



9월 1주차(2024.09.02~2024.09.08) - 모델

Random Forest + XGBoost

데이터셋 형태 변환

1. 기존 방법
 2. 새로운 방법
- 위의 두 가지 형태에 대한 차이

Random Forest 학습

모델 테스트

Random Forest + XGBoost

- 각 구역의 데이터 개수: 626개
- 테스트 데이터 개수

구역	개수	구역	개수	구역	개수	구역	개수
A_1	76	B_1	121	C_1	21	D_1	129
A_2	76	B_2	128	C_2	19	D_2	130
A_3	75			C_3	20		

- 학습 및 검증 결과

랜덤 포레스트 최적 하이퍼파라미터: {'bootstrap': False, 'max_depth': 16, 'max_features': 'log2', 'm
XGBoost 최적 하이퍼파라미터: {'colsample_bytree': 0.6, 'gamma': 0, 'learning_rate': 0.1, 'max_de
양상블 모델 학습 완료
검증 정확도: 0.8447
모델과 인코더 저장 완료

- 테스트 결과

파일 'A_1_testdata'에 대한 예측 요약:

구역 'A_1': 71번 예측됨

구역 'A_2': 4번 예측됨

구역 'B_1': 1번 예측됨

파일 'A_1_testdata'의 최종 예측 구역: A_1

모델의 정확도: 0.9342

파일 'A_2_testdata'에 대한 예측 요약:

구역 'A_2': 74번 예측됨

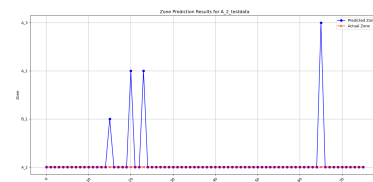
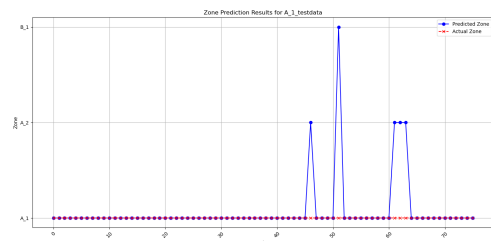
구역 'B_1': 1번 예측됨

구역 'A_1': 1번 예측됨

파일 'A_2_testdata'의 최종 예측 구역: A_2

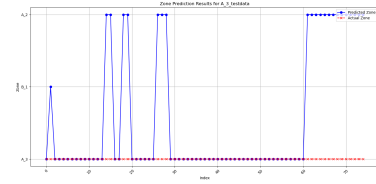
모델의 정확도: 0.9737

파일 'A_3_testdata'에 대한 예측 요약:



구역 'A_3': 56번 예측됨
 구역 'A_2': 18번 예측됨
 구역 'B_1': 1번 예측됨
 파일 'A_3_testdata'의 최종 예측 구역: A_3

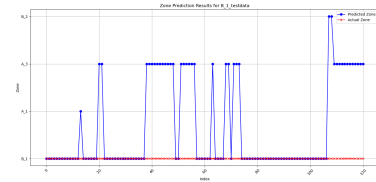
모델의 정확도: 0.7467



파일 'B_1_testdata'에 대한 예측 요약:

구역 'B_1': 76번 예측됨
 구역 'A_3': 37번 예측됨
 구역 'B_2': 6번 예측됨
 구역 'A_1': 1번 예측됨
 구역 'A_2': 1번 예측됨
 파일 'B_1_testdata'의 최종 예측 구역: B_1

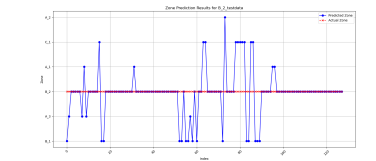
모델의 정확도: 0.6281



파일 'B_2_testdata'에 대한 예측 요약:

구역 'B_2': 89번 예측됨
 구역 'B_1': 18번 예측됨
 구역 'C_1': 10번 예측됨
 구역 'A_3': 6번 예측됨
 구역 'A_1': 4번 예측됨
 구역 'A_2': 1번 예측됨
 파일 'B_2_testdata'의 최종 예측 구역: B_2

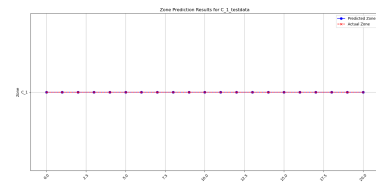
모델의 정확도: 0.6953



파일 'C_1_testdata'에 대한 예측 요약:

구역 'C_1': 21번 예측됨
 파일 'C_1_testdata'의 최종 예측 구역: C_1

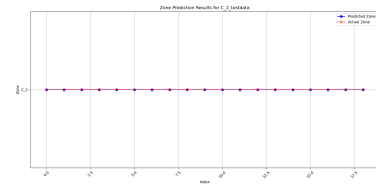
모델의 정확도: 1.0000



파일 'C_2_testdata'에 대한 예측 요약:

구역 'C_2': 19번 예측됨
 파일 'C_2_testdata'의 최종 예측 구역: C_2

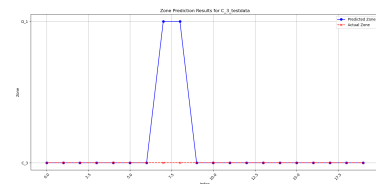
모델의 정확도: 1.0000



파일 'C_3_testdata'에 대한 예측 요약:

구역 'C_3': 18번 예측됨
 구역 'D_1': 2번 예측됨
 파일 'C_3_testdata'의 최종 예측 구역: C_3

모델의 정확도: 0.9000



파일 'D_1_testdata'에 대한 예측 요약:

구역 'D_2': 63번 예측됨

구역 'D_1': 50번 예측됨

구역 'E_3': 16번 예측됨

파일 'D_1_testdata'의 최종 예측 구역: D_2

모델의 정확도: 0.3876

파일 'D_2_testdata'에 대한 예측 요약:

구역 'D_1': 64번 예측됨

구역 'D_2': 49번 예측됨

구역 'C_3': 16번 예측됨

구역 'E_3': 1번 예측됨

파일 'D_2_testdata'의 최종 예측 구역: D_1

모델의 정확도: 0.3769

파일 'E_1_testdata'에 대한 예측 요약:

구역 'E_1': 51번 예측됨

구역 'E_2': 9번 예측됨

파일 'E_1_testdata'의 최종 예측 구역: E_1

모델의 정확도: 0.8500

파일 'E_2_testdata'에 대한 예측 요약:

구역 'E_2': 37번 예측됨

구역 'E_1': 20번 예측됨

구역 'E_3': 3번 예측됨

파일 'E_2_testdata'의 최종 예측 구역: E_2

모델의 정확도: 0.6167

파일 'E_3_testdata'에 대한 예측 요약:

구역 'E_3': 34번 예측됨

구역 'D_1': 14번 예측됨

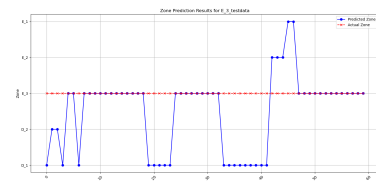
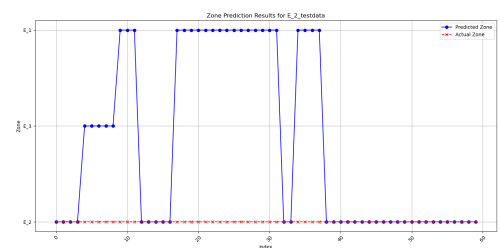
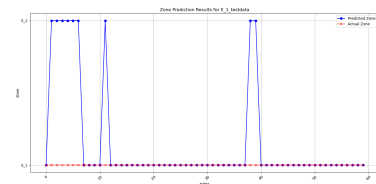
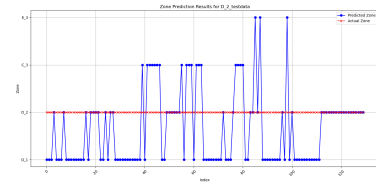
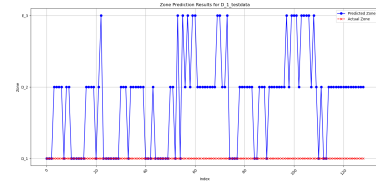
구역 'D_2': 6번 예측됨

구역 'E_2': 3번 예측됨

구역 'E_1': 3번 예측됨

파일 'E_3_testdata'의 최종 예측 구역: E_3

모델의 정확도: 0.5667



데이터셋 형태 변환

1. 기존 방법

- 학습 시에 MAC Address를 입력으로 사용하였다.
- 그러나, MAC Address를 입력으로 사용하지 않는 것이 일반적이다.

```
No., TimeStamp, MAC Address, RSSI, Zone
1, 01:36:43, 60:98:66:32:8E:28, -76, A_1
2, 01:36:45, 60:98:66:32:8E:28, -77, A_1
3, 01:36:46, 60:98:66:32:8E:28, -74, A_1
4, 01:36:50, 60:98:66:33:42:D4, -58, A_1
5, 01:36:52, 60:98:66:33:42:D4, -67, A_1
6, 01:36:53, 60:98:66:33:42:D4, -64, A_1
7, 01:36:54, 60:98:66:33:42:D4, -79, A_1
8, 01:36:58, 60:98:66:32:BC:AC, -82, A_1
9, 01:36:59, A0:6C:65:99:DB:7C, -96, A_1
10, 01:36:59, 60:98:66:32:BC:AC, -86, A_1
```

2. 새로운 방법

- 그렇기에 MAC Address는 모두 column으로 변환하고, 입력은 RSSI 값만 되도록 한다.
 - 동일한 시간에 여러 개의 비콘 신호를 하나의 row에 저장한다.

```
TimeStamp, B1, B2, B3, B4, B5, B6, B7, B8, B9, B10, B11, B12, B13, B14, B15, B16, B17, B18, Zone
00:23:58, 0, 0, -81, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
00:24:00, 0, 0, -80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
00:24:01, -64, 0, -80, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
00:24:02, -73, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
00:24:03, -81, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
00:24:04, -65, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
00:24:06, 0, -73, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
00:24:08, -59, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
00:24:09, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, -88, 0, 0, 0, A_1
00:24:10, 0, -75, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, A_1
```

위의 두 가지 형태에 대한 차이

1. 기존 방법: 비콘별 개별 기록

```
No., TimeStamp, MAC Address, RSSI, Zone
1, 01:36:43, 60:98:66:32:8E:28, -76, A_1
...
```

- 특징
 - 각 행은 하나의 비콘(MAC Address)의 RSSI 값을 기록
 - 하나의 타임스탬프에서 하나의 비콘 신호만 기록
- 장점
 - 개별 비콘의 신호를 기록하기에 비콘 간의 독립적인 변화를 관찰할 때 유용
- 적합한 경우
 - 비콘 신호 자체의 변화를 더 세밀하게 학습하거나, 개별 비콘의 신호를 분석할 때 유리

- 각각의 비콘 신호를 개별적으로 분석하고, 그 신호를 바탕으로 예측하는 경우

2. 새로운 방법: 정렬된 RSSI 값

```
TimeStamp, B1, B2, B3, ..., Zone
01:36:43, 0, -76, 0, 0, ..., A_1
...
```

- 특징
 - 각 비콘(MAC Address)에 해당하는 RSSI 값을 각각 B1, B2, B3로 표현
 - 각 행은 하나의 타임스탬프에서 모든 비콘의 RSSI 값 기록
- 장점
 - 각 타임스탬프에서 모든 비콘의 상태를 한 번에 보여주므로 슬라이딩 윈도우 방식이나 특정 구간의 데이터를 묶어서 학습 가능
- 적합한 경우
 - 여러 비콘의 신호를 한 번에 처리
 - 특정 시간 구간 내에서 변화 패턴 학습 시 유리
 - 여러 비콘의 신호를 한 번에 분석하고 예측하는 경우

Random Forest 학습

- 수집한 데이터셋의 80%는 학습 데이터로, 20%는 검증 데이터로 사용한다.
- 10초 동안 수집된 데이터로 하나의 슬라이딩 윈도우를 설정한다.
- 10초 간의 각 비콘에 해당하는 RSSI 값의 평균을 활용하여 하나의 입력으로 사용한다.

TimeStamp	B1	B2	B3	Zone
10:00:01	-70	-75	-80	A_1
10:00:02	-68	-72	-78	A_1
10:00:03	-69	-73	-79	A_1

만약 10초 동안 측정된 RSSI 값이 이와 같을 때, 이 데이터들은 하나의 슬라이딩 윈도우로 처리된다.

- B1의 평균: $(-70 + -68 + -69) / 3 = -69$
- B2의 평균: $(-75 + -72 + -73) / 3 = -73.33$
- B3의 평균: $(-80 + -78 + -79) / 3 = -79$

이와 같이 평균을 내어, 해당 슬라이딩 윈도우의 대표 값을 구한다.

B1	B2	B3	Zone
-69	-73.33	-79	A_1

평균을 구하는 이유?

- 데이터의 변동성을 줄이기 위해

이동과 RSSI 값의 경향성을 보았을 때, 짧은 시간 내에 RSSI 값들은 많이 변동 될 수 있다. 그렇기에 각 비콘 신호의 평균 값을 활용함으로써 해당 구간에 대한 신호 강도의 전반적인 경향을 더 잘 반영될 수 있도록 한다.

- 모델 학습의 효율성을 높이기 위해

각 순간의 개별 데이터를 학습하는 대신, 일정 구간 내의 대표 값을 사용하여 모델의 복잡성을 줄이도록 하며, 연속된 데이터를 처리하도록 한다.

- 해당 10초 간의 데이터를 바탕으로 하나의 Zone을 예측할 목표 값으로 설정한다.
- 이동 간격을 3으로 설정하여, 3개의 데이터 포인트를 이동하여 슬라이딩 윈도우를 구성한다.

[결과]

Best Parameters Found: {'max_depth': 20, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 2}
검증 정확도: 0.73

분류 보고서:

	precision	recall	f1-score	support
A_1	0.84	0.94	0.89	34
A_2	0.86	0.58	0.69	33
A_3	0.65	0.93	0.77	30
B_1	0.95	0.68	0.79	28
B_2	0.79	0.93	0.86	29
C_1	0.83	0.80	0.81	30
C_2	0.81	0.79	0.80	33
C_3	0.68	0.72	0.70	29
D_1	0.63	0.48	0.55	25
D_2	0.56	0.52	0.54	29
E_1	0.84	0.90	0.87	30
E_2	0.59	0.65	0.62	31
E_3	0.52	0.52	0.52	27
accuracy			0.73	388
macro avg	0.74	0.73	0.72	388
weighted avg	0.74	0.73	0.73	388

모델 테스트

- 학습을 통해 저장한 모델을 불러온다.
- 테스트 데이터를 학습 때와 동일하게 10초 동안 수집된 데이터를 그룹화하여 슬라이딩 윈도우를 설정한다.
- 해당 슬라이딩 윈도우의 RSSI 값들을 평균내어 모델에 입력으로 사용한다.
- 테스트 데이터를 모델에 입력하여 예측된 Zone 값을 얻는다.
- 예측된 Zone 값과 실제 Zone 값을 비교하여 정확도를 계산한다.

```
Processing file: transformed_A_1_testdata.csv
Predictions for transformed_A_1_testdata.csv:
Window 1: Predicted Zone = A_1, True Zone = A_1
Window 2: Predicted Zone = A_1, True Zone = A_1
Window 3: Predicted Zone = A_1, True Zone = A_1
Window 4: Predicted Zone = A_1, True Zone = A_1
Window 5: Predicted Zone = A_1, True Zone = A_1
Window 6: Predicted Zone = A_1, True Zone = A_1
Window 7: Predicted Zone = A_1, True Zone = A_1
Window 8: Predicted Zone = A_1, True Zone = A_1
Window 9: Predicted Zone = A_1, True Zone = A_1
Window 10: Predicted Zone = A_1, True Zone = A_1
```

Accuracy for transformed_A_1_testdata.csv: 1.00

Processing file: transformed_A_2_testdata.csv
Predictions for transformed_A_2_testdata.csv:
Window 1: Predicted Zone = A_1, True Zone = A_2
Window 2: Predicted Zone = A_2, True Zone = A_2
Window 3: Predicted Zone = A_2, True Zone = A_2
Window 4: Predicted Zone = A_2, True Zone = A_2
Window 5: Predicted Zone = A_2, True Zone = A_2
Window 6: Predicted Zone = A_2, True Zone = A_2
Window 7: Predicted Zone = A_2, True Zone = A_2
Window 8: Predicted Zone = A_2, True Zone = A_2
Accuracy for transformed_A_2_testdata.csv: 0.88

Processing file: transformed_A_3_testdata.csv
Predictions for transformed_A_3_testdata.csv:
Window 1: Predicted Zone = A_3, True Zone = A_3
Window 2: Predicted Zone = A_3, True Zone = A_3
Window 3: Predicted Zone = A_3, True Zone = A_3
Window 4: Predicted Zone = A_3, True Zone = A_3
Window 5: Predicted Zone = A_3, True Zone = A_3
Window 6: Predicted Zone = B_1, True Zone = A_3
Window 7: Predicted Zone = A_3, True Zone = A_3
Window 8: Predicted Zone = A_2, True Zone = A_3
Window 9: Predicted Zone = A_3, True Zone = A_3
Accuracy for transformed_A_3_testdata.csv: 0.78

Processing file: transformed_B_1_testdata.csv
Predictions for transformed_B_1_testdata.csv:
Window 1: Predicted Zone = B_2, True Zone = B_1
Window 2: Predicted Zone = B_2, True Zone = B_1
Window 3: Predicted Zone = B_1, True Zone = B_1
Window 4: Predicted Zone = B_2, True Zone = B_1
Window 5: Predicted Zone = B_2, True Zone = B_1
Window 6: Predicted Zone = B_1, True Zone = B_1
Window 7: Predicted Zone = B_1, True Zone = B_1
Window 8: Predicted Zone = A_1, True Zone = B_1
Window 9: Predicted Zone = A_3, True Zone = B_1
Window 10: Predicted Zone = B_1, True Zone = B_1
Window 11: Predicted Zone = B_1, True Zone = B_1
Window 12: Predicted Zone = B_1, True Zone = B_1
Window 13: Predicted Zone = B_1, True Zone = B_1
Window 14: Predicted Zone = A_1, True Zone = B_1
Accuracy for transformed_B_1_testdata.csv: 0.50

Processing file: transformed_B_2_testdata.csv
Predictions for transformed_B_2_testdata.csv:
Window 1: Predicted Zone = B_1, True Zone = B_2
Window 2: Predicted Zone = B_2, True Zone = B_2
Window 3: Predicted Zone = B_2, True Zone = B_2
Window 4: Predicted Zone = B_2, True Zone = B_2
Window 5: Predicted Zone = B_2, True Zone = B_2
Window 6: Predicted Zone = B_2, True Zone = B_2
Window 7: Predicted Zone = B_2, True Zone = B_2
Window 8: Predicted Zone = C_1, True Zone = B_2
Window 9: Predicted Zone = B_2, True Zone = B_2
Window 10: Predicted Zone = B_2, True Zone = B_2
Window 11: Predicted Zone = B_2, True Zone = B_2

Window 12: Predicted Zone = B_2, True Zone = B_2
Window 13: Predicted Zone = A_3, True Zone = B_2
Window 14: Predicted Zone = B_2, True Zone = B_2
Accuracy for transformed_B_2_testdata.csv: 0.79

Processing file: transformed_C_1_testdata.csv
Predictions for transformed_C_1_testdata.csv:
Window 1: Predicted Zone = C_1, True Zone = C_1
Window 2: Predicted Zone = C_1, True Zone = C_1
Window 3: Predicted Zone = C_1, True Zone = C_1
Accuracy for transformed_C_1_testdata.csv: 1.00

Processing file: transformed_C_2_testdata.csv
Predictions for transformed_C_2_testdata.csv:
Window 1: Predicted Zone = C_2, True Zone = C_2
Window 2: Predicted Zone = C_2, True Zone = C_2
Window 3: Predicted Zone = C_2, True Zone = C_2
Accuracy for transformed_C_2_testdata.csv: 1.00

Processing file: transformed_C_3_testdata.csv
Predictions for transformed_C_3_testdata.csv:
Window 1: Predicted Zone = C_3, True Zone = C_3
Window 2: Predicted Zone = C_3, True Zone = C_3
Accuracy for transformed_C_3_testdata.csv: 1.00

Processing file: transformed_D_1_testdata.csv
Predictions for transformed_D_1_testdata.csv:
Window 1: Predicted Zone = D_1, True Zone = D_1
Window 2: Predicted Zone = D_1, True Zone = D_1
Window 3: Predicted Zone = D_1, True Zone = D_1
Window 4: Predicted Zone = D_1, True Zone = D_1
Window 5: Predicted Zone = D_1, True Zone = D_1
Window 6: Predicted Zone = E_3, True Zone = D_1
Window 7: Predicted Zone = D_2, True Zone = D_1
Window 8: Predicted Zone = D_2, True Zone = D_1
Window 9: Predicted Zone = D_2, True Zone = D_1
Window 10: Predicted Zone = E_3, True Zone = D_1
Window 11: Predicted Zone = D_1, True Zone = D_1
Accuracy for transformed_D_1_testdata.csv: 0.55

Processing file: transformed_D_2_testdata.csv
Predictions for transformed_D_2_testdata.csv:
Window 1: Predicted Zone = D_1, True Zone = D_2
Window 2: Predicted Zone = D_1, True Zone = D_2
Window 3: Predicted Zone = D_2, True Zone = D_2
Window 4: Predicted Zone = D_1, True Zone = D_2
Window 5: Predicted Zone = C_3, True Zone = D_2
Window 6: Predicted Zone = C_3, True Zone = D_2
Window 7: Predicted Zone = D_1, True Zone = D_2
Window 8: Predicted Zone = D_1, True Zone = D_2
Window 9: Predicted Zone = D_1, True Zone = D_2
Window 10: Predicted Zone = D_2, True Zone = D_2
Accuracy for transformed_D_2_testdata.csv: 0.20

Processing file: transformed_E_1_testdata.csv
Predictions for transformed_E_1_testdata.csv:
Window 1: Predicted Zone = E_2, True Zone = E_1
Window 2: Predicted Zone = E_1, True Zone = E_1

Window 3: Predicted Zone = E_1, True Zone = E_1
 Window 4: Predicted Zone = E_1, True Zone = E_1
 Window 5: Predicted Zone = E_1, True Zone = E_1
 Window 6: Predicted Zone = E_1, True Zone = E_1
 Window 7: Predicted Zone = E_1, True Zone = E_1
 Accuracy for transformed_E_1_testdata.csv: 0.86

Processing file: transformed_E_2_testdata.csv
 Predictions for transformed_E_2_testdata.csv:
 Window 1: Predicted Zone = E_3, True Zone = E_2
 Window 2: Predicted Zone = E_2, True Zone = E_2
 Window 3: Predicted Zone = E_2, True Zone = E_2
 Window 4: Predicted Zone = E_2, True Zone = E_2
 Window 5: Predicted Zone = E_2, True Zone = E_2
 Window 6: Predicted Zone = E_1, True Zone = E_2
 Accuracy for transformed_E_2_testdata.csv: 0.67

Processing file: transformed_E_3_testdata.csv
 Predictions for transformed_E_3_testdata.csv:
 Window 1: Predicted Zone = D_1, True Zone = E_3
 Window 2: Predicted Zone = E_3, True Zone = E_3
 Window 3: Predicted Zone = E_3, True Zone = E_3
 Window 4: Predicted Zone = E_3, True Zone = E_3
 Window 5: Predicted Zone = E_3, True Zone = E_3
 Window 6: Predicted Zone = E_3, True Zone = E_3
 Accuracy for transformed_E_3_testdata.csv: 0.83

▼ 결과 그래프

