

ニューラルネットワークによる騒音予測

平方 勝*, 馬 沖**, 小沢 匠*, 谷口 智之*

Ship Noise Prediction by Using Neural Network

by

Masaru HIRAKATA, Chong MA, Takumi OZAWA and Tomoyuki TANIGUCHI

Abstract

At IMO MSC 91, Code on noise levels on board ships was adopted. Noise measurement is required by this code. Designs and constructions were to be imposed on the shipyards so that the required areas and spaces were below the required noise levels. It was found that the code is very strict for small-sized vessels.

Noise prediction calculation is necessary to make a design that conforms to the noise code. In particular, considering the arrangement of anti-noise products, it is necessary to roughly estimate noise levels at the basic design stage. NMRI has created a noise prediction program by Janssen method as one of the noise predictions. The noise prediction program based on the Janssen method by using the actual noise measurement results of the same type of ship to make predictions. The Janssen method is an empirical method based on actual ship noise levels. If there is no same type ship, it is predicted based on the noise measurement results of similar ships with similar characteristics as much as possible. In selecting similar ships, empirical knowledge is required.

Machine Learning by neural networks can complement and infer missing information as seen in brain functions. Such inference can be realized by repeating many pattern learnings. In this paper, the noise prediction method using neural network is developed with reference to the noise prediction method by Janssen method, which is an empirical method, and the contents are systematized and introduced.

* 構造基盤技術系, ** 構造安全評価系

原稿受付 2019年4月23日

審査日 2019年6月21日

目 次

1. まえがき	47
2. ニューラルネットワークについて	48
2.1 全般	48
2.2 順伝播 (予測)	48
2.3 逆伝播 (学習)	49
2.4 ミニバッチ	50
2.5 ドロップアウト	50
2.6 学習係数	51
2.7 汎化性能と過学習	51
2.8 SOM (自己組織化マップ: Self-Organizing Map) ⁴⁾	52
3. ニューラルネットワークによる騒音予測手法 ⁵⁾	53
3.1 全般	53
3.2 入力データ	54
3.3 ニューラルネットワークの構造	54
3.4 SOM との併用	54
4. ニューラルネットワークによる騒音予測の検証	55
4.1 類似船騒音計測データを教師データとした騒音学習と予測	55
4.2 全データを教師データとした騒音学習と予測	57
4.3 誤差の検証	58
4.4 ニューラルネットワークの特徴と今後について ⁶⁾	58
5. まとめ	60
参考文献	60

記 号

u	: ユニットに入力される総入力
w_i	: シナプス重み
x_i	: シナプスへの入力信号
b	: バイアス
z	: 総入力に対する活性化関数 f の出力
f	: 活性化関数
$E(w)$	: 誤差関数 $E_n(w)$ の和
$E_n(w)$	: 誤差関数
ε	: 学習係数
D_t	: ミニバッチ
$E_t(w)$	: D_t の含む全サンプルに対する誤差
N_t	: ミニバッチが含む全サンプル数
$g_{t,i}$	: 誤差関数の勾配のベクトル成分
m_t	: Adam パラメータ
α	: Adam パラメータ
β_1	: Adam パラメータ
β_2	: Adam パラメータ
m_t	: 勾配の一次モーメント

v_t : 勾配の二次モーメント
 $\hat{m}$: 移動指数平均を用いた際に生じるバイアス
 $\hat{v}$: 移動指数平均を用いた際に生じるバイアス
 $m_i(t)$: 結合荷重
 $h_{ci}(t)$: 学習率係数 $a(t)$ を含んだ近傍関数
 $x(t)$: 入力ベクトル
 $a(t)$: 学習率係数
 N_c : 近傍領域
 a_0 : a の初期値
 t : 学習回数
 T : 設定した学習回数

1. まえがき

2012年11月、国際海事機関 (IMO) 第91回海上安全委員会 (MSC91)において、勧告案であった船内騒音コード (Code on noise levels on board ships)が改正され、改正された船内騒音コード（以下、騒音コードという）を総トン数1,600トン以上の新造船にその適用を義務化する SOLAS 条約の改正案が採択された¹⁾。本改正は、2014年7月1日に発効した。騒音コードが義務化されることにより、海上公試時及び港内時の騒音計測が義務付けられた。要求される区域・区画等では、騒音規制値を下回るように、設計・建造を行うことが造船所に課せられることになった。

騒音コードは、騒音に起因する船員の健康被害、操船時における指示伝達の阻害による安全性への影響を最小限にすることを目的としている。近年、国内外において船員の労働環境の改善に対する要請が高まっており、そのような国際的な流れの中で実現した規則である。一方、中小型船は、従来通りの設計では、騒音コードに適合させることが困難であることが実船調査で明らかになった。造船所は騒音予測・対策を徹底することが求められることになった。

騒音コードに適合した設計を行うためには、騒音予測計算が必要になる。特に、騒音対策品の手配を考えると、基本設計段階でおおよその騒音予測を行う必要がある。騒音予測のひとつとして、Janssen 法による騒音予測手法がある²⁾。Janssen 法による騒音予測手法は、船体・機関要目、内装情報等と音響放射効率、平均吸音率等、騒音予測に必要な音響パラメータを用いて、同型船の実船計測結果を利用して予測を行う手法である。すなわち、実績に基づく経験的な手法である。同型船がない場合は、可能な限り要目の近い類似船の騒音計測結果を基に予測する。類似船の選定にあたっては、経験的知識が必要となる。

人工知能の機械学習のひとつに、脳機能を模擬したニューラルネットワークによる機械学習手法がある。ニューラルネットワークによる推論（計算）では、欠損情報を補完し推論することができる。近年、畳み込みニューラルネットワーク (CNN : Convolutional Neural Network) をベースにした深層学習による画像認識が実用化されている。ニューラルネットワークによる機械学習は、データを蓄積していけばいくほど、予測精度が改善されることが期待できる。ニューラルネットワークによる予測は、経験的に処理するような回帰問題や分類問題に利用できる。

本研究では、経験的手法である Janssen 法による騒音予測手法を参考に、ニューラルネットワークによる騒音予測へと発展させた。本稿では、ニューラルネットワークによる騒音予測手法の紹介を行い、ニューラルネットワークによる騒音予測精度を検証する。また、ニューラルネットワークによる騒音予測精度を上げるために、教師データ（騒音計測データ）を蓄積していく必要がある。そこで、騒音計測データを系統立てて取得していくために、Web アプリケーションシステムを製作した。本稿では、Web アプリケーションシステム (AI-AcousticNet) について紹介する。

2. ニューラルネットワークについて

2.1 全般

1956 年のダートマス会議で、「人工知能」という用語が初めて使用された。人工知能は、推論、認識、判断など、人間と同じ知的な処理能力をもつコンピュータシステムである。しかしながら、明確な定義は未だにない。現在、第三次人工知能ブームといわれている。ダートマス会議にはじまる第一次ブームから、記号を用いた論理的推論を行う記号主義の分野と、ローゼンブラットが提唱したパーセプトロンによる人間の神経回路を模擬したニューラルネットワークの流れの研究と、二つの流れの研究ができた。第3次人工知能は、ニューラルネットワークをベースにした画像処理、文章生成、深層強化学習が牽引している。

ニューラルネットワークは、ローゼンブラットが提唱したパーセプトロンモデルが発祥であるが、その他にも相互結合型ニューラルネットワークや自己組織化マップ(SOM: Self Organizing Map)など、いくつかのタイプが存在する。多層パーセプトロンタイプのニューラルネットワークも、画像処理などを取り扱う場合は、畳み込みニューラルネットワーク (CNN) が使用される。また、時系列データ (音声・言語データ) を取り扱う場合は、再帰型ニューラルネットワーク (RNN: Recurrent Neural Network) が使用される。いずれも、最近では、深層タイプのニューラルネットワーク、いわゆるディープラーニングが使用されることが多い。用途に応じて、使用するニューラルネットワークを使い分ける。

多層パーセプトロンタイプのニューラルネットワークは、予測を行うような回帰問題や判別を行うような分類問題に使用されることが多い。このニューラルネットワーク (図 2.1) は、順伝播 (入力ユニットから出力ユニット) に信号が流れるときは予測・推論を行い、学習を行う時は、逆伝播 (出力ユニットから入力ユニット) に信号が流れ、教師データと予測結果の誤差が小さくなるように計算が進行する。

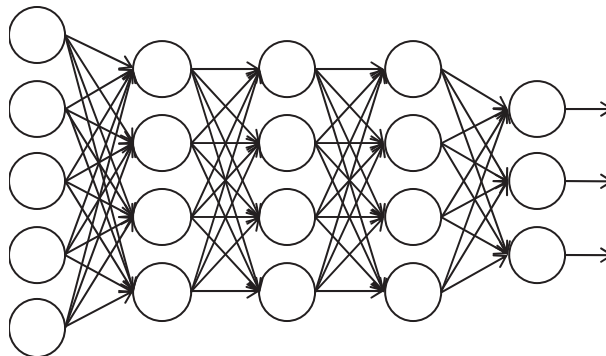


図 2.1 多層パーセプトロンタイプのニューラルネットワーク

2.2 順伝播 (予測)

順伝播は、図 2.1 に示すように、層状に並んだユニットが隣接する層間でのみ結合した構造で、入力層から出力層への伝播である。図 2.2 に示すように、完全結合 (入力側の隣層の全てのユニットから信号が流れる) にし、式(2.1)のように、各入力にそれぞれ異なるシナプス重みを掛け合わせて、バイアスを足し合わせた式で、ユニットに入力される総入力を与えられる。

$$u = w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + b \quad (2.1)$$

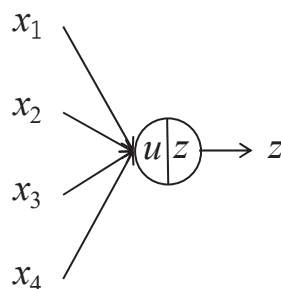


図 2.2 ユニットへの入出力

この総入力に対して、活性化関数と呼ばれる関数にしたがって、当該ユニットから出力される。出力 z は、総入力 u に対する活性化関数と呼ばれる関数 f の出力である。

$$z = f(u) \quad (2.2)$$

この仕組みは、人間の脳神経細胞における信号伝達を模擬している。第二次人工知能ブームでは、活性化関数は、人の脳神経機構を模擬したシグモイド関数が多く利用されていた。しかし、最近のディープラーニングでは、脳神経機構とはかけはなれた ReLU (Rectified Linear Unit) 関数が多く使用される。出力層の活性化関数は、回帰（予測）問題では、シグモイド関数や恒等写像を利用することが多く、分類問題では、ソフトマックス関数を使用するなど、問題設定に応じて使い分ける。回帰（予測）問題とは、主に出力に連続値をとる関数を対象に、訓練データを再現するように学習を行う問題である。一方、分類問題では、入力データに対して、有限個のクラスに分類する問題である。

2.3 逆伝播（学習）

ニューラルネットワークによる機械学習は、教師データを与えて、出力層から入力層に誤差を小さくするように伝播しながら、シナプス重みを修正する学習方法であり、誤差逆伝播学習法と呼ぶ。誤差逆伝播学習法が紹介されたことにより、第一次人工知能ブームで課題となった学習問題が解決されることになった。第二次人工知能ブームでは、勾配降下法による学習を行っていた。勾配降下法は、全訓練サンプルに対して計算される誤差関数 $E(w)$ を最小化する。 $E(w)$ は各サンプル 1 個だけについて計算される誤差 E_n の和として式(2.3)のように与えられる。

$$E(w) = \sum_{n=1}^N E_n(w) \quad (2.3)$$

シナプス重みは、式(2.4)に従って、 $E(w)$ の勾配を用いて更新される。

$$w^{(t+1)} = w^{(t)} - \varepsilon \nabla E \quad (2.4)$$

この方法をバッチ学習と呼ぶ。勾配降下法は、比較的に計算負荷が高い。

これに対してサンプルの一部、またはサンプル一つだけを使用してパラメータの更新を行う方法がある。これを確率的勾配降下法(Stochastic Gradient Descent: SGD)という。この方法では、 w の更新は一つのサンプル n について計算される誤差関数 $E_n(w)$ の勾配 ∇E_n を計算し、式(2.5)のように w を更新する。

$$w^{(t+1)} = w^{(t)} - \varepsilon \nabla E_n \quad (2.5)$$

次の $w^{(t+1)}$ の更新の際は、別のサンプル n' を取り出し、これを使って同様に勾配 $\nabla E_{n'}$ を評価し、 w を更新する。このようにサンプルを毎回取り換え、 w を更新する。確率的勾配降下法は、訓練データに冗長性がある場合、計算効率が向上し学習が速く行える特徴がある。また、局所的な極小解にトラップされてしまうリスクを低減することができる。

2.4 ミニバッチ

計算効率を上げるために、シナプス重みの更新をサンプル一つ単位で行うのではなく、少数のサンプルの集合をひとまとめにし、その単位でシナプス重みを更新する手法がある。このひとまとめにしたサンプル集合をミニバッチという。

一つ目のミニバッチを D_t とする。これは、少数のサンプルの集合を表す。添字の t は、 t 回目の更新ごとにそのサンプル集合が変わることを表す。 D_t の含む全サンプルに対する誤差を式(2.6)に従って計算し、この勾配の方向にパラメータを更新する。

$$E_t(w) = \frac{1}{N_t} \sum_{n \in D_t} E_n(w) \quad (2.6)$$

なお、 $N_t = |D_t|$ はこのミニバッチが含むサンプル数である。ミニバッチのサイズを決める系統的なやり方はないが、10~100 サンプル前後とすることが多い。ミニバッチのサイズをあまり大きくすると、確率的勾配降下法の良さを損なうことになり、あまり良くない。

2.5 ドロップアウト

ドロップアウトは、多層ニューラルネットワークのユニットを確率的に選別して学習する方法である。この技術は第三次人工知能ブームになって出てきた技術である。ドロップアウトは、過学習（後述）を起こさないようにとる手段の一つである。

ドロップアウトは、学習時は、隠れ層の各層と入力層のユニットを決まった割合 p でランダムに選出し、それ以外を無効化、つまり存在しないかのように扱う手法である。訓練データをネットワークに入力して順伝播計算を行い、誤差を求めた後、逆伝播計算を行って誤差の勾配を計算し、シナプス重みを更新する。なお、ユニットの選出はシナプス重み更新のたびにランダムに行う。ミニバッチを使っている場合には、ミニバッチ単位でユニットを選びなおすことになる。ドロップアウトの計算にあたっては、各層のドロップアウトする確率 p を与えることになる。図 2.3 は、ドロップアウトしたユニットをグレーで示す。

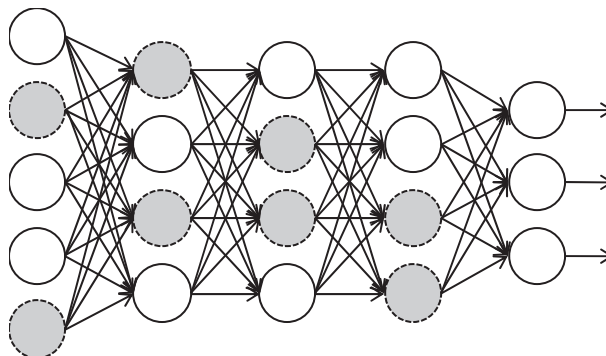


図 2.3 ドロップアウト

2.6 学習係数

勾配降下法では、シナプス重みの更新量の大きさは、学習係数によって変わる。この学習係数をどう決めるかが学習の成否を左右するため重要である。学習係数を手動で試行錯誤的に決定することもあるが、最近では、自動的に決定することが多い。

学習係数を自動的に決定する（変化させる）方法はいくつかあるが、最近ではAdaGradと呼ばれる手法がある。これは、誤差関数の勾配を $g_t \equiv \nabla E_t$ と書き、このベクトルの成分を $g_{t,i}$ と書くと、普通の勾配降下法では更新量の i 成分は $-\epsilon g_{t,i}$ であるが、AdaGradでは、これを式(2.7)とする。

$$-\frac{\epsilon}{\sqrt{\sum_{t'=1}^t g_{t',i}^2}} g_{t,i} \quad (2.7)$$

頻出する勾配の成分よりもまれに現れる成分をより重視しつつ、パラメータを更新することになる。

Adam (Adaptive moment estimation)という手法は、一次モーメントと二次モーメントの偏りをバイアス補正した推定値を計算することによって、偏りを小さくし精度を高めているところに特徴がある。これは、以下の式(2.8)のようにシナプス重みを更新する。パラメータは $\alpha = 0.001, \beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 10^{-8}$ が論文³⁾内で推奨されている。

$$\begin{aligned} m_{t+1} &= \beta_1 m_t + (1 - \beta_1) \nabla E(w^t) \\ v_{t+1} &= \beta_2 v_t + (1 - \beta_2) \nabla E(w^t)^2 \\ \hat{m} &= \frac{m_{t+1}}{1 - \beta_1^t} \\ \hat{v} &= \frac{v_{t+1}}{1 - \beta_2^t} \\ w^{t+1} &= w^t + \alpha \frac{\hat{m}}{\sqrt{\hat{v} + \epsilon}} \end{aligned} \quad (2.8)$$

初期値は $m_0 = 0, v_0 = 0$ である。Adamでは、移動指数平均を用いた際に生じるバイアスをそれぞれ $\hat{m}$ 、 $\hat{v}$ で打ち消すことになるところに特徴がある。

2.4～2.6で述べた学習技術が出てきたことにより、ニューラルネットワークによる学習の多くの課題が解決されるようになった。

2.7 汎化性能と過学習

ニューラルネットワークの性能を評価する上で、常に、汎化性能を確認する必要がある。通常、ニューラルネットワークの学習に使用される訓練データに対して、学習は進み（誤差が小さくなり）、訓練データに対する予測精度は上がるが、訓練データ以外のデータに対しても、同様に予測精度を確保する必要がある。そこで、訓練を行わない検証データに対しても、予測精度を確認するための検討を同時に行う必要がある。

学習によるパラメータの更新に伴って、訓練誤差及びテスト誤差がどのように変化するかを、図2.4に示す。図2.4は学習曲線と呼ぶ。訓練誤差は一般的に学習が進むにつれ収束に向かう。図2.4左図のように、検証データも同時に収束に向かう場合、安定して学習が行われているといえるが、検証データによる誤差が図2.4右のように、発散する場合、学習が安定していないと判断できる。このような状態を過学習という。ニューラルネットワークの学習時には、常に、学習曲線の様子に注意し、過学習がおこらないように対応しなければならない。

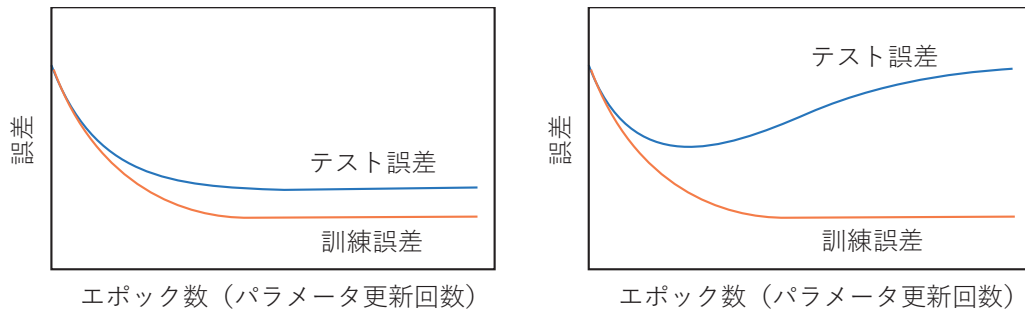


図 2.4 学習曲線 (左図：安定した学習，右図：不安定な学習 (過学習))

2.8 SOM (自己組織化マップ: Self-Organizing Map) ⁴⁾

SOM (自己組織化マップ) は、コホネンが脳皮質の視覚野をモデル化したニューラルネットワークの一つである。似たものをまとめて分類するアルゴリズムである。SOM は、図 2.5 に示すように、入力層と出力層からなる 2 層構造である。教師データを必要としない、いわゆる教師なし学習のニューラルネットワークである。入力 は n 次元の数値データであり、分類したいものの属性値からなる。分類したいものは、出力結果として出力層にマッピングされる。出力層のユニットは、入力データの次元と同じ次元のベクトル (参照ベクトルという) を有している。入力ベクトルと出力層、及び参照ベクトルの関係は、図 2.6 に示すような関係にある。

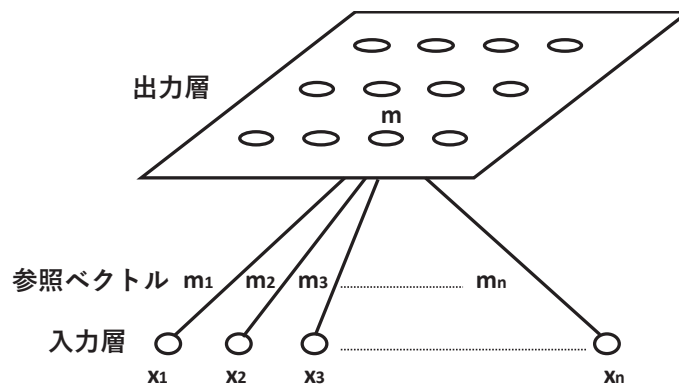


図 2.5 SOM のモデル

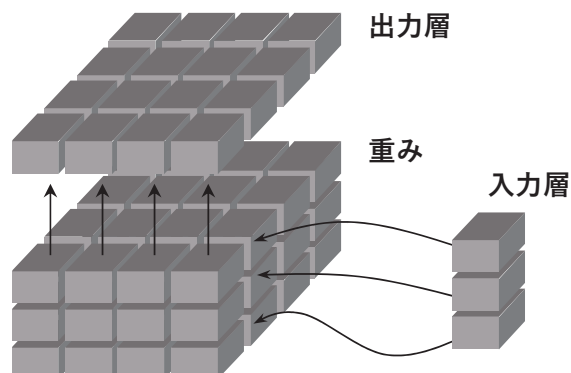


図 2.6 入力ベクトルと重みの関係

SOM のアルゴリズムは、2 段階の計算が繰り返される (図 2.7 参照)。

第一段階：入力ベクトルと最もよく一致する出力層での勝者ユニット（参照ベクトル）を探す．入力ベクトルと最もよく一致する出力層を検索するには，ここではユークリッド距離 $|x - m|$ が最小になるユニットを探す．

第二段階：この勝者ユニットの近傍の内側ユニットを式(2.9)に従って更新（学習）する．近傍の外側ユニットは，更新（学習）しない（式(2.10)参照）．

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)] \quad (2.9)$$

ここで， $h_{ci}(t)$ は，学習率係数 $a(t)$ を含んだ近傍関数という．

$$\begin{aligned} h_{ci} &= a(t) & i \in N_c \\ h_{ci} &= 0 & i \notin N_c \end{aligned} \quad (2.10)$$

参照ベクトルは式(2.9)に従って，入力ベクトルに応じて更新される．すなわち，学習の進行に伴って参照ベクトルが更新される．近傍領域の外側では学習は行われない．学習が進むにつれ，近傍のサイズを減らし，また学習率係数の値も式(2.11)に従って減らしていく．

$$a(t) = a_0(1 - t/T) \quad (2.11)$$

ここで， a_0 は a の初期値であり，ここでは 0.1 とした． T は設定した学習回数での上限値， t は現在の学習回数である．

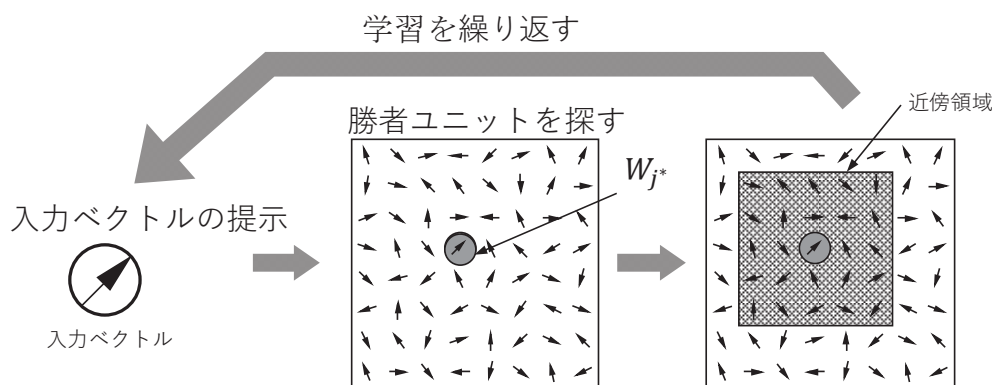


図 2.7 SOM の学習モデル

3. ニューラルネットワークによる騒音予測手法⁵⁾

3.1 全般

ニューラルネットワークによる騒音予測プログラムを作成し，騒音レベルの学習及び予測を行った．プログラム作成はプログラム言語 Python を使用して行った．ニューラルネットワークの学習計算は微分計算が多い．そのため，Function が充実している数値計算ライブラリ Theano を利用した．

学習計算にあたる逆伝播計算では，データの正規化，誤差逆伝播学習法，確率的勾配降下法，ミニバッチ，正則化，ドロップアウト処理が行えるようになっている．ただし今般計算ではドロップアウトは行っていない．また，予測にあたる順伝播計算は，データの正規化を行っている．正規化を行うことによって，特に，逆伝播計算（学習）の計算速度が大きく改善する．

3.2 入力データ

ニューラルネットワークの学習・予測計算に使用した船舶は、実船計測を行った 55 隻である。表 3.1 に、対象とした船舶とその大きさ（総トン数 GT）を示す。

表 3.1 対象船舶

船番	GT	船番	GT	船番	GT	船番	GT	船番	GT
S01	約 3,600	S12	約 16,100	S23	約 9,700	S34	約 600	S45	約 21,100
S02	約 3,800	S13	約 3,400	S24	約 8,700	S35	約 9,900	S46	約 13,100
S03	約 4,000	S14	約 3,600	S25	約 8,700	S36	約 9,900	S47	約 13,100
S04	約 4,200	S15	約 8,000	S26	約 10,000	S37	約 3,500	S48	約 13,100
S05	約 13,000	S16	約 8,000	S27	約 7,600	S38	約 3600	S49	約 13,100
S06	約 5,700	S17	約 10,000	S28	約 9,300	S39	約 11,500	S50	約 23,400
S07	約 5,500	S18	約 44,000	S29	約 8,700	S40	約 9,800	S51	約 14,800
S08	約 5,400	S19	約 4,300	S30	約 10,000	S41	約 3,000	S52	約 14,800
S09	約 12,200	S20	約 8,000	S31	約 7,600	S42	約 15600	S53	約 10,000
S10	約 12,200	S21	約 23,300	S32	約 9,300	S43	約 2,900	S54	約 8,000
S11	約 16,100	S22	約 23,200	S33	約 10,800	S44	約 21,100	S55	約 8,500

3.3 ニューラルネットワークの構造

騒音レベルの学習及び予測は、図 3.1 に示す多層パーセプトロンタイプのニューラルネットワークを使用した。入力層は、Janssen 法による騒音予測で入力データとして使用している音響パラメータや、騒音計測・音源探索の実績から騒音レベルに影響を与えそうな項目を、入力ユニットとして与える。隠れ層を深層にすることも考えられるが、隠れ層を深層にするにつれて、学習時間が増加すること、学習精度が悪化することもあるため、隠れ層は 1 層とした。隠れ層及び出力層での活性化関数は、回帰（予測）問題で一般的に使用されるシグモイド関数を使用した。計算効率を上げるために、バッチサイズを 10 として、ミニバッチによる学習を行った。学習計算を速くするために、データの正規化を行った。過学習を防ぐためドロップアウトを採用することもあるが、学習を行ったところ過学習になるような傾向がみられなかったため、今回は、ドロップアウトを行っていない。学習係数の決定には、Adam を採用した。

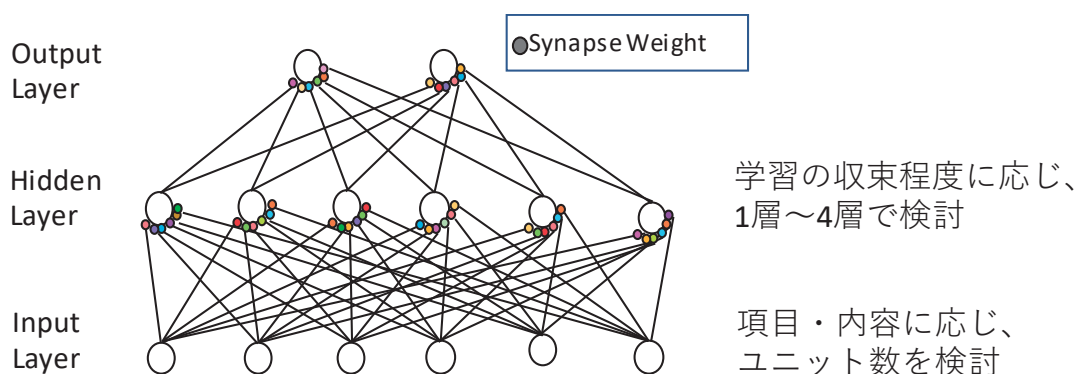


図 3.1 ニューラルネットワークの構造

3.4 SOM との併用

ニューラルネットワークの学習は、教師データが多いほど精度が上がるといわれている。学習は部屋ごとに行う。今計算では、学習した部屋数は 1656 部屋に相当し、部屋の数では十分な教師データになっていると思わ

れる．さらに，船のサイズや構造など，今後多様性を増していく必要がある．図 3.2 は，1 隻の船舶をひとつのピースにたとえている．類似船は近くに表示されているとする．

たとえば，(a)のような時点では“船 26”の類似船は，“船 13”か“船 42”になるとおもわれる．教師データが充実してきて，たとえば，(b)のような時点では“船 26”の類似船は，“船 33”，“船 25”，“船 22”など，(a)の時点より類似性が大きくなり，類似船の隻数も増してくる．(c)の場合は，さらに教師データ（船舶の数）が充実してくる．すなわち学習により騒音予測の精度が向上する．

本ニューラルネットワークによる騒音予測も，Janssen 法による騒音予測と同様で，同型船または類似船の騒音計測結果を基に予測する経験的な手法である．Janssen 法による騒音予測は，同型船または類似船 1 隻を対象にチューニングを実施しており，その影響を強く受けることがある．ニューラルネットワークによる騒音予測は教師データ（騒音計測を行った船舶の隻数）を充実させれば，特異なノイズに影響にされことなく騒音メカニズムをとらえた予測が行えると思われる．そして，それら計測データが今後の船舶の設計に有効に蓄積・活用でき，新設計の船舶の予測等，騒音予測の精度が向上できることがニューラルネットワークによる騒音予測を行う上でポイントになる．

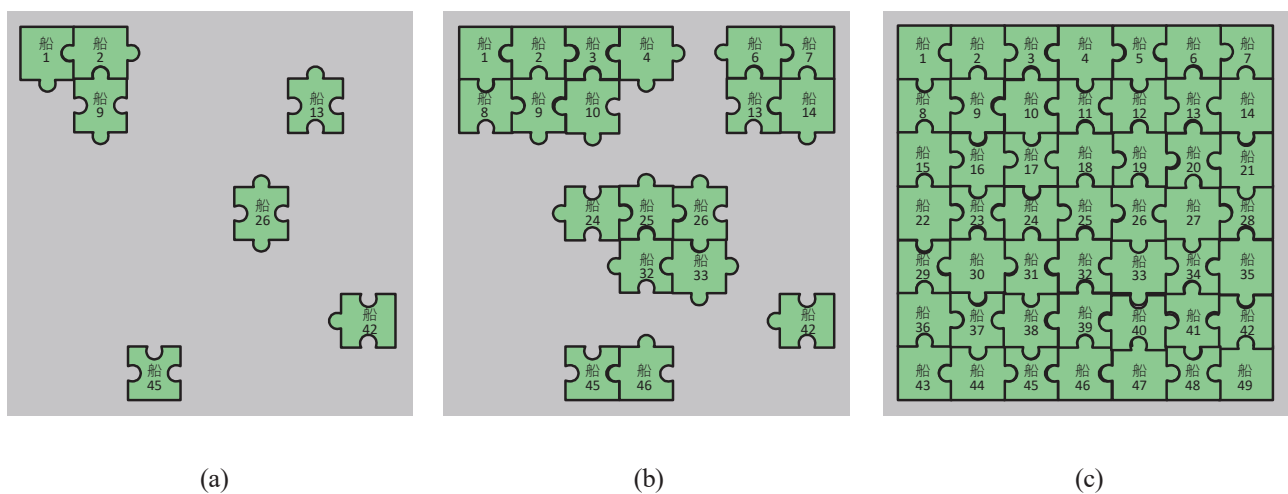


図 3.2 教師データの充実（イメージ）

船舶の隻数・多様性が充実してくるまでの間は，SOM を利用して類似船（又はその集団）を見つけてきて，その類似船（又はその集団）の学習結果をもとに，騒音予測を行うことが運用上望ましいと考えている．

4. ニューラルネットワークによる騒音予測の検証

4.1 類似船騒音計測データを教師データとした騒音学習と予測

同型船の判別は一般的に，船の長さ，船の型幅，船の型深さが一つの指標となるが，ここでは，騒音レベルの特徴を表現するために，これらの情報に加えて，航路（外航か内航か），総トン数，機関室の長さ，定員，機関室内のフレームスペース，主機・発電機の位置，主機の種類等さらに多くの情報を入力データとした．

ここでは，36 隻の船舶を対象に，SOM による船舶の分類を行った．図 4.1 に，船舶を色ごとに分類した結果を示す．図 4.1 内太枠で囲んだ船舶 A と船舶 B は同一造船所で建造された船の長さ(L)，型幅(B)，型深さ(D)の値が近い類似船である．図 4.1 に示すように，類似船が隣り合うように SOM でマッピングされていることが分かる．なお，図 4.1 では，パラメータが近い船舶が多い場合，一隻当たりの面積が小さく表示され，パラメータが近い船舶が少ない場合，一隻当たりの面積が大きく表示される．

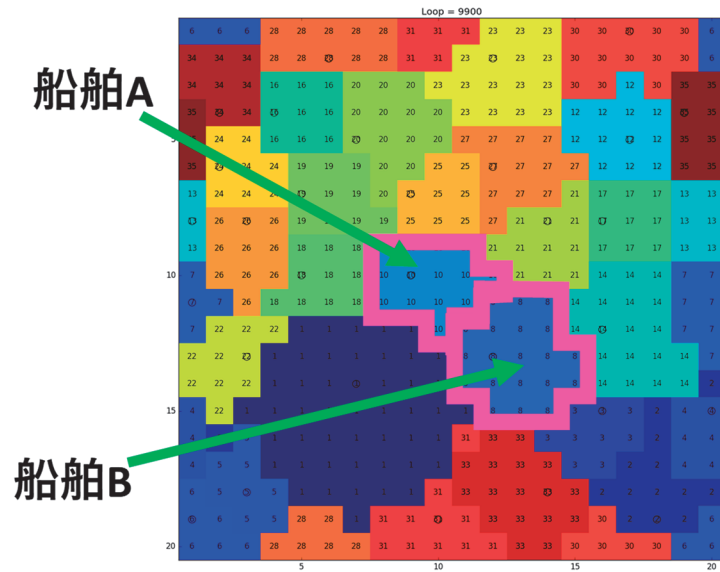


図 4.1 SOM マッピング結果

図 4.1 の SOM 結果に示した類似船（船舶 A）の部屋を訓練データと検証データに分けて学習を行った。ミニバッチによる学習を行い、確率的勾配降下法は使用していない。学習の様子を図 4.2 に示す。訓練データも検証データも学習がすすむにつれ、収束していく様子が分かる。すなわち、過学習は起きていないことが分かる。

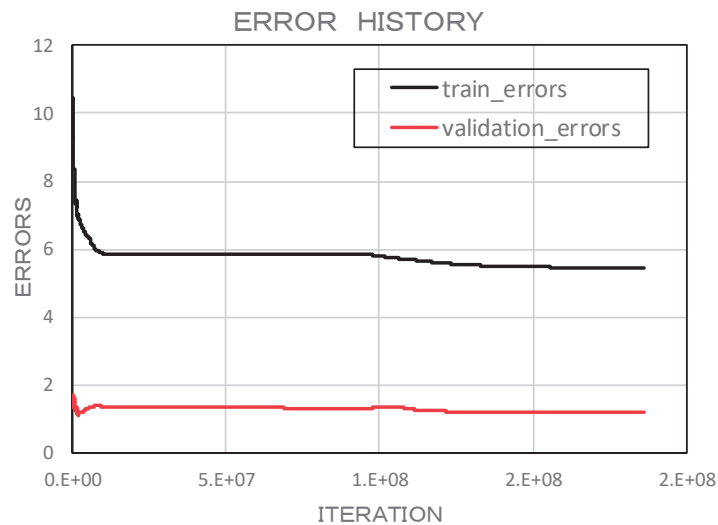


図 4.2 学習の様子

類似船（船舶 A）の学習結果（シナプス重み）を用いて、計画船（船舶 B）の騒音予測を行った。計画船（船舶 B）の騒音予測結果と計測結果の誤差（予測誤差）と、類似船（船舶 A）の計測誤差を比較した（図 4.3 参照）。なお、計測誤差は、船舶 A と同型船の別の船舶とで、（同じ部屋で）騒音計測を行った際の、騒音計測結果の差にあたる。図 4.3 より、ニューラルネットワークによる予測誤差は 1 部屋のみ 6～7dB(A)と大きいですが、計測誤差と同じ程度の $\pm 4\text{dB(A)}$ 以内で評価できている。

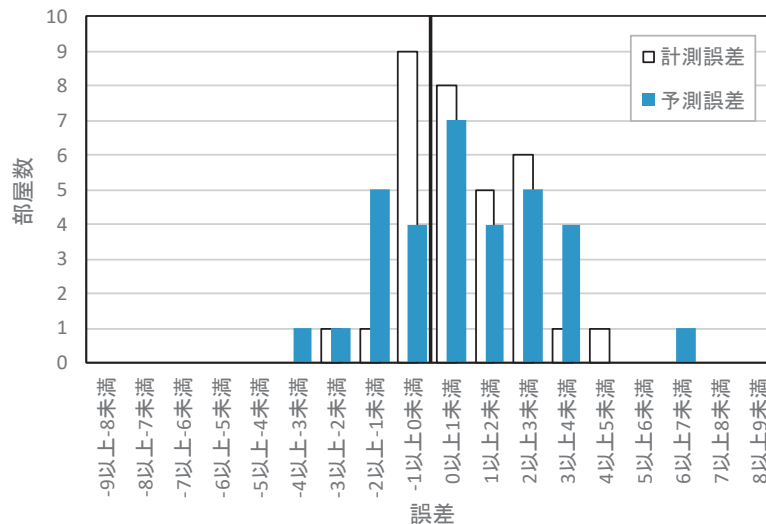


図 4.3 船舶 A の計測誤差と船舶 B の予測誤差

4.2 全データを教師データとした騒音学習と予測

つぎに、55 隻の船舶を対象に 10,000GT 以上の船舶と 10,000GT 未満で学習を分けて行い、それぞれの予測を行った。ニューラルネットワークによる騒音予測誤差のヒストグラムを図 4.4 に示す。CASE1 は、対象船舶は、学習を行わず、他の 54 隻の学習結果を基に予測を行った結果である。CASE2 は、対象船舶も含めた 55 隻の学習を行い、55 隻の学習結果を基に予測を行った結果である。予測誤差はニューラルネットワークによる予測結果から計測結果を差し引いた値である。

CASE1 は、予測誤差が-10dB(A)未満であった部屋が多数みられる。その内容を確認したところ、居住区が前方に位置した船舶の部屋が多数を占めていた。学習を行った船舶には、居住区が前方に位置した船舶がなかったことが原因である。ニューラルネットワークによる予測は、学習したパターンから計算されるものであるため、新たなパターンに相当するこのような船舶では、予測誤差が大きくなる。

CASE2 は、対象船を含めた 55 隻全てを学習しているため、CASE1 でみられたような新たなパターンがなく、予測誤差も改善している。学習データが蓄積してくるほど、CASE2 のような分布、さらに精度が増していくものと思われる。予測誤差頻度分布は、マイナス側にバイアスがかかっている。

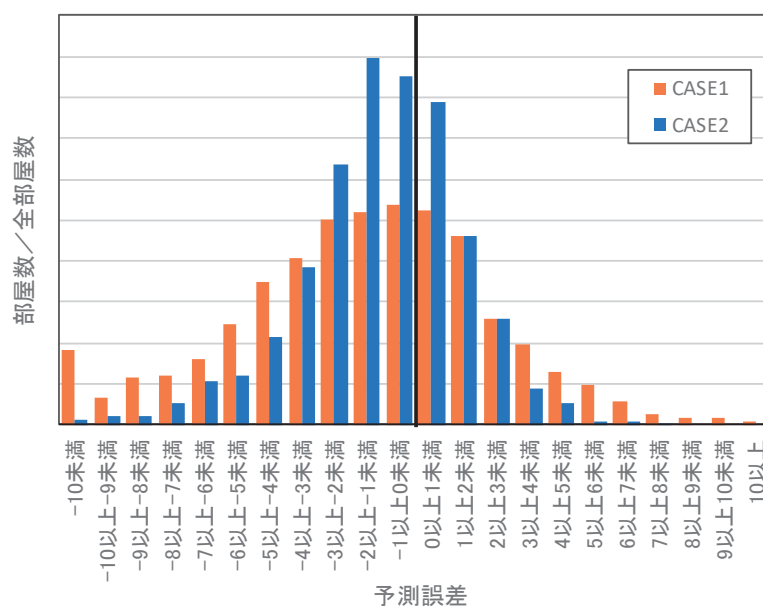


図 4.4 ニューラルネットワークによる騒音予測誤差

4.3 誤差の検証

ニューラルネットワークによる騒音予測誤差が大きかった部屋では、**図 4.5 右図**のような騒音スペクトルがみられた。**図 4.5 左図**は、当該部屋で行った音源探査の結果であり、天井に設置されたエアコン吹き出し口に騒音レベルを重ねて表示した画像である。**図 4.5 左図**から、この部屋はエアコンの吹き出し口のノイズが大きかったことが分かる。予測誤差が大きかったケースは、このようにローカルなノイズが原因の部屋がほとんどであった。**図 4.4** をみると、騒音予測誤差の頻度分布がマイナス側（予測結果よりも計測結果の方が大きい）にバイアスがかかっている。ニューラルネットワークによる騒音予測結果は、ローカルなノイズを含めない予測結果になっていることが考えられる。

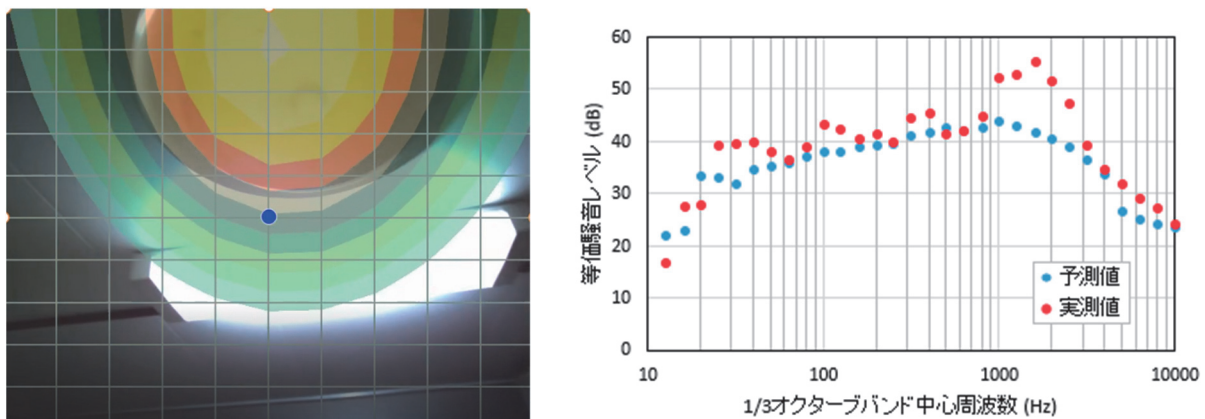


図 4.5 ニューラルネットワークによる騒音予測誤差が大きかった部屋の騒音スペクトルと音源探査結果

4.4 ニューラルネットワークの特徴と今後について⁶⁾

ニューラルネットワークによる騒音予測は、騒音予測精度・信頼性を向上させるためには、船種や船体構造の多様性を増しながら、ノイズの少ない騒音計測データ（教師データ）数を増やしていく必要がある。なお、4.3 で述べたエアコンノイズを低減するためには、エアコンの騒音予測を別途モデル化して評価する必要がある。そこで、民間クラウドサーバーを活用した Web アプリケーションシステム(AI-AcousticNet)を製作した。AI-AcousticNet は、**図 4.6** に示すように、サーバー側（管理者）が、ユーザー管理、システムメンテナンス、SOM マッピング管理（類似船の予測）、騒音レベルの学習、機関・内装品の管理等を行う。

ユーザーは、騒音対策を検討する造船所を想定している。事業所ごとにユーザー名とパスワードで管理される。ユーザーはパソコン端末で騒音予測に必要な入力データを作成し、作成した入力データは Internet を介してサーバー側に送信する。送信された入力データに基づいて、学習済みのシナプス重みを利用してニューラルネットワークによる騒音予測計算を行う。ここでの騒音予測計算は、サーバーに保存されている予め学習されたシナプスの重み（保存されている全船舶を対象に学習したシナプスの重み）を利用して行われる（**図 4.6** 標準船予測）。全船舶を対象にした学習は、隻数が増加するにつれ、管理者が定期的に行う。ユーザーは、サーバーから送信されてきた騒音予測結果（騒音スペクトル等）を、ユーザーの端末画面で閲覧する。

保存されている船舶の隻数が少ない段階では、類似船（又はその集団）を対象に学習したシナプスの重みを利用した騒音予測の方が、予測精度がいいことも考えられる。しかしながら、SOM を実施した後、類似船（又はその集団）を対象に騒音レベルの学習を行うには、時間を要する。したがって、まずは速報として、全船舶を対象に学習した重みで騒音予測計算した結果をユーザーに送る（**図 4.6** 標準船予測）。類似船（又はその集団）を対象に学習した重みで計算した結果は、後日送信する仕組みとなっている（**図 4.6** 類似船予測）。

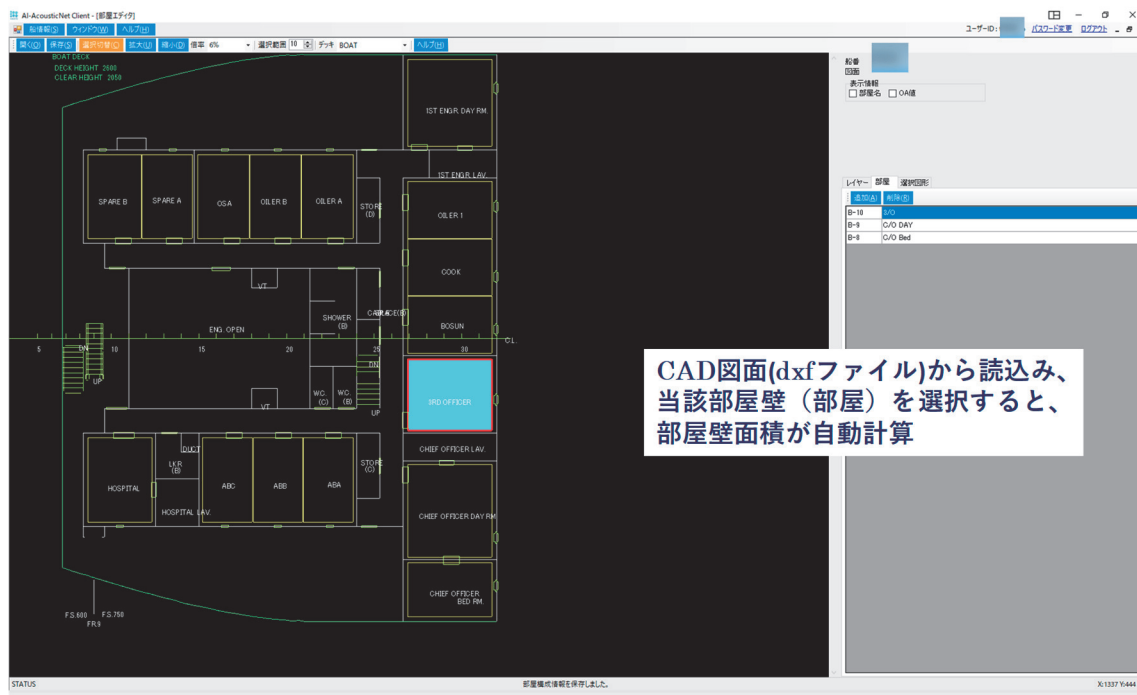


図 4.8 ユーザーの入力画面（部屋情報）

5. まとめ

経験的手法である Janssen 法による騒音予測手法を参考に、ニューラルネットワークによる騒音予測手法を構築した。騒音レベルの学習及び予測にあたっては、多層パーセプトロンタイプのニューラルネットワークを利用した。過学習が起こることなく安定した学習結果を基に騒音レベルの予測を行うことができた。類似船の検索にあたっては、SOM（自己組織化マップ）タイプのニューラルネットワークを使用した。多層パーセプトロンタイプのニューラルネットワークは、教師あり学習である。教師あり学習である多層パーセプトロンタイプのニューラルネットワークによる予測問題は、教師データが充実してくることによって、予測精度は向上する。IMOにおいて、騒音コードが義務化されたことにより、騒音計測が義務付けられた。したがって、騒音計測データを蓄積していき、これらを有効に活用すること、すなわち騒音対策の設計に有効活用することができる。そのため、本研究では、騒音計測データを一元管理するために、教師データとなる騒音計測データをサーバー側で一元管理する Web アプリケーションシステム (AI-AcousticNet) を製作した。本 Web アプリケーションシステムをサーバー側では、騒音レベルの学習及び予測、SOM マッピング管理、機関・内装品等予測計算に必要なデータベース管理等を行う。クライアントは、騒音対策を検討する造船所を想定している。本 Web アプリケーションサービスでは、利用者の入力データ作成を支援する仕組みを提供する等、利便性を高める機能も用意した。

本 Web アプリケーションシステムをベースに、ローカルな騒音レベルの予測や騒音予測の上流に位置する振動計算・評価等処理する機能の追加など、本 Web アプリケーションシステムの発展は、ひきつづき行っていく予定である。

参考文献

- 1) CODE ON NOISE LEVELS ON BOARD SHIPS, International Maritime Organization (IMO), (2012)
- 2) J. H. Janssen, J. Buiten, On acoustical designing in naval architecture, Inter noise73(1973), pp. 349-356
- 3) Diederik P. Kingma, Jimmy Lei Ba: ADAM: A METHOD FOR STOCHASTIC OPTIMIZATION, ICLR 2015

- 4) 徳高平蔵, 藤村喜久郎, 自己組織化マップとその応用事例, 日本神経回路学会誌, Vol. 10 No.3 (2003), pp147-157
- 5) 平方勝, 馬沖, ニューラルネットワークによる騒音予測, 日本船舶海洋工学会講演論文集, 第 25 号 (2017) , pp.637-642
- 6) 平方勝, 馬沖, 谷口智之, ニューラルネットワークによる騒音予測 Web アプリシステムの構築, 第 18 回海上技術安全研究所発表会 (2018)