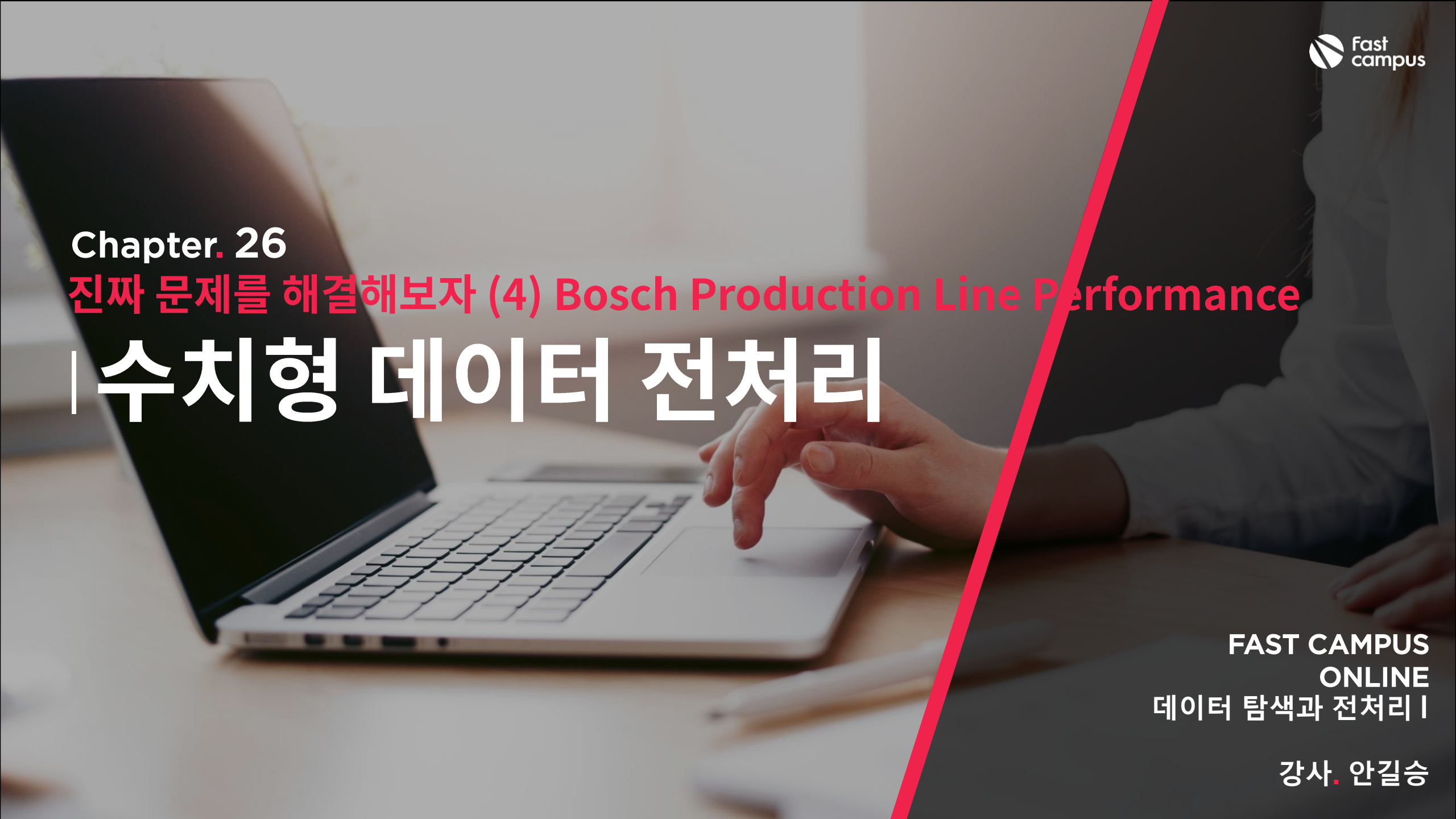
- **train_numeric / test_numeric: 공정 데이터 중 수치형 변수만 포함**
  - L생산라인_S스테이션_F특징번호와 같이 센서 관련 특징이 총 969개가 있음 (예: L3_S36_F3939)
  - 특정 생산라인, 스테이션을 거치지 않은 경우에는 해당하는 특징이 모두 결측
  - Response: 불량 여부 (1: 불량, 0: 양품)

# I 사용 데이터

- **train_categorical / test_categorical: 공정 데이터 중 범주형 변수만 포함**
  - ➢ L생산라인_S스테이션_F특징번호와 같이 범주형 데이터가 총 2140개로 구성 (예: L3_S36_F3939)
  - ➢ 특정 생산라인, 스테이션을 거치지 않은 경우에는 해당하는 특징이 모두 결측

# I 사용 데이터의 특징

- 비식별화된 특징이 매우 많음
- 전체 데이터 가운데, 결측치가 매우 많음
- 심각한 클래스 불균형 문제 존재



Chapter. 26

진짜 문제를 해결해보자 (4) Bosch Production Line Performance

# 수치형 데이터 전처리

FAST CAMPUS
ONLINE
데이터 탐색과 전처리 I

강사. 안길승

# I 데이터 불러오기 및 기본 설정

- sampled_train_numeric.csv를 df로 불러옴
- set_index를 사용하여 제품 식별번호를 나타내는 Id 컬럼을 인덱스로 설정
- 불량 여부를 나타내는 Response를 기준으로 특징 벡터 X와 라벨 Y를 분리함

# I 라인별 스테이션과 특징 확인

- X의 컬럼으로부터 확인한 라인 목록: L0, L1, L2, L3
- X의 컬럼을 _ (under bar)를 기준으로 분할하여, 라인별 스테이션 및 특징 목록을 확인
- 라인은 너무 적고 특징은 너무 많고, 라인별 스테이션이 겹치지 않아, 스테이션을 기준으로 데이터를 정제하기로 결정함

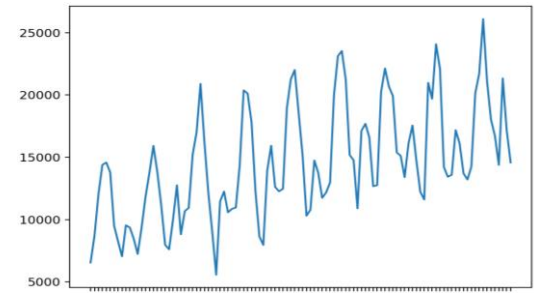
| 라인 | 스테이션 |
|---|---|
| L0 | S0, S1, S2, S3, S4, S5, S6, S7, S8, S9, S10, S11, S12, S13, S14, S15, S16, S17, S18, S19, S20, S21, S22, S23 |
| L1 | S24, S25 |
| L2 | S26, S27, S28 |
| L3 | S29, S30, S31, S32, S33, S34, S35, S36, S37, S38, S39, S40, S41, S43, S44, S45, S47, S48, S49, S50, S51 |

# 제품별 거친 스테이션 목록 확인

- iterrows()를 사용하여 X의 각 행을 row로 순회하면서 결측이 아닌 부분 확인: row.notnull()
- 결측이 아닌 컬럼을 확인: X.columns[row.notnull()]
- isin 함수를 사용하여 제품별 거친 스테이션 목록을 이진 데이터 station_X를 생성함 (1: 거쳐 감, 0: 거쳐가지 않음)

# 측정 값의 통계량 추출

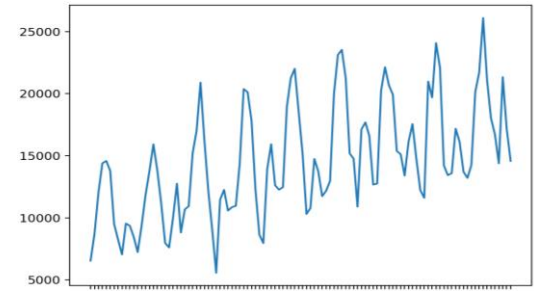
- 시계열에서의 통계량 추출은 **길이가 다른 시계열을 분류**할 때 자주 사용하는 전처리 방법임




$$X^{(1)} = \left(x_1^{(n)}, x_2^{(n)}, \cdots, x_{n_1}^{(n)}\right)$$

통계량 추출

| $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|---|---|---|---|---|
| $S_1^{(1)}$ | $S_2^{(1)}$ | $S_3^{(1)}$ | $S_4^{(1)}$ | $S_5^{(1)}$ |




$$X^{(n)} = \left(x_1^{(n)}, x_2^{(n)}, \cdots, x_{n_t}^{(n)}\right)$$

통계량 추출

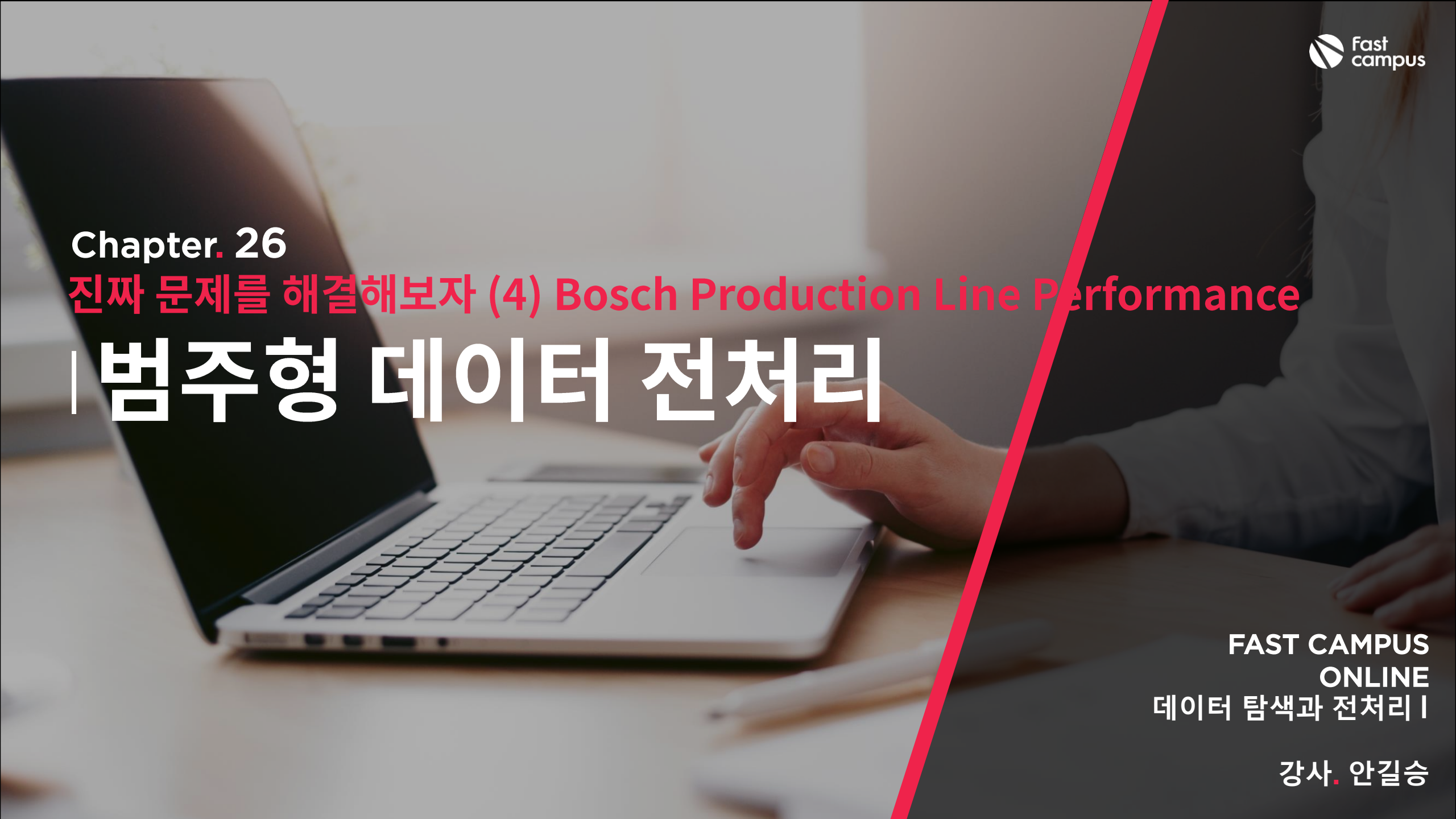
| $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|---|---|---|---|---|
| $S_1^{(n)}$ | $S_2^{(n)}$ | $S_3^{(n)}$ | $S_4^{(n)}$ | $S_5^{(n)}$ |

| $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|---|---|---|---|---|
| $S_1^{(1)}$ | $S_2^{(1)}$ | $S_3^{(1)}$ | $S_4^{(1)}$ | $S_5^{(1)}$ |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| $S_1^{(n)}$ | $S_2^{(n)}$ | $S_3^{(n)}$ | $S_4^{(n)}$ | $S_5^{(n)}$ |

시계열 분류를 위한 데이터

# 측정 값의 통계량 추출 (계속)

- 각 행에 대해 통계량을 추출하기 위한 extract_statistical_features 함수를 정의
- 이 함수는 한 행에 대해 결측 및 이상치를 제거한 뒤, 평균, 분산, 최대, 최소, 첨도, RMS (root mean square)를 구하는 함수임
- extract_statistical_features를 X에 apply한 결과를 stat_feature_X에 저장
- 최종적으로 station_X와 stats_feature_X를 병합하여 numeric_X를 생성



Chapter. 26

진짜 문제를 해결해보자 (4) Bosch Production Line Performance

# 범주형 데이터 전처리

FAST CAMPUS
ONLINE
데이터 탐색과 전처리 I

강사. 안길승

# I 데이터 불러오기 및 기본 설정

- sampled_train_categorical.csv 를 df로 불러옴
- set_index를 사용하여 제품 식별번호를 나타내는 Id 컬럼을 인덱스로 설정

# 등장가능한 모든 값 확인

- df의 모든 행을 순회하면서 NaN이 아닌 모든 값을 codes라는 리스트에 추가함
- 약 서른 개의 값이 발생함을 확인하였으며, T1, T256등과 같이 Txxx꼴의 값임을 확인함
- 데이터 설명이 자세히 되어 있지 않아, 정확하진 않지만, 추측하건대 각 스테이션에서 사용한 툴 코드로 예상됨

# 결측값이 아닌 값이 등장한 스테이션 목록 확인

- df의 모든 행을 돌면서, 결측이 아닌 컬럼에서 스테이션을 추출하여 정리함
- 약 스무 개의 스테이션에서만 결측이 아닌 값이 등장함을 확인

# Code_X 데이터 생성

- 컬럼: 특정 스테이션에서 특정 코드가 발생했는지 여부를 나타냄
  - (예시) S29_T1: 스테이션 S29에 T1이 등장했는지 여부를 나타내는 컬럼

- df의 모든 레코드와 Code_X의 모든 컬럼을 순회하면서, 다음 두 조건을 동시에 만족하는 요소를 확인
  - 특정 컬럼의 값을 가짐
  - 결측이 아님

- Code_X와 X를 병합하는 방식으로 최종 특징 벡터를 생성

Chapter. 26

## 진짜 문제를 해결해보자 (4) Bosch Production Line Performance

# 모델 학습

FAST CAMPUS
ONLINE
데이터 탐색과 전처리 I

강사. 안길승

# 데이터 분할

- 학습 데이터와 평가 데이터를 분할
- 학습 데이터의 크기를 확인

# I 스케일링

- Min – max scaling을 사용하여 학습 데이터와 평가 데이터를 스케일링함

# | 모델 선택

- 특징 대부분이 이진형이며, 연속형이 일부만 섞여 있음
- 타입 차이가 크지 않으므로, 어느 모델을 사용하더라도 무방함
- 샘플과 특징이 모두 많은편이므로, 서포트 벡터 머신을 사용
- 클래스 불균형이 존재하므로, class_weight를 조정

# I 파라미터 튜닝

- SVM 커널을 linear와 rbf를 사용하며, rbf 커널에만 gamma라는 파라미터가 있어서 두 개의 파라미터 그리드를 생성함

| 특징 선택 기준 | 선택하는 특징 개수 | 커널 | 파라미터 |
|---|---|---|---|
| 상호 정보량 | {10, 20, 30, …, 130} | Linear | ✓ C: $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$<br>✓ Class_weight: {{0:1, 1:CI * w}, w = {0.2, 0.4, 0.6, 0.8, 1.0}, 여기서 CI는 클래스 불균형 비율<br>✓ random_state: {10, 20, 30} |
| | | rbf | ✓ C: $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$<br>✓ Class_weight: {{0:1, 1:CI * w}, w = {0.2, 0.4, 0.6, 0.8, 1.0}, 여기서 CI는 클래스 불균형 비율<br>✓ gamma: $\{10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$<br>✓ random_state: {10, 20, 30} |

Chapter. 26

진짜 문제를 해결해보자 (4) Bosch Production Line Performance

# 모델 적용

FAST CAMPUS
ONLINE
데이터 탐색과 전처리 I

강사. 안길승

# 파이프라인 구축

- 새로운 데이터(주의: 이 문제에서는 새로운 데이터가 두 개임)에 대한 예측을 수행하기 위해, 하나의 함수 형태로 파이프라인을 구축함

- 파이프라인을 사용하여 새로운 데이터를 예측함