

# 見守りシステムのためのクラスラベルを用いたGANによる教師なし異常検知

○皆本啓太 半田久志 (近畿大学)

## Unsupervised Anomaly Detection by GAN Using Class Label for Watching System

\*K. Minamoto and H. Handa (Kindai University)

**Abstract**— In this paper, we propose an anomaly detection method that considers the data class label for a monitoring system based on power consumption. Specifically, by considering the usual power usage pattern as normal data and detecting the case where the daily power usage pattern is different from the past one, it is possible to detect a day when the daily rhythm is different from usual. If there is a difference in the daily rhythm depending on the day of the week, the power usage pattern may differ for each day of the week. Therefore, we propose an anomaly detection method that considers the day of the week by using GAN that uses class labels. As a result of conducting an experiment using the power consumption data obtained from the TV at the elderly's home, we were able to detect the abnormality with a certain accuracy by the proposed method.

**Key Words:** 異常検知, GAN, AnoGAN, 見守りシステム

### 1 序論

1970年頃から日本国では高齢化が進んでおり、高齢者の安否を遠隔から確認するための、見守りシステムに対する需要が高まっている。見守りシステムの例としてカメラを利用したものがあるが、プライバシーを侵害することに繋がる恐れがある<sup>1)</sup>。また、赤外線センサを利用した見守りシステムもあるが、ペットなどに対して誤作動を起こす恐れがある<sup>2)</sup>。加えて、上記で述べたような見守りシステムは高齢者が監視されている感覚を覚えることで、心理的な負担を感じる問題点がある。

そこで、テレビなどの家電の消費電力を用いて異常検知を行うことで、高齢者の心理的負担に配慮した見守りシステムを構築する。見守りシステムはテレビなどの家電のコンセントにセンサを接続して消費電力を計測し、得られたデータをもとに高齢者の現在の状態を親族や介護事業者に通知する。具体的には、親族や介護事業者のモバイル端末に専用のアプリをインストールしてもらい、アプリの画面にて高齢者の状態を確認してもらおうシステムである。高齢者の現在の状態は正常、生活リズムが普段と異なる、電源が付けられたままの状況が続くなど明らかに異常な状態の3段階で評価する。これに通信不良の場合を含めた4種類の通知をアプリ側で行う。本研究では、正常な状態と生活リズムが普段と異なる状態を判別する手法について提案する。

本研究では普段の電力の使用パターンを正常なデータと考えて、1日の電力の使用パターンが過去のものとは異なる場合を検知することで、普段と生活リズムが異なる日の検出を行う。Fig. 1は本研究で扱うデータの例であり、1日の消費電力の12分ごとの推移である。Fig. 1の横軸は時間、縦軸は消費電力を表している。Fig. 1のような消費電力の波形に対して、1-Dimensional Convolutional Neural Network (1-DCNN)を適用して特徴を抽出することで異常検知を行う<sup>3)</sup>。

曜日によって生活リズムに違いがある場合、曜日ごと

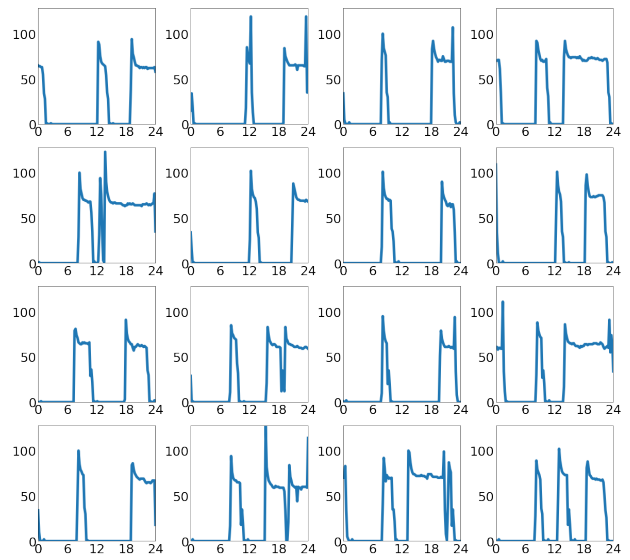


Fig. 1: 本研究で扱うデータの例

に電力の使用パターンに違いが現れることがある。例えば、平日は予定がありテレビを視聴していない時間帯でも、日曜日は予定がない視聴している場合が考えられる。そのため、単に普段の電力の使用パターンの分布から外れ値を検出する手法を用いると、特定の曜日に着目した際に普段と生活リズムが異なる日を正常と判定してしまう場合がある。また、曜日ごとにデータを分けて学習を行う方法が考えられるがデータ数が不足する問題がある。

そこで、敵対的生成ネットワーク (Generative Adversarial Networks: GAN)<sup>4)</sup>を用いた異常検知手法であるAnoGAN (Anomaly Detection with GAN)<sup>5)</sup>に対して、クラスラベルを用いたGANであるAC-GAN (Auxiliary Classifier GAN)<sup>6)</sup>の構造を応用することで、データのクラスラベルを考慮した異常検知手法を提案する。また、現実には必ずしも曜日ごとに生活リズムが異な

るとは限らない。そこで、混合ガウスモデル (Gaussian Mixture Model: GMM) によるクラスタリング<sup>7)</sup>を利用して、生活リズムが類似している曜日同士を同一のクラスとするアルゴリズムを提案する。そして、得られたクラスをクラスラベルとして異常検知を行う。実際に高齢者の自宅のテレビから得られたデータを用いて、実験を行うことで提案手法について検証する。

## 2 提案手法

### 2.1 GAN による異常検知

本研究では、学習した AC-GAN のモデルを AnoGAN に適用するために、Generator にクラスラベルを表す条件ベクトル  $\mathbf{y}$  を入力できるように改良する。提案手法により異常検知を行う手順を以下に示す。

1. 正常なサンプルを用いて AC-GAN を学習。
2. Generator, Discriminator の重みは固定して、判定を行うデータ  $\mathbf{x}$  を再構成するような、Generator の入力  $\mathbf{z}_\gamma$  を探索。
3.  $\mathbf{x}$  と再構成したデータ  $G(\mathbf{z}_\gamma, \mathbf{y})$  を比較し、その誤差の大きさを異常度として異常を判定。

提案手法の損失関数は Residual Loss と Discrimination Loss の 2 つを合わせたものである。Residual Loss は式 (1) に示すように、判定を行うデータ  $\mathbf{x}$  と生成データ  $G(\mathbf{z}_\gamma, \mathbf{y})$  の差の合計である。Discrimination Loss は式 (2) のように、 $\mathbf{x}$  と  $G(\mathbf{z}_\gamma, \mathbf{y})$  の Discriminator の中間層の出力  $f(\cdot)$  の差の合計である。

$$\mathcal{L}_R(\mathbf{z}_\gamma, \mathbf{y}) = \sum |\mathbf{x} - G(\mathbf{z}_\gamma, \mathbf{y})| \quad (1)$$

$$\mathcal{L}_D(\mathbf{z}_\gamma, \mathbf{y}) = \sum |f(\mathbf{x}) - f(G(\mathbf{z}_\gamma, \mathbf{y}))| \quad (2)$$

提案手法の損失関数は Residual Loss と Discrimination Loss を合わせて、式 (3) のようになる。異常度は学習後の  $\mathcal{L}(\mathbf{z}_\gamma, \mathbf{y})$  によって定義される。また、 $\lambda$  は  $\mathcal{L}_D(\mathbf{z}_\gamma, \mathbf{y})$  の割合を決定するハイパーパラメータである。

$$\mathcal{L}(\mathbf{z}_\gamma, \mathbf{y}) = (1 - \lambda) \cdot \mathcal{L}_R(\mathbf{z}_\gamma, \mathbf{y}) + \lambda \cdot \mathcal{L}_D(\mathbf{z}_\gamma, \mathbf{y}) \quad (3)$$

### 2.2 前処理

#### 2.2.1 データの除去

本研究では、テレビなどの家電にセンサを設置して得られた消費電力を利用して異常検知を行う。しかし、日によっては 1 日の間に 1 度もその家電を利用していない日もある。また、稀に実際に利用した消費電力とは考えられない、異常に高い値が計測されることがある。そこで、1 日の間に 1 度も利用していない日のデータは除去する。また、平均値を  $\mu$ 、標準偏差を  $\sigma$  とし、1 日の間に 1 度も  $\mu + 4\sigma$  を超える日のデータも除去する。

#### 2.2.2 Data Augmentation

ディープラーニングは学習に大量のデータを要するが、必ずしも大量のデータが手に入るとは限らない。そこで、訓練データに対して人工的な操作を行うことでデータを拡張する Data Augmentation と呼ばれる手法が用いられている。Data Augmentation は主に画像に対して用いられており、画像に対して移動、回転、反

転、拡大・縮小、ランダムノイズの付与などを行いデータ数を水増しする<sup>8)</sup>。本研究では、1%以内の波形の並行移動処理とガウス分布からランダムにサンプリングした値の付与を行う。ランダムノイズの付与は、式 (4) のように消費電力が計測された時間帯のデータに対して  $\epsilon$  を付与する Data Augmentation を行う。ただし、 $\epsilon$  は訓練データ中の消費電力が計測された時間帯のデータの標準偏差を  $\sigma$  とし、ガウス分布  $N(0, \sigma^2)$  からランダムにサンプリングした値に 0.01 を乗じた値とした。

$$x \leftarrow \begin{cases} 0, & x = 0 \\ x + \epsilon, & x \neq 0 \end{cases} \quad (4)$$

#### 2.2.3 GMM を用いた最適なクラスラベルの探索

本研究では、曜日、祝日を用いてラベル付を行う。具体的には、ある日のデータが祝日のデータの場合、祝日にラベル付する。祝日でない場合、その日の曜日でラベル付を行う。つまり、曜日、祝日によって 8 クラスにラベル付を行う。

しかし、必ずしも曜日、祝日ごとに電力の使用パターンが異なるとは限らないため、使用パターンが類似しているもの同士を同一のクラスとしてラベル付を行うための、適切なクラスを求めるアルゴリズムを提案する。例えば、{月, 火, 金}, {水, 木}, {土, 日, 祝} のように 3 クラスにラベル付を行うパターンが考えられる。

具体的には、GMM によるクラスタリング<sup>7)</sup>を行い、曜日、祝日ごとに最も所属確率の高いクラスに割り当てることでクラスを得る。この処理をクラス数 2 から 8 まで行い、cross entropy loss<sup>9)</sup> が最も低いときの結果を、最適なクラスとしてラベル付を行う。そして、得られたクラスラベルを用いてモデルの学習を行う。また、いずれのクラス数においても、クラスが属していないクラスが存在している場合は、適切なクラスなしと判定する。その場合は通常の AnoGAN を用いる。最適なクラスラベルを探索するアルゴリズムを Algorithm1 に示す。ただし、 $\text{gmm}(\mathbf{X}, c)$  は学習データに対してクラス数  $c$  で GMM<sup>7)</sup> によるクラスタリングを行い、各クラスへの所属確率を得ることを表す。また、 $\text{assign}(\mathbf{X}, p)$  はクラスタリング結果  $p$  を用いて、 $\mathbf{X}$  の各クラスごとに最も所属確率の高いクラスに割り当てることで、クラスを得ることを表す。

### 2.3 ハイパーパラメータ

#### 2.3.1 活性化関数

活性化関数はニューラルネットワークの層において、線形変換された入力に非線形性を導入し、モデルの表現力を上げる役割を果たす関数である。本研究では、中間層の活性化関数として Mish 関数<sup>10)</sup> を用いた他、Discriminator の出力層に Softmax 関数<sup>11)</sup> と Sigmoid 関数<sup>12)</sup> を用いた。また、Generator の出力層には、入力  $x$  に対して  $\max(0, x)$  とする Relu 関数と softsign 関数<sup>13)</sup> を改良して組み合わせた式 (6) のような関数を用いた。ただし  $\alpha$  はハイパーパラメータであり、本研究では  $\alpha = 0.1$  とした。以降、この関数を Alphasign とする。

$$f(x) = \begin{cases} 0, & x \leq 0 \\ \frac{x}{\frac{1}{\alpha} + |x|}, & x > 0 \end{cases} \quad (6)$$

---

**Algorithm 1** Search the best combination

---

**Input:**  $\mathbf{X}$ : Train data**Output:** best combination

```
1: set  $c \leftarrow 2, b \leftarrow -1$ 
2: while do
3:    $p \leftarrow \text{gmm}(\mathbf{X}, c)$ 
4:    $q \leftarrow \text{assign}(\mathbf{X}, p)$ 
5:   if All clusters have some labels then
6:      $e \leftarrow$  The cross entropy loss  $p, q$ 
           
$$H(p, q) = - \sum_x q(x) \log(p(x)) \quad (5)$$

7:     if  $b < e$  or  $b < 0$  then
8:        $b \leftarrow e$ 
9:        $Q \leftarrow q$ 
10:    end if
11:  end if
12:  if  $c \geq 8$  then
13:    break
14:  end if
15:   $c \leftarrow c + 1$ 
16: end while
17: if  $b < 0$  then
18:  return None combination
19: end if
20: return  $Q$ 
```

---

### 2.3.2 ネットワーク構造

本研究で用いた提案手法のモデルのアーキテクチャと学習時のハイパーパラメータの詳細を Table 1 に示す。Table 1 中の Output は出力層のサイズを、BN は Batch Normalization を表す。また、Layer to search for  $z_\gamma$  は提案手法により 100 次元のランダムノイズ  $z$  から元データを再構成するような Generator の入力  $z_\gamma$  の探索を行う層を表す。 $G(z_\gamma, y)$  は AC-GAN の Generator,  $D(x)$  は Discriminator のネットワークを表す。また、提案手法により  $z_\gamma$  の探索を行うときの iteration は 100 とした。

## 3 実験

本研究では実際に高齢者の自宅のテレビに設置したセンサから取得した、2019 年 2 月 9 日から 2020 年 9 月 3 日までの消費電力のデータを使用した。実験では、2019 年 2 月 9 日から 2020 年 3 月 31 日までの 381 件を学習データ、2020 年 4 月 1 日から 2020 年 9 月 3 日までの 153 件を検証用データとして用いた。各データは、学習データに対して Algorithm1 を適用することによって得られたクラス、 $A = \{\text{月, 火, 金}\}$ ,  $B = \{\text{水, 木}\}$ ,  $C = \{\text{土, 日, 祝}\}$  の 3 クラスのいずれかにラベルを付与して、学習・検証を行った。最小値 0, 最大値 1 で正規化を行ってから学習・検証を行った。学習データ、検証用データのラベルごとの個数を Table 2 に示す。Fig. 2 に各データを UMAP<sup>14)</sup> により 2 次元に削減した散布図を示す。

### 3.1 Generator による生成

提案手法によりモデルが適切に学習していることを検証するため、学習済みの Generator により生成されたデータを確認した。Fig. 3 は Generator により生成

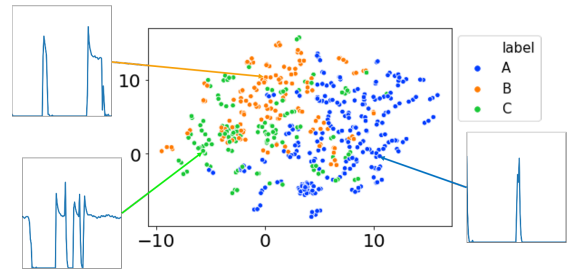


Fig. 2: 本研究で用いたデータを UMAP により 2 次元に削減した散布図

されたデータの例である。Fig. 3 から学習データに類似した波形を生成することに成功していることが分かる。また、多様な波形を生成することに成功しており、モード崩壊と呼ばれる Generator が単一のデータのみを生成する現象も起きていないことから、モデルが適切に学習していることが示された。

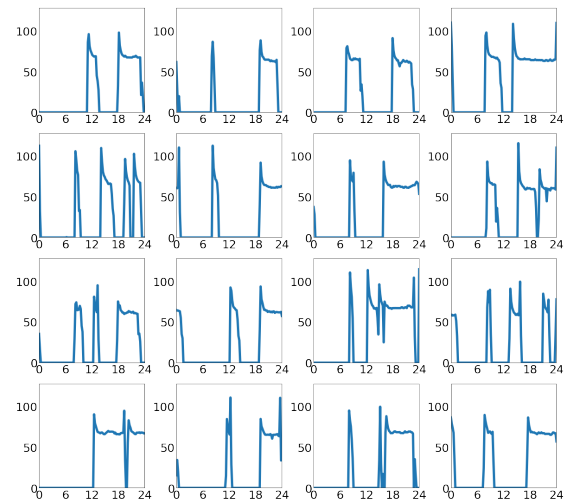


Fig. 3: Generator により生成されたデータ

### 3.2 Discriminator による特徴量抽出

Discriminator による特徴量抽出の有効性について検証するため、2020 年 4 月中のデータ 30 件を用いて、実データと提案手法により生成されたデータ間のユークリッド距離  $\mathcal{L}_R$  と、それぞれのデータを Discriminator に入力して得られる中間層の出力の間の距離  $\mathcal{L}_D$  を求めて比較した。また、提案手法により 30 件のデータを生成する際の損失  $\mathcal{L}$  における Discrimination Loss の割合を決定するハイパーパラメータは  $\lambda = 0.1$  として実験を行った。 $\mathcal{L}_D, \mathcal{L}_R$  をそれぞれの平均値で減算し、標準偏差で割った値を  $d(\mathcal{L}_R), d(\mathcal{L}_D)$  とする。Fig. 4 に 1 日ごとの  $d(\mathcal{L}_R), d(\mathcal{L}_D)$  を示す。

Fig. 4 の 2020 年 4 月 10 日のような  $d(\mathcal{L}_D) - d(\mathcal{L}_R)$  の値が高い日のデータは、実データと Generator により生成されたデータ間の差分は小さいが、Discriminator に入力して得られる中間層の出力はあまり類似していないデータである。一方、2020 年 4 月 20 日のような  $d(\mathcal{L}_D) - d(\mathcal{L}_R)$  の値が低い日のデータは、実データと提案手法により生成されたデータ間の差分は大きい、Discriminator に入力して得られる中間層の出力で比較すると類似しているデータである。Fig. 5 に 2020 年 4 月 10 日の実データと提案手法により生成されたデー

Table 1: 本研究で用いたモデルのアーキテクチャとハイパーパラメータ

Operation	Kernel	Stride	Output	BN	Nonlinearity
Layer to search for $z_\gamma$					
Linear	N/A	N/A	100	×	Tanh
$G(z_\gamma, y)$					
Linear	N/A	N/A	30*256	✓	Mish
Transposed Convolution	5	2	60*64	✓	Mish
Transposed Convolution	15	2	120*1	×	Alphasign
$D(x)$					
Convolution	15	2	60*64	✓	Mish
Convolution	5	2	30*128	✓	Mish
Linear	N/A	N/A	1024	✓	Mish
Linear	N/A	N/A	3&1	×	Softmax&Sigmoid
Optimizer	Adam( $\alpha = 10^{-3}, \beta_1 = 0.9, \beta_2 = 0.999$ )				
Batchsize	32				
Latent dimension	100				
Iteration	3000				
Weight, bias initialization	Isotropic gaussian( $\mu = 0, \sigma = 0.05$ )				

Table 2: 本研究で用いたデータのラベルごとの個数

	A	B	C	合計
学習データ	152	104	125	381
検証用データ	61	41	51	153

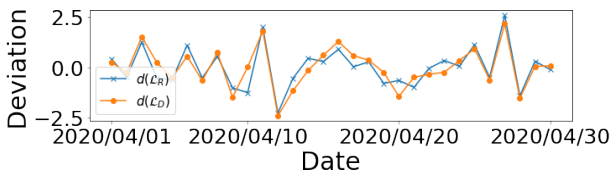


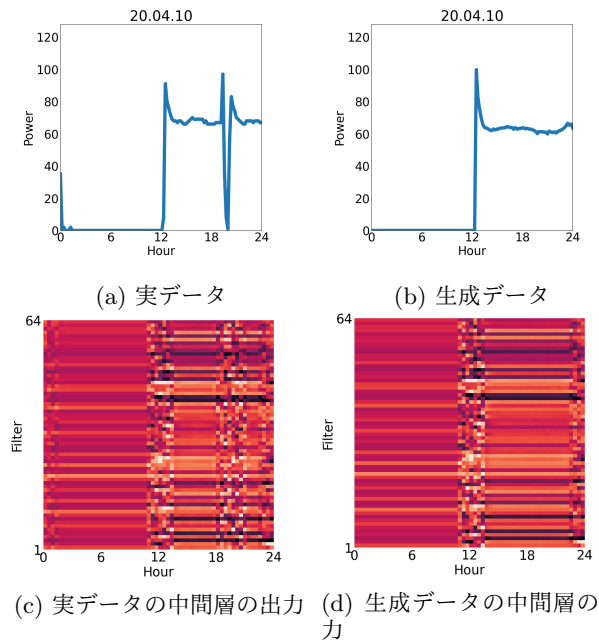
Fig. 4: 2020年4月中の1日ごとの  $d(\mathcal{L}_R)$ ,  $d(\mathcal{L}_D)$ .  $d(\mathcal{L}_D) - d(\mathcal{L}_R)$  の値が高い日程, 実データと生成されたデータ間の差分は小さいが, Discriminator に入力して得られる中間層の出力で比較するとあまり類似していないデータであることを意味する.

タの比較を示す. Fig. 5c, Fig. 5d は, 実データと生成データ, それぞれのデータを Discriminator に入力して得られる中間層の出力のヒートマップである. 同様に Fig. 6 に 2020年4月20日の実データと提案手法により生成されたデータの比較を示す. 10日のデータと20日のデータを比較すると, 実データと生成データの差分は20日のデータの方が大きい, 視覚的には10日の方が異なる波形になっていることが分かる. 一方, それぞれのデータの中間層の出力のヒートマップを比較すると, 20日のデータは実データと生成されたデータで非常に類似したものとなっているが, 10日のデータは異なるものとなっていることがわかる.

これらの結果から Discriminator に入力して得られる中間層の出力を用いることによって, 消費電力データ同士の距離をより正確に計測することができたため, Discriminator による特徴量抽出が有効であることが示された.

### 3.3 異常検知の検証

異常検知の検証を行うため, 検証用データに対して, 元のラベルと異なる誤ったラベルを付与することで擬似的な異常データと定義した. 各ラベルごとに別々に学習させた GAN のモデルを用いて既存手法の AnoGAN

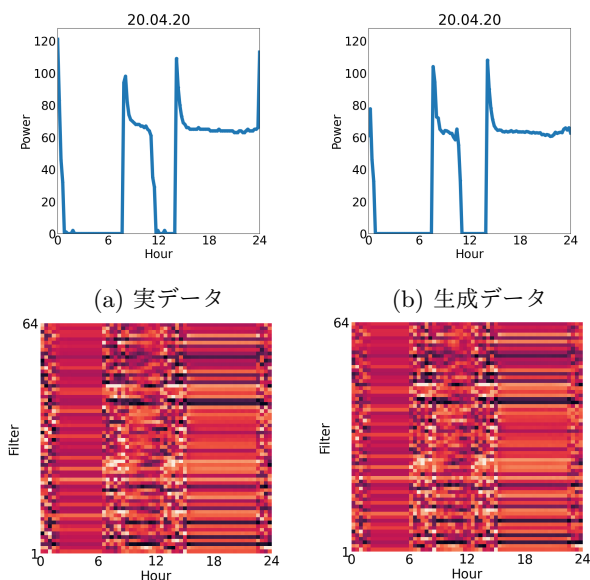


(c) 実データの中間層の出力 (d) 生成データの中間層の出力

Fig. 5: 2020年04月10日のデータと生成データの比較

により異常検知を行った結果と, 提案手法により異常検知を行った結果を比較する. また, 損失  $\mathcal{L}$  において Discrimination Loss の割合を決定するハイパーパラメータ  $\lambda = 0, 0.1, 0.5, 1$  の4通りで実験を行った. Fig. 7 にそれぞれの  $\lambda$  における実験結果の ROC 曲線を示す. また, Table 3 に ROC-AUC 及び, Youden's index<sup>15)</sup> による ROC 曲線上の最適なカットオフ地点を閾値に設定した際の precision, recall, specificity, F1-score を示す (小数点第4位以下切り捨て). 結果として, ROC-AUC と F1-score のいずれも提案手法で  $\lambda = 0.1$  のときに最も高くなった. このときの異常度の分布を Fig. 8 に示す. また,  $\lambda$  が低くなる程 precision, specificity が高くなり,  $\lambda$  が高くなる程 Recall が高くなる傾向が見られた.

この結果から, Residual Loss は precision, specificity を高めることに寄与するのに対して, Discrimination

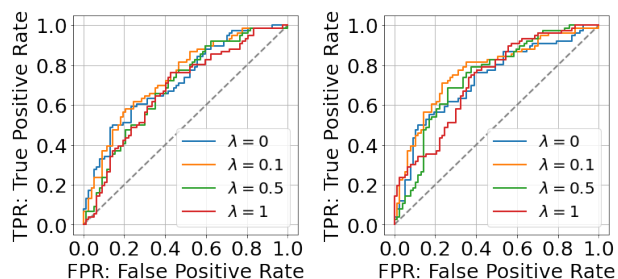


(a) 実データ (b) 生成データ  
(c) 実データの中間層の出力 (d) 生成データの中間層の出力  
Fig. 6: 2020年04月20日の実データと提案手法により生成されたデータの比較

Loss は Recall は高めることに寄与することが示された。また、Discrimination Loss と Residual Loss を両方用いることで、消費電力データ間の距離をより正確に捉えられることが示された。以降の分析は提案手法による  $\lambda = 0.1$  のときの実験結果を元に行う。

Table 3: Youden's index による ROC 曲線上の最適なカットオフ地点を閾値に設定した際の異常検知結果

Model	$\lambda$	Precision	Recall	Specificity	F1	AUC
AnoGAN	0	0.709	0.578	0.766	0.637	0.721
	0.1	0.707	0.605	0.753	0.652	0.738
	0.5	0.617	<b>0.763</b>	0.532	0.682	0.692
	1	0.633	0.750	0.571	0.686	0.676
Our model	0	<b>0.759</b>	0.539	<b>0.831</b>	0.630	0.728
	0.1	0.746	0.697	0.766	<b>0.721</b>	<b>0.771</b>
	0.5	0.718	0.671	0.740	0.693	0.738
	1	0.651	<b>0.763</b>	0.597	0.703	0.724



(a) AnoGAN による実験結果 (b) 提案手法による実験結果  
Fig. 7: 異常度に基づく ROC 曲線

検証用データに加えて、提案手法により学習データの異常度の算出を行った。学習データ、検証用の正常データ、異常データの異常度の分布を Fig. 8 に示す。Fig. 8 から正常データと異常データの異常度には相違があると考えられる。正常データと異常データの異常度に対して Brunner-Munzel 検定<sup>16)</sup>を行った。Brunner-Munzel 検定は両群から標本を1つずつ抽出したときに、2つの確率変数のどちらが大きい確率も等しいこ

との帰無仮説を検定する。また、Brunner-Munzel 検定は等分散性の仮定が不要なノンパラメトリックな検定方式である。有意水準 0.01 で Brunner-Munzel 検定を行ったところ、正常データと異常データの異常度が一致することが棄却されたため、これらの異常度には有意な差があることが示された。

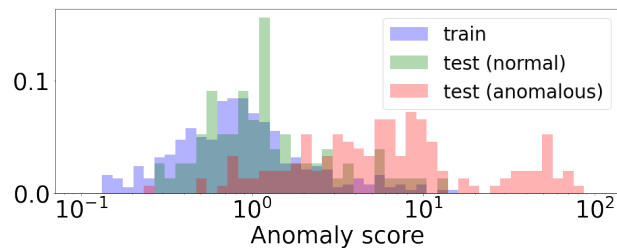
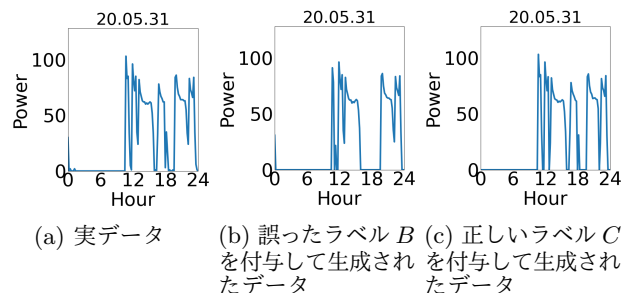


Fig. 8: 学習データ、検証用の正常データ、異常データに提案手法を適用したときの異常度の分布

擬似的な異常データの中で、提案手法による異常度が比較的高く算出された 2020 年 05 月 31 日のデータに着目する。Fig. 9 に 2020 年 05 月 31 日の実データと提案手法により生成されたデータを示す。Fig. 9a と Fig. 9b を比較すると分かるように、誤ったラベルを付与して生成されたデータは実データと比較して一部欠けた波形となっている。一方、正しいラベルを付与して生成された場合は、Fig. 9c のように実データに非常に類似したデータを生成することに成功していることがわかる。



(a) 実データ (b) 誤ったラベル B を付与して生成されたデータ (c) 正しいラベル C を付与して生成されたデータ  
Fig. 9: 2020年05月31日の実データと提案手法により生成されたデータの比較

擬似的な異常データの中で、提案手法による異常度が比較的低く算出された 2020 年 06 月 13 日のデータに着目する。Fig. 10 に 2020 年 06 月 13 日の実データと提案手法により生成されたデータを示す。Fig. 10a と Fig. 10b を比較すると分かるように、誤ったラベルを付与したにも関わらず、実データに非常に類似したデータを生成するにしている。その、異常度が低く算出されたと考えられる。また、13日のデータを生成する際に付与した誤ったラベルである A に属する他の日の実データを Fig. 11 に示す。13日の実データ Fig. 10a によく似たデータが Fig. 11 に見受けられることがわかる。そのため、モデルが Fig. 10b のような、実データに類似したデータを生成することができたと考えられる。このような理由から、13日のデータの異常度が低く算出されたと予測される。また、同様の現象が他の日のデータにも起きていることが示唆されるため、実運用上は specificity が高くなることで、Table 3 で示した実験結果よりも高い精度で異常を検知することが期待できる。

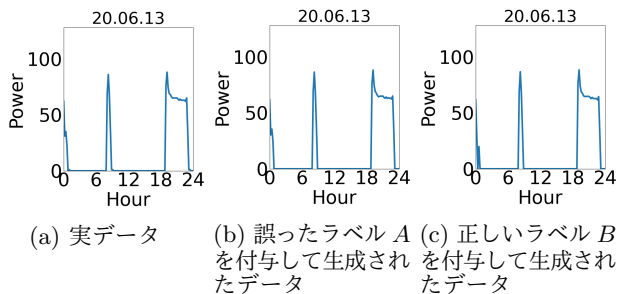


Fig. 10: 2020年06月13日の実データと提案手法により生成されたデータの比較

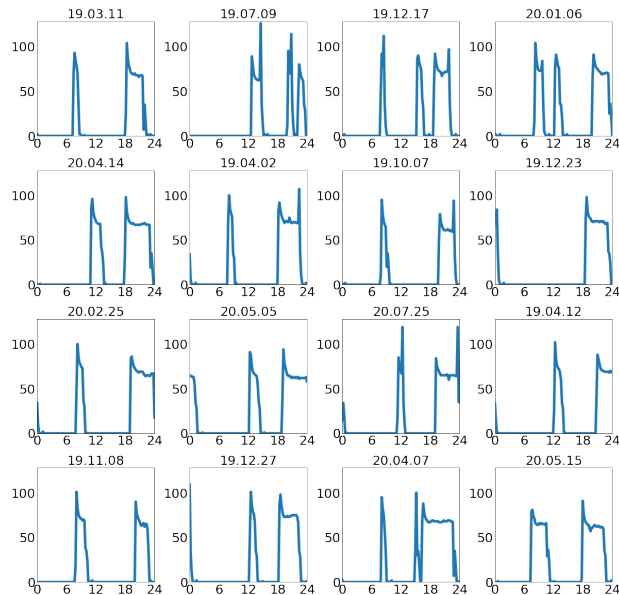


Fig. 11: 提案手法により2020年06月13日のデータを生成するときに付与した誤ったラベルAに属する他の日のデータの例

## 4 結論

本研究では、消費電力に基づく見守りシステムのための異常検知手法を提案して、実際に高齢者の自宅のテレビから得られた消費電力データを用いて実験を行った。学習済みの Generator により生成を行った結果、モード崩壊を起こすことなく学習データに類似した波形を生成することに成功したことから、モデルが適切に学習していることが示された。

次に、Discriminator に入力して得られる中間層の出力のヒートマップを作成して可視化することで実データと比較を行った。また、実データと提案手法により生成されたデータ間の距離と、それぞれのデータを Discriminator に入力して得られる中間層の出力の距離で比較を行った。結果として、中間層の出力が波形の形を捉えることができていることが確認できた。また、中間層の出力を用いることで波形同士の距離をより正確に計測できることが示された。

また、異常検知の検証を行うため実データを用いて異常検知の実験、及び結果の分析を行った。AnoGAN と提案手法による異常検知結果を比較した結果、提案手法が高い精度で異常を検知できることが示された。また、Residual Loss の割合が高くなる程 Precision, Specificity が高くなり、Discrimination Loss の割合が高く

なる程 Recall が高くなる傾向が見られた。さらに、これらの Loss を両方用いることで、より高い精度で異常を検知できることが示された。また、正常データと異常データの異常度に対して Brunner-Munzel 検定を行ったところ、これらの異常度には有意な差があることが示された。さらに、異常データにも関わらず異常度が低く算出された日のデータに着目して分析した結果、実運用上は実験結果よりも、高い精度が期待できることが示された。

## 参考文献

- 1) Satoru Izumi, Kazuhiro Yamanaka, Yoshikazu Tokairin, Hideyuki Takahashi, Takuo Suganuma, and Norio Shiratori. Ubiquitous supervisory system based on social contexts using ontology. *Mobile Information Systems*, 5(2):141–163, 2009.
- 2) 岡田 遼太郎 and 矢入 郁子. 赤外線グリッドセンサを用いた室内位置・行動推定の粒度及び精度の検討. 人工知能学会全国大会論文集, JSAI2014:2J14–2J14, 2014.
- 3) Wensi Tang, Guodong Long, Lu Liu, Tianyi Zhou, Jing Jiang, and Michael Blumenstein. Rethinking 1d-cnn for time series classification: A stronger baseline. *CoRR*, abs/2002.10061, 2020.
- 4) Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.
- 5) Thomas Schlegl, Philipp Seeböck, Sebastian M. Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. *CoRR*, abs/1703.05921, 2017.
- 6) J. Shlens A. Odena, C. Olah. Conditional Image Synthesis With Auxiliary Classifier GANs. *CVPR*, 2016.
- 7) Douglas A. Reynolds, Thomas F. Quatieri, and Robert B. Dunn. Speaker verification using adapted gaussian mixture models. *Digit. Signal Process.*, 10(1-3):19–41, 2000.
- 8) Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. *J. Big Data*, 6:60, 2019.
- 9) Christopher M. Bishop. *Pattern Recognition and Machine Learning*. Springer, New York, 2006.
- 10) Diganta Misra. Mish: A self regularized non-monotonic neural activation function. *CoRR*, abs/1908.08681, 2019.
- 11) Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MITP, 2018.
- 12) Jun Han and Claudio Moraga. The influence of the sigmoid function parameters on the speed of back-propagation learning. volume 930 of *Lecture Notes in Computer Science*, pages 195–201, 1995.
- 13) Bergstra, J. Desjardins, G. Lamblin, P. Bengio, Y. Quadratic polynomials learn better image features Technical Report 1337, 2009.
- 14) Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2018. cite arxiv:1802.03426Comment: Reference implementation available at <http://github.com/lmcinnes/umap>.
- 15) W. J. Youden. Index for rating diagnostic tests. *Cancer*, 3(1):32–35, 1950.
- 16) Brunner E. and Munzel U. The nonparametric behrens-fisher problem: Asymptotic theory and a small-sample approximation, *biometrical journal*, vol. 42, 17-25. 2000.