



센서 로그 데이터 파일을 활용한 Random Forest 기반의 오류 예측 모델

비전공자를 위한 데이터 분석 특강

목차

01

1. 데이터 분석의 목적

02

2. 학습 유형

03

3. 센서 로그 데이터를 통한 지도 학습

04

4. 데이터 준비와 전처리

05

5. 센서 로그 데이터에 적합한 분류 모델 선택

06

6. 모델 평가 및 성능 개선

1. 데이터 분석의 목적

sensor_id	timestamp	sensor_type	value	status	location
S5	2024-01-01	Pressure	26.61953849	OK	Location_A
S1	2024-01-02	Pressure	66.62262344	OK	Location_C
S4	2024-01-03	Humidity	47.24647258	WARN	Location_C
S4	2024-01-04	Pressure	42.53429551	OK	Location_C
S4	2024-01-05	Pressure	62.25103461	OK	Location_C
S2	2024-01-06	Pressure	48.89567541	ERROR	Location_A
S4	2024-01-07	Pressure	37.26510615	WARN	Location_B
S3	2024-01-08	Pressure	50.31786152	OK	Location_A
S5	2024-01-09	Temperature	54.62358334	WARN	Location_C
S1	2024-01-10	Humidity	53.5354835	OK	Location_A
S1	2024-01-11	Pressure	39.38020679	WARN	Location_A

- 데이터 분석의 목적: 데이터를 통해 의미 있는 정보를 찾아내고, 예측 가능성을 높이는 것
- 센서 데이터에서 ERROR 상태를 예측함으로써, 잠재적인 고장이나 이상 상황을 조기에 감지하여 문제를 사전에 예방

2. 학습 유형

- Supervised vs. Unsupervised

Supervised					Unsupervised			
X1	X2	X3	X4	Y	X1	X2	X3	X4

Target

No Target

- 지도 학습 (Supervised) 정답이 있는 데이터를 학습하여 새로운 데이터의 결과를 예측
ex) 센서의 status가 ERROR일 확률 예측
- 비지도 학습 (Unsupervised) 정답 없이 데이터의 패턴을 찾아내는 학습
ex) 여러 센서를 그룹으로 묶어 각 그룹의 특성을 분석
- 강화 학습 (Reinforcement) 주어진 환경에서 최적의 행동을 선택하도록 학습
ex) 로봇이 자율적으로 장애물을 피하는 방법 학습

3. 센서 로그 데이터를 통한 지도 학습

sensor_id	timestamp	sensor_type	value	status	location
S5	2024-01-01	Pressure	26.61953849	OK	Location_A
S1	2024-01-02	Pressure	66.62262344	OK	Location_C
S4	2024-01-03	Humidity	47.24647258	WARN	Location_C
S4	2024-01-04	Pressure	42.53429551	OK	Location_C
S4	2024-01-05	Pressure	62.25103461	OK	Location_C
S2	2024-01-06	Pressure	48.89567541	ERROR	Location_A
S4	2024-01-07	Pressure	37.26510615	WARN	Location_B
S3	2024-01-08	Pressure	50.31786152	OK	Location_A
S5	2024-01-09	Temperature	54.62358334	WARN	Location_C
S1	2024-01-10	Humidity	53.5354835	OK	Location_A
S1	2024-01-11	Pressure	39.38020679	WARN	Location_A

- **sensor_id** : S1, S2, S3
각 센서를 구분하는 고유 ID
- **timestamp** : 2024-01-01 10:30
데이터가 기록된 날짜와 시간
- **sensor_type** : 온도, 습도, 압력
센서의 유형
- **value** : 45.6
센서가 기록한 값
- **status** : OK, WARN, ERROR
센서의 상태
- **location** : Location_A
센서가 위치한 장소

4. 데이터 준비와 전처리

1. sensor_id 컬럼 원핫 인코딩

```
sensor_id_dummies = pd.get_dummies(data['sensor_id'], prefix='sensor_id', dtype=int)
print("\nEncoded sensor_id (One-Hot Encoding):\n", sensor_id_dummies.head())
```

Encoded sensor_id (One-Hot Encoding):

	sensor_id_0	sensor_id_1	sensor_id_2	sensor_id_3	sensor_id_4
0	0	0	0	0	1
1	1	0	0	0	0
2	0	0	0	1	0
3	0	0	0	1	0
4	0	0	0	1	0

2. sensor_type 컬럼 원핫 인코딩

```
sensor_type_dummies = pd.get_dummies(data['sensor_type'], prefix='sensor_type', dtype=int)
print("\nEncoded sensor_type (One-Hot Encoding):\n", sensor_type_dummies.head())
```

Encoded sensor_type (One-Hot Encoding):

	sensor_type_0	sensor_type_1	sensor_type_2
0	0	1	0
1	0	1	0
2	1	0	0
3	0	1	0
4	0	1	0

3. location 컬럼 원핫 인코딩

```
location_dummies = pd.get_dummies(data['location'], prefix='location', dtype=int)
print("\nEncoded location (One-Hot Encoding):\n", location_dummies.head())
```

Encoded location (One-Hot Encoding):

	location_0	location_1	location_2
0	1	0	0
1	0	0	1
2	0	0	1
3	0	0	1
4	0	0	1

01

범주형 데이터 인코딩

텍스트 데이터를 숫자로 변환하는 과정

One-Hot Encoding 방법을 사용하여 sensor_id, sensor_type 등을 처리

02

타겟 변수 변환

status 변수를 이진 분류로 변환

ERROR는 1, 나머지 상태(OK, WARN)는 0으로 변환하여

모델이 이해할 수 있는 형태로 만들기

03

데이터 정규화

데이터의 스케일을 조정하는 과정

Min-Max Scaling으로 0-1 사이 값으로 조정

값의 크기를 일정하게 조정하여 학습 성능을 높임

5. 센서 로그 데이터에 적합한 분류 모델

01

logistic regression

이진 분류 문제에 적합하며, 주로 특정 사건이 발생할 가능성을 예측할 때 사용
ex) 두 가지 결과를 예측할 때 (ERROR 상태인지 아닌지)

02

Decision tree

데이터의 특정 조건을 따라 분기하며 예측을 수행하는 간단한 모델
ex) 여러 질문을 단계적으로 나뉘가며 답을 찾는 과정
질문을 던지며 점점 더 구체적인 답을 찾아가는 20문제 게임과 비슷

03

Random Forest

여러 개의 의사결정나무를 조합하여 예측 성능을 높이는 모델
ex) 다양한 변수(value, sensor_type, location)을 조합해 ERROR 가능성 예측

04

K-Nearest Neighbors

데이터가 위치한 가까운 데이터의 레이블을 보고 예측
ex) value가 비슷한 이웃 데이터가 ERROR면 새로운 데이터도 ERROR로 예측

05

Support Vector Machine

데이터를 경계에 맞춰 분류하는 모델
데이터를 두 그룹으로 나누는 선(경계선)을 찾는 방식
ex) 손글씨 숫자 이미지에서 각 숫자를 인식

5. 센서 로그 데이터에 적합한 분류 모델

01

logistic regression

이진 분류 문제에 적합하며, 주로 특정 사건이 발생할 가능성을 예측할 때 사용
ex) 두 가지 결과를 예측할 때 (ERROR 상태인지 아닌지)

02

Decision tree

데이터의 특정 조건을 따라 분기하며 예측을 수행하는 간단한 모델
ex) 여러 질문을 단계적으로 나뉘가며 답을 찾는 과정
질문을 던지며 점점 더 구체적인 답을 찾아가는 20문제 게임과 비슷

03

Random Forest

여러 개의 의사결정나무를 조합하여 예측 성능을 높이는 모델
ex) 다양한 변수(value, sensor_type, location)을 조합해 ERROR 가능성 예측

04

K-Nearest Neighbors


데이터가 위치한 가까운 데이터의 레이블을 보고 예측
ex) value가 비슷한 이웃 데이터가 ERROR면 새로운 데이터도 ERROR로 예측

05

Support Vector Machine

데이터를 경계에 맞춰 분류하는 모델
데이터를 두 그룹으로 나누는 선(경계선)을 찾는 방식
ex) 손글씨 숫자 이미지에서 각 숫자를 인식

Random Forest 모델을 사용하여 ERROR 상태를 예측
랜덤 포레스트는 센서 데이터처럼
다양한 변수가 있는 데이터에 잘 맞으며,
예측 성능이 높은 편

- 
1. 다양한 변수 처리 능력
 2. 비선형 관계 학습 가능
 3. 데이터의 잡음(Noise)에 강함
 4. 적은 데이터 전처리 요구

6. 모델 평가 및 성능 개선

01

정확도(Accuracy)

전체 예측에서 맞힌 비율을 의미
정확도가 높을수록 잘 예측한 것 같지만, 데이터가 불균형한 경우
(예: ERROR가 적게 발생할 때)에는 이 지표만으로는 부족할 수 있음

02

정밀도(Precision)

ERROR라고 예측한 값 중에서 실제로 ERROR였던 비율
ex) ERROR가 아닌 상황에서 ERROR로 예측하는 것을 줄이고 싶을 때 중요

03

재현율(Recall)

실제 ERROR 상태 중에서 모델이 올바르게 ERROR로 예측한 비율
ERROR 상황을 놓치지 않고 잘 예측해야 할 때 중요한 지표

04

F1-Score

데이터가 위치한 가까운 데이터의 레이블을 보고 예측
ex) value가 비슷한 이웃 데이터가 ERROR면 새로운 데이터도 ERROR로 예측

05

혼동 행렬(Confusion Matrix)

모델이 예측한 결과를 표 형태로 시각화하여, 예측 성공과 실패를 한눈에 파악
True Positive, False Positive, True Negative, False Negative로 나뉘어
각 예측의 정확성과 오류를 확인 할 수 있음

6. 모델 평가 및 성능 개선

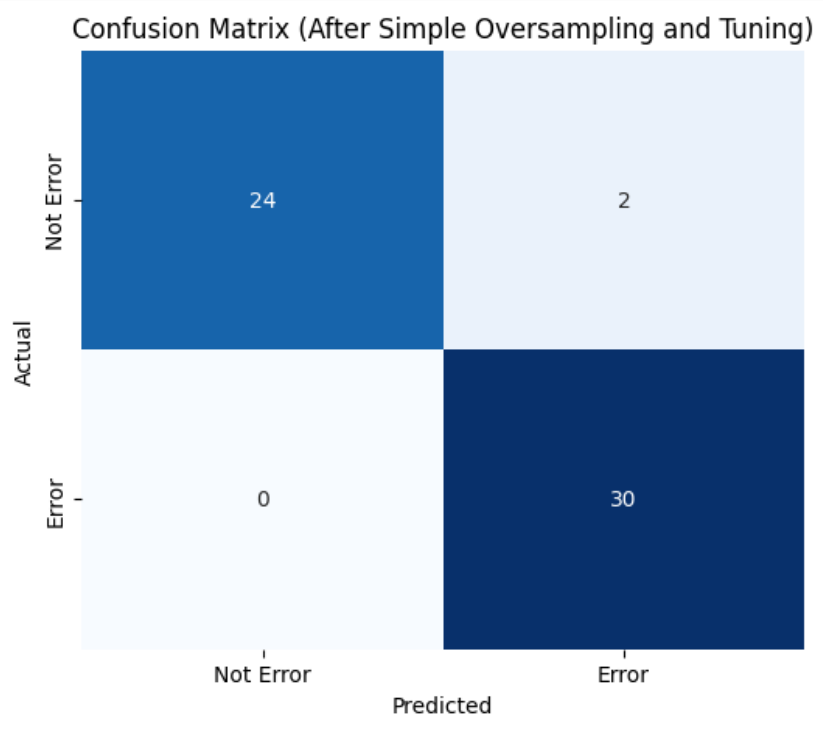


$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1-Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$



6. 모델 평가 및 성능 개선

01

하이퍼파라미터 튜닝

랜덤 포레스트 모델의 `n_estimators`(트리 개수), `max_depth` (트리의 최대 깊이) 등의 하이퍼파라미터를 조정하여 모델 성능을 향상

02

데이터 불균형 처리

센서 데이터에서 ERROR 상태가 적다면 오버샘플링(적은 ERROR 데이터를 늘림) or 언더샘플링(많은 OK 데이터를 줄임)을 통해 데이터 균형을 맞출 수 있음

03

특성 엔지니어링

데이터의 새로운 패턴을 파악할 수 있는 추가 변수를 생성

04

다른 모델 시도

XGBoost, LightGBM과 같은 더 복잡한 앙상블 모델을 시도하여 랜덤 포레스트보다 더 높은 성능을 얻을 수 있음

Task

AI를 활용하여 센서 로그 데이터 파일인 `sensor_log.csv`를 읽고,
Random Forest 기반의 오류 예측 모델 만들기

(필수) 예측 결과 저장

예측 결과와 실제 값이 함께 포함된 DataFrame을 Excel 파일로 저장하여 제출
