Sample proposal

Lecture 21

Changho Suh

January 30, 2024

Outline

Will investigate the last sample proposal:

3. 센서 데이터를 활용한 차량 이상감지 (autoencoder 활용)

Will study python packages for other machine learning techniques.

Sample proposal #3

센서 데이터를 활용한 차량 이상감지

홍길동 / 책임연구원

Mar. 12, 2021

실시간 모니터링 센서

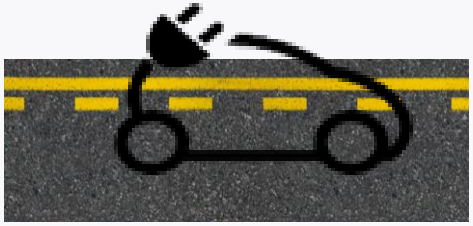
목표: 다양한 센서신호로 부터 **차량 이상**을 감지

센서 데이터

언덕길

커브길

내리막



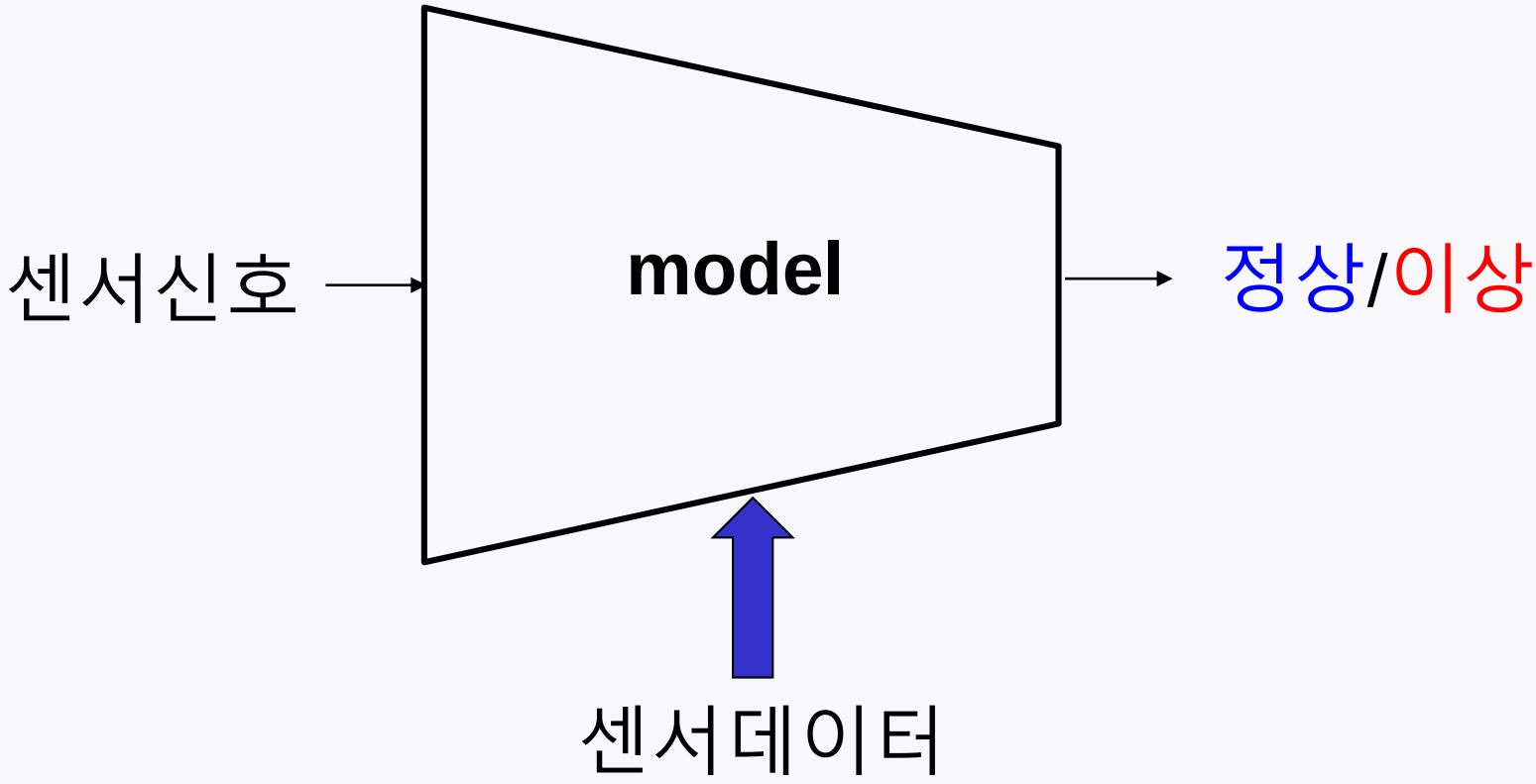
주행상황

고속도로

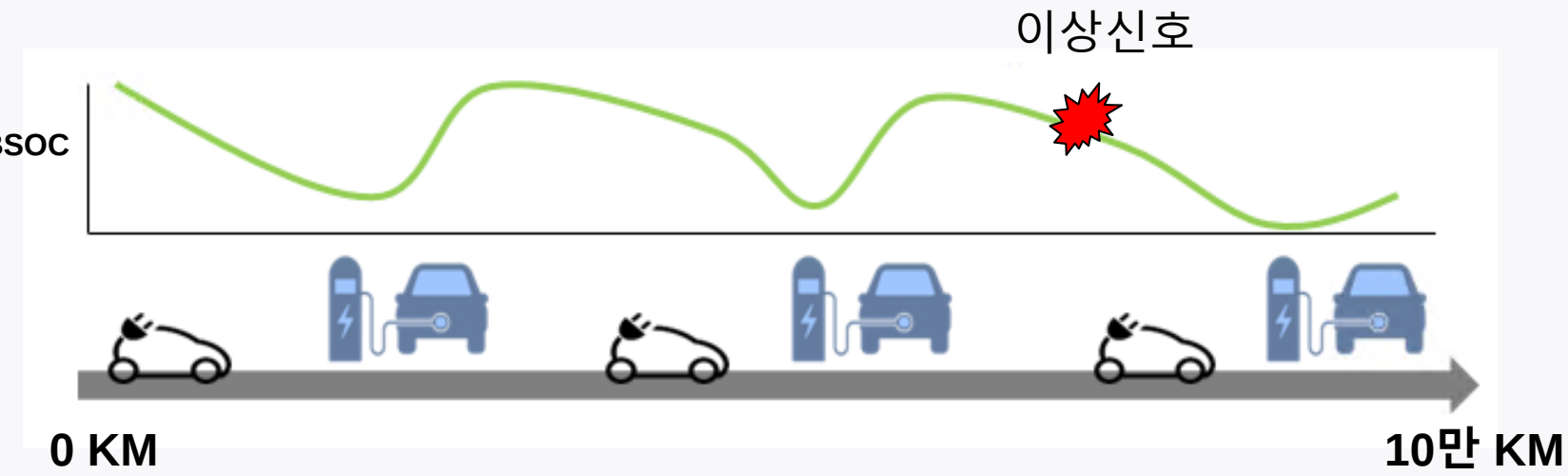
도심/교차로

다양한 주행 상황에 대한 센서데이터를 보유

센서신호로 부터 정상/이상 판단



Challenge

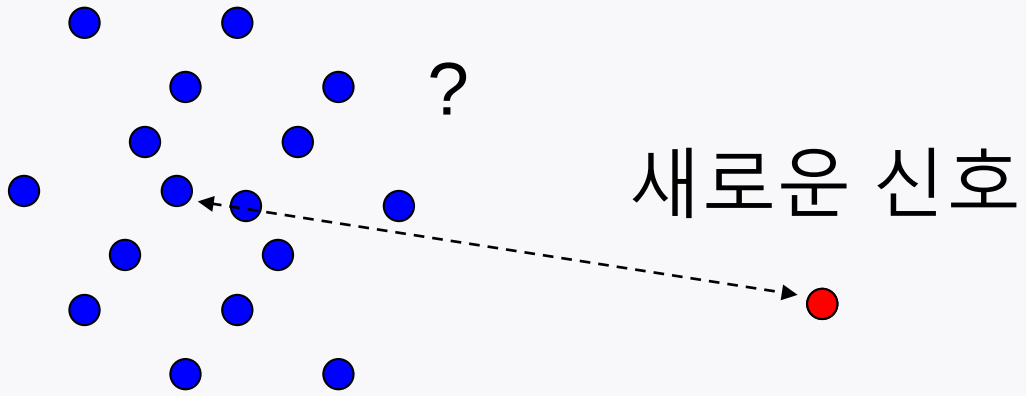


"이상"신호 관련 example 부족

지도학습이 어려움

정상 센서 신호 분포

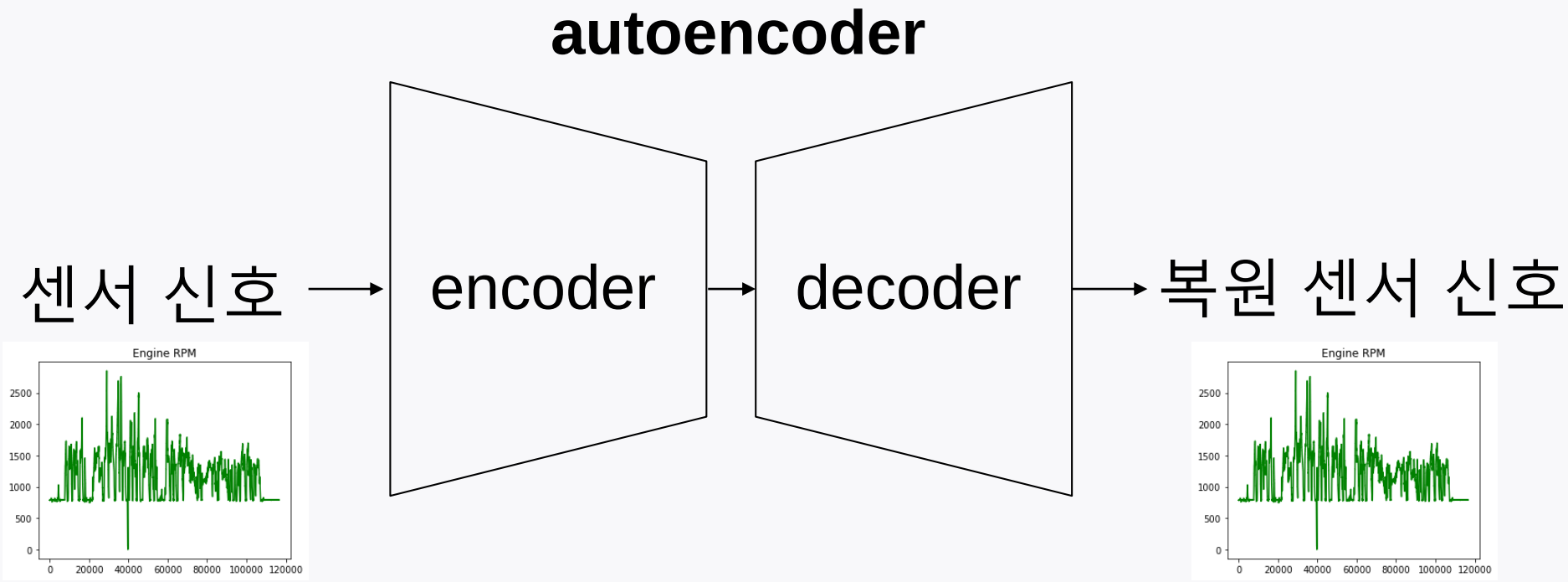
- : 정상 신호



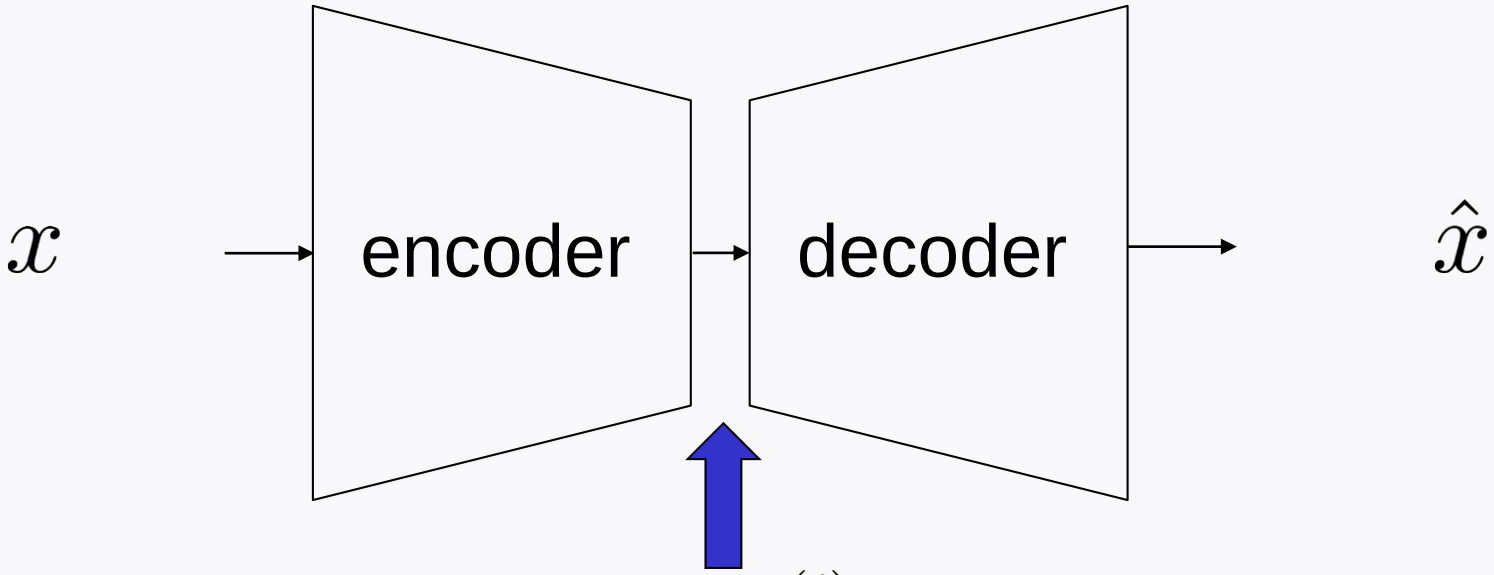
정상 신호의 분포를 **학습**

➡ 분포차이로 차량 이상 감지

정상신호로만 autoencoder 학습시킴



학습방법

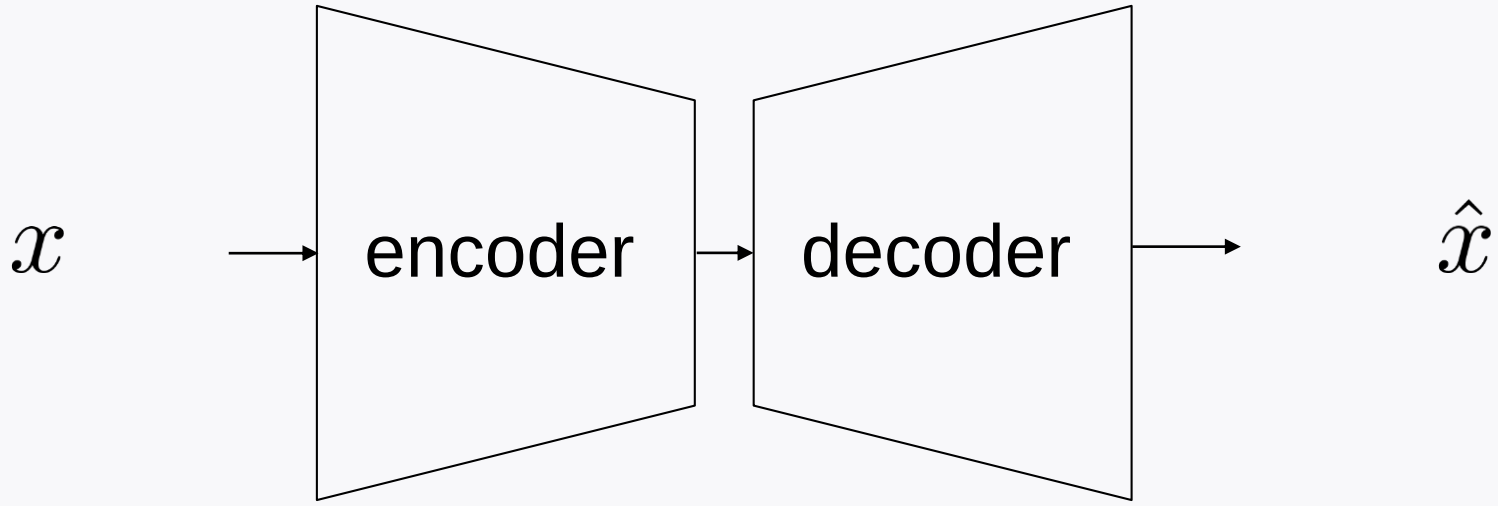


data $\{x^{(i)}\}_{i=1}^{m}$

$$\min_{w} \frac{1}{m} \sum_{i=1}^{m} (x^{(i)} - \hat{x}^{(i)})^2$$

reconstruction loss

이상감지 방법



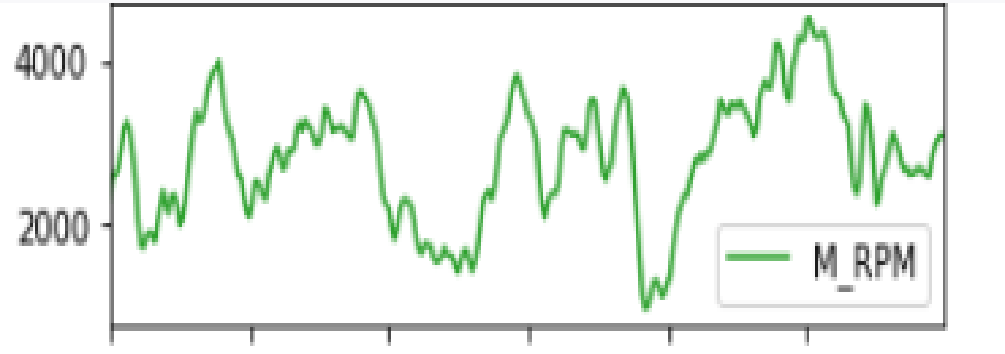
reconstruction loss < threshold ➡ 정상

reconstruction loss > threshold ➡ 이상

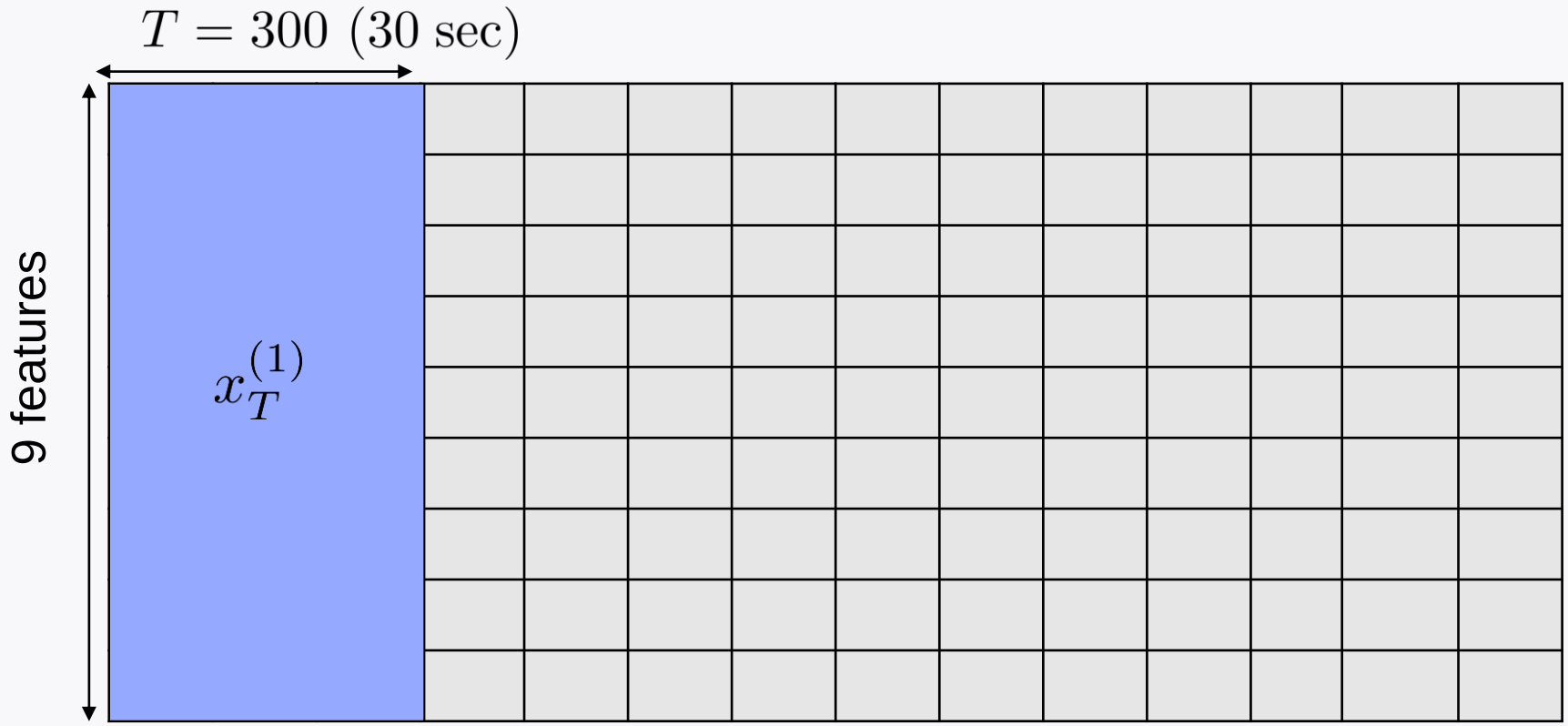
Raw data

9개의 시계열 **센서 데이터**

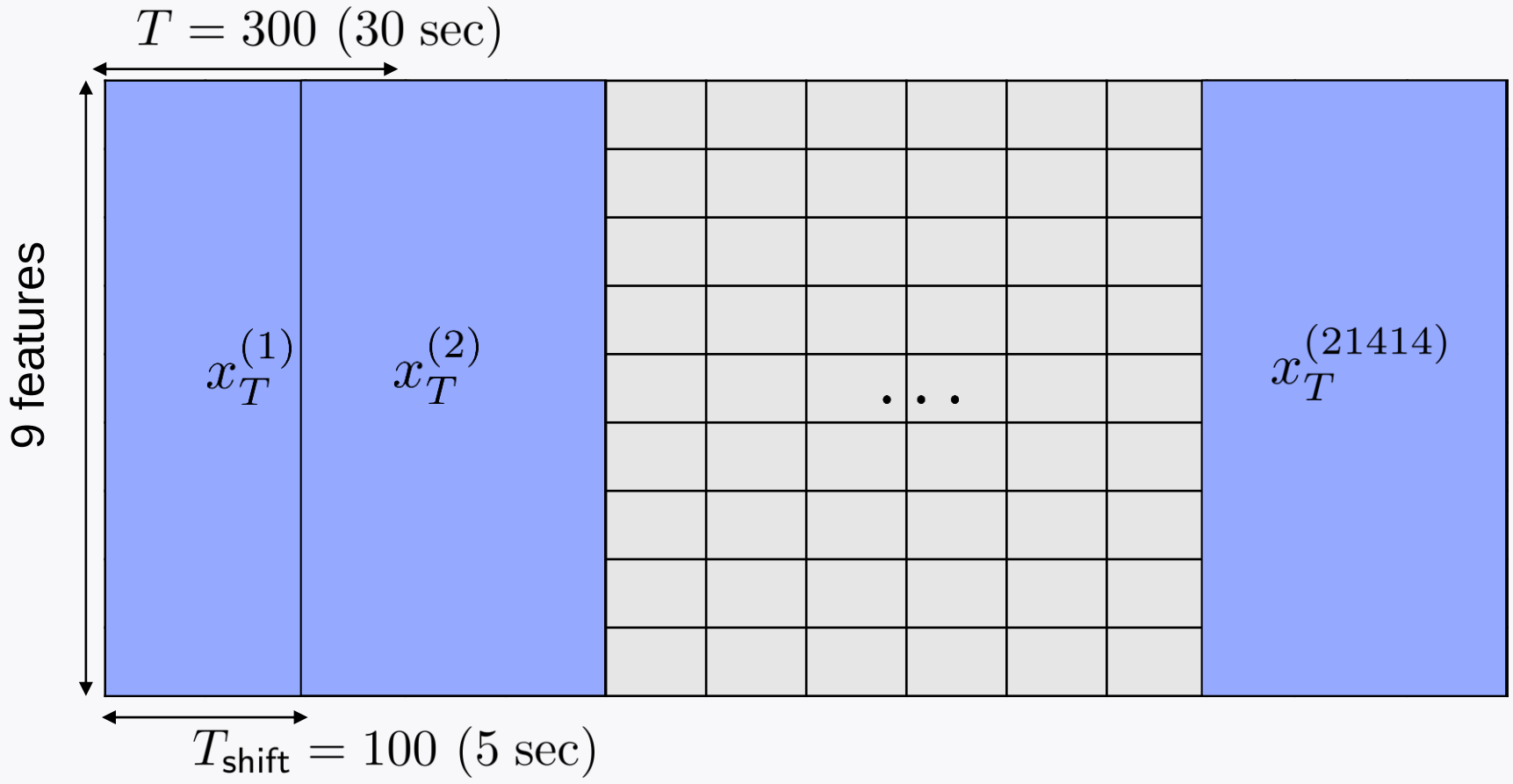
Ex) 모터 RPM



Data preprocessing



Data preprocessing



Dataset

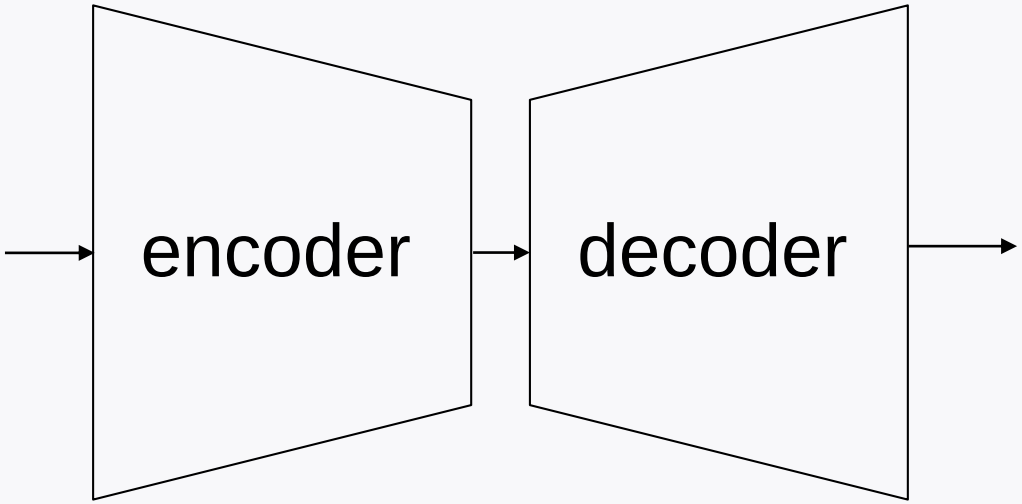
of examples: 21,414

data: 9개의 시계열 센서 데이터

* **dimension**: 300 (30 sec) × 9

label: 없음 (비지도학습)

Model



Encoder: DNN (or 1d CNN)

Decoder: DNN (or 1d transposed CNN)

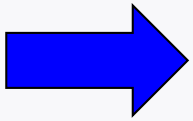
Target performance

Autoencoder: Normalized RMSE

정상/이상 분류기: **Accuracy**

Impact

1. 차량의 문제를 사전 예측 및 조치 (**고객 만족도** 증가)



브랜드 이미지 향상

2. 중고차 **잔존가치** 향상

Confusion matrix

		prediction: Positive (P)	prediction: Negative (N)
actual	T	True Positive (TP)	False Negative (FN)
	F	False Positive (FP)	True Negative (TN)

Total population = T+F

Two types of error

		prediction: Positive (P)	prediction: Negative (P)
actual	T	True Positive (TP)	False Negative (FN)
	F	False Positive (FP)	True Negative (TN)

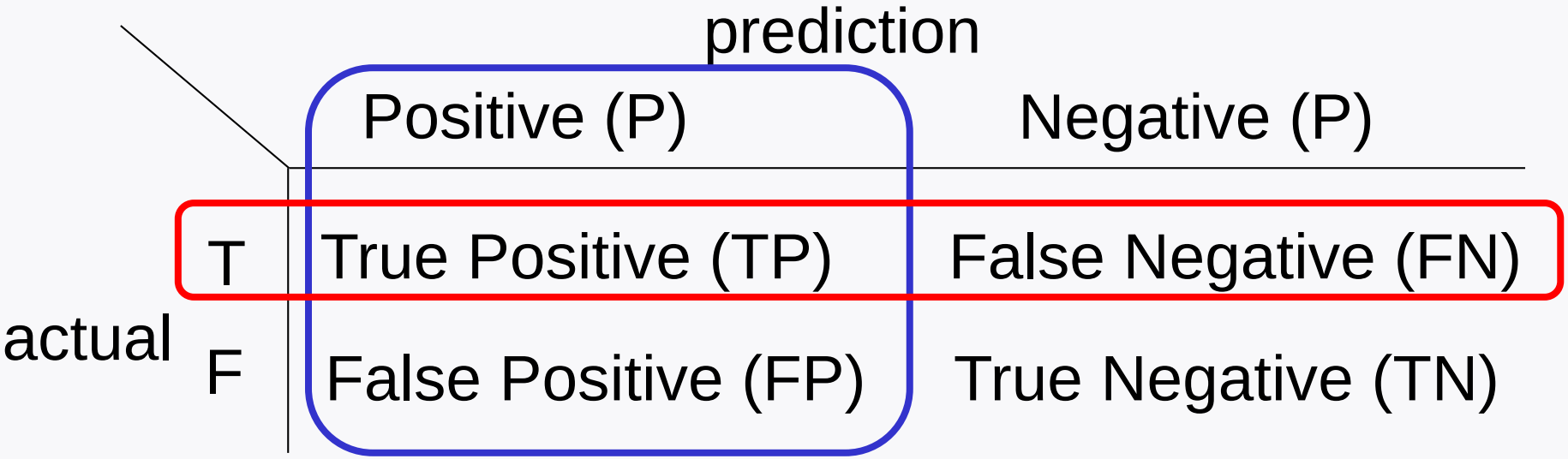
Type I error: False Positive Rate (FPR) $= \frac{\mathrm{FP}}{\mathrm{F}}$

Type II error: False Negative Rate (FNR) $= \frac{\mathrm{FN}}{\mathrm{T}}$

Fire alarm: Important to reduce FNR.

Criminal judge: Important to reduce FPR.

Precision & recall



$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$$ (How accurate test result is)

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

$$\text{F1 score} = \frac{2}{\frac{1}{\text{Precision}} + \frac{1}{\text{Recall}}}$$

이상감지모델: 다른 사례들

자동변속감 OK/NG 판정 모델

사이드미러 OK/NG 판정 모델

주행상태 정상/비정상 분류기

서브마린 발생 여부 판정 모델

댐퍼 누유 정상/비정상 분류기

전동화 차량의 모터시스템 이상감지모델

Python packages for other machine learning techniques

Unsupervised learning

1. Clustering

2. Principal component analysis (PCA), autoencoder

3. Generative Adversarial Networks (GANs)

Clustering

1. K-means

```
from sklearn.cluster import KMeans
```

2. K-medoids

```
from sklearn_extra.cluster import  KMedoids
pip install scikit-learn-extra
```

3. Hierarchical cluster (agglomerative clustering)

```
from sklearn.cluster import AgglomerativeClustering
```

PCA, autoencoder, t-SNE

1. PCA

```
from sklearn.decomposition import PCA
from sklearn.decomposition import KernelPCA
```

2. Autoencoder

```
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input
from tensorflow.keras.layers import Dense
```

3. t-SNE

```
from sklearn.manifold import TSNE
```

GANs

```
from tensorflow.keras.models import Model
from tensorflow.keras.layers import Input
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import BatchNormalization
```

Small data techniques

1. Semi-supervised learning

2. Transfer learning

3. Simulator-based learning

Semi-supervised learning

```
from sklearn.semi_supervised import LabelSpreading
```

Transfer learning

```
from tensorflow.keras.applications import Xception
```

```
base_model = Xception(weights='imagenet',
                      input_shape=(150,150,3),
                      include_top=False)
```


whether to include the fully-connected layer at the top of the network

```
base_model.trainable = False
```