

# 工場内データ分析案件の推進を目的とした 時系列センサデータに対するパターン認識のための センサプラットフォームの提案

廣瀬 雅治<sup>1,2,a)</sup> 荒川 陸<sup>1</sup> 稲見 昌彦<sup>1</sup>

概要：本論文では工場現場における9軸センサのデータ取得からモデリング、また予測モデルの検証実験を一気通貫で行うことができるセンサプラットフォームを提案する。工作機器や作業員に取り付けた小型のEdge Deviceから9軸センサデータを取得することで、物体の振動や移動の簡易な計測が可能であり、工場で取得したデータや予測モデルのFeasibility Studyを短期間かつ低コストで行うことができる。システムの検証実験として、センサプラットフォームを用いて実際に工作機器の動作分類モデルを生成し、100%の精度で工作機器の駆動状態を分類することに成功した。工場内データの活用推進を目的としたフレームワークとしての応用を考案する。

## A Proposal about Sensor Platform for Patter Recognition for Time Series Sensor Data to Prompt Industry Data Analysis Project

### 1. はじめに

データに潜むパターンを見つけ出すことは効率化や問題解決の根源であり、事象に対して規則性を見出すことで未知のデータに対して予測を行うことができる。この規則性を、数理的アルゴリズムを用いて自動的に探索することが機械学習であり [1]、ある条件において一定以上の精度で未来を予測することは多大な利益を生む。ユーザ属性の分析によるオンライン広告の高品質化 [2]、力覚信号の分析による製造装置の故障予知 [3]、化学物質の有害性に関するデータベースの学習及び危険性のモデル化による毒性予測 [4]、タンパク質構造の予測 [5] など、機械学習の応用は多岐に渡り、様々な分野で成果を上げている。

機械学習は Industry4.0 [6] における「物流の最適化」や「資産の利活用」においても主要な技術要素として位置づけられている。Industry4.0 はドイツ連邦政府が先導するハイテク戦略プロジェクトである。設計、開発、生産に関

連するデータの蓄積が、コンピュータと製造装置が一体となって駆動するスマート工場により可能となり、それらのデータを活用することで製造機器の自律的な動作、材料の使用及び供給プロセスの改善と最適化を実現することを目指している。工場内はデータと課題の宝庫であり、データを活用することで工場内の物理的事象をある程度高い精度で予測することは、故障予知、歩留まりの改善、行為保証やエルゴノミクスによる作業負荷の改善を実現する可能性がある。

しかし既存の製造装置の内部にセンサーとコンピュータを組み込み、装置の稼働に関するデータを常時収集するシステムの構築は高いコストがかかるため導入が難しい。工場内データの活用プロジェクトにおいて、工場の担当者は「取得できるデータから有用な情報が得られるのであれば高いコストがかかったとしても導入したい」と考えるが、一方でデータ分析担当者は「データから有用な情報が得られるかどうかを検証するためには、データを取得する必要がある」と考えるのが妥当である。このように判断と結果が循環し端緒を同定できないサーキュラーな状態はプロジェクトの鈍化の原因となる。

<sup>1</sup> 東京大学  
University of Tokyo  
<sup>2</sup> トヨタ自動車株式会社  
Toyota Motor Corporation  
<sup>a)</sup> hirose@star.rcast.u-tokyo.ac.jp

そこで我々は9軸センサを用いたセンサプラットフォームを提案する。このプラットフォームを用いることで、9軸センサのデータ取得からモデリング、工場現場にモデルを適用した際の検証実験までを一気通貫で行うことができる。工作機器や作業員に9軸センサモジュールを組み込んだ小型のEdge Deviceを取り付けることにより、大規模なシステムを使用せずに物体の振動や移動を計測することが可能であり、短期間かつ低コストで予測モデルのFeasibility Studyを行うことができるため、本研究の課題であるシステムの導入コストの問題を解決できるのではないかと考えた。

本論文ではセンサプラットフォームのシステムデザインについて記述する。また実際にセンサプラットフォームを用いて工作機器の動作を分類するためのモデルを生成し、一連の分析プロセスをシステムの検証実験として記述する。

## 2. 関連研究

### 2.1 工場内におけるデータ分析の活用

一章で述べた通り、工場は分析対象データと解くべき課題の宝庫と言える。工場内データの利活用に関して、Rodriguezらは外力を要因とした作業の失敗が頻発するアームロボットの組み立て作業において、6軸の垂直多関節型ロボットの力覚信号に対するモデリングにより、作業の失敗を予測することに成功した[3]。またAmiratらはヒルベルト変換を用いた故障検知手法を提案、風力発電のタービンの電気的特性からベアリングの故障予知を行った[7]。またGarcia-Seguraらは電弧炉の電圧と電流の測定データを用いたニューラルネットワークのモデリングに基づいて、エネルギー効率と生産性の向上を行った[8]。これらの事例の様に、データの利活用を行い動作の自律化やプロセスの最適化を実現することはIndustry 4.0の目指す一つのコンセプト[6]である。本研究においても、簡易に工場内のデータ取得を行うことができるプラットフォームの構築により、工場内データの利活用を推進することを目的としている。

また工場内データの利活用は工場現場だけではなくシステムベンダーの関心も集めており、自社の強みを生かした多様なフレームワークが提案されている。Alibabaはクラウド連携のためのAPIやクラウドストレージを含むIoT Platformを提供しており、ユーザーは迅速なIoTアプリケーションを構築することができる[9]。また富士通は独自の強固なセキュリティソフトウェアを活用することができるソフトウェアフレームワーク、FUJITSU Cloud Service for OSSを提供している[10]、IntelはOEMやソフトウェアベンダーとのパートナーシップを活用し、拡張性に優れたシステムアーキテクチャーのリファレンスデザインとしてIntel IoTプラットフォームを公開している[11]。これらの事例が目指しているのはEdge DeviceからCloudま

でのシステムを一気通貫でインテグレーションするためのフレームワークである。汎用的なフレームワークを確立することで、低価格かつ短期間でのシステムの導入を目指している。これらは工場内データの利活用を推進するための構想である一方で、取得データのFeasibilityを明らかにする機能は含まれていない。本研究はクラウドまで含めた工場内の大規模なシステムではなく、物理的事象に近いEdge Deviceにより取得できるデータが、工場内の物理的事象を予測するにあたってどの程度有用であるかを、短期間かつ低コストで検証することに焦点をあてている。

センサプラットフォームでは9軸センサのデータ取得を想定している。機械学習による回帰や分類などの予測は画像データが用いられる場合が多いが、工場内でのシステムの運用を考慮すると、カメラの設置が必須であることはシステムの適用環境に物理的な制限が生じる可能性がある。例えば、狭い空間内で作業をする作業員の様子を俯瞰視点で撮影することは難しい。カメラによる画像データは一枚一枚のデータが高次元である一方でデータ取得環境がカメラを設置できる環境に限られる。またハイスピードセンシングにより時系列方向に密な情報を取得することが難しい。一方で9軸センサは小型であり、Wearable Deviceやスマートフォンを用いてもデータを取得することができる。画像データに比べるとデータ取得環境のリミテーションが少なく、物理的事象そのものに影響を与えることなくデータ取得を行うことができる。また9軸センサデータは画像データに比べると低次元のデータであるが、画像に比べてハイスピードセンシングも容易である。そのため適切な特徴量エンジニアリングにより、物理的事象を十分に説明できる学習データを作りこむことで、9軸センサの長所を生かした予測を行うことが可能であると考えられる。活用の幅は大変広く、自動車、製造機器、スマートフォン、ラップトップや様々なガジェットの内部にも組み込まれ人々の生活の中に普及している。Primesapは6軸センサを用いた高速センシングで関節角度を正確に取得し、スポーツ選手の身体的故障予知に応用している[12]。Moffは6軸センサデータを内蔵したWearable Deviceを販売し、スマートフォンアプリとの連携で高度なハビリティサービスを展開している[13]。またYoungらは6軸センサを用いた腕部装着型Wearable Deviceを提案し操作性の高い入力デバイスを提案した[14]。これらの研究は6軸センサというレガシーなセンサデバイスを自らのビジネスや研究目的の達成のために活用し、独自の使い方を確立した事例であると言える。本研究でも9軸センサの様々な用途を想定しており、自由度の高い9軸センサの使い方ができるような柔軟性のあるプラットフォームを構築することで、工場内データの利活用を目指す。

## 2.2 特定用途に特化したプラットフォーム構築

特定の用途に特化したプラットフォームの構築は工数の削減、技術的障壁の低減、技術の普及など多大な恩恵をもたらす。本研究では、工場内の物理的事象に関するデータを簡易に取得することができるセンサプラットフォームを用いることで、工場内データの Feasibility Study の簡易化を目指している。構造力学において、長山らは MEMS センサを用いた振動情報を取得することができるプラットフォームを開発し、建造物のデータ取得によるヘルスケアモニタリングを行うためのシステムとして提案した [15]。安価な MEMS 型センサ、無線通信と演算処理を兼ね備えたスマートセンサでデータの収集を行い、健全性評価や判断の自動化及び自立化を目指している。無線通信を備えたセンサモジュールによるデータを収集と利活用は本研究で提案するセンサプラットフォームにおいても重要な機能の一つであり、加えて工場内データを用いたモデリングの機能も想定している。またインタラクティブ技術におけるプラットフォームとして、南澤らは触覚情報の簡易な取得、加工やシステム実装を行うことができる TECHTILE toolkit を提案した [16]。触覚情報のデータ取得や加工に関するノウハウをツールキットという形で提供することで、エンジニアリングを専門としないユーザーも触覚情報を用いたアイデアを提案することができる。

本システムはデータの取得からモデリングまでを一気通貫で行うことを意図したシステム設計を行っており、データ分析におけるデータクレンジングや可視化等、前処理に該当する作業を軽減するための機能も含まれる。データ分析業務において、前処理やデータの理解に多くの時間と人的工数が充てられており、一般的に 8 割ほどの工数がこれらの工程に費やされていると言われている [17]。この現状に対し、前処理の補助を行うためのソフトウェアツールは多数提案されている。NVIDIA は RAPIDS という GPU を活したデータ分析業務の計算処理のライブラリを OSS で提供している [18]。CUDA ライブラリの最適化、GPU の並列処理と広帯域幅のメモリアクセススピードの恩恵により、データの前処理の高速化を実現した。

本システムにはリアルタイムの可視化機能が含まれている。ユーザーはデータをリアルタイムで目視することで、有用な特徴量が含まれているかどうかをその場で確認することができる。その場でリアルタイムにデータを確認することにより、効率の良い分析サイクルの実現を意図している。データという無機質な数値の羅列を人間が視覚的に理解できる形式に変換することは、分析対象の特性や規則性を見つけるにあたって極めて重要な作業である。Talbot らはアンサンブル学習における適切な分類器の組み合わせをグラフィカルビューで可視化しながら構築することができる視覚化システムを提案した。可視化によるモデルの理解と探索を行うことにより、ユーザーは高精度のモデリン

グを迅速に行うことができる [19]。データの活用に焦点を当てた可視化システムに関して、Tableau は様々な形式のデータに対する接続から可視化をスムーズに行うことができるデータ分析プラットフォームを提供している [20]。同様に Metabase は可視化結果の情報共有を目的としたソフトウェアであり、OSS として提供されている。様々なデータ形式に対して、リアルタイムに処理を行いながら、チームにその結果を共有することができる。

データのモデリングにおいて、本センサプラットフォームは分析に関するノウハウを提供するため Jupyter Notebook 形式の分析コードを Tutorial として提供している。学習データを用いたモデリングの工程はデータ分析業務の後工程に分類され、アルゴリズムに対する理解やコーディングによるパラメータのチューニングが必要となる。MatrixFlow は Deep Learning のパラメータチューニングを容易に行うことができる独自のソフトウェアを開発することで、後工程の効率化を目指している [21]。また Google が提供している AutoML は Vision API によってプログラミングの知識が無くても画像認識のためのモデリングを行うことができるサービスである [22]。これらのツールは効率化だけでなく、技術的な障壁を取り除くことで機械学習の適用の幅を広げることも意図している。本プラットフォームにおいても、初学者が機械学習を用いた問題解決に挑戦できること、またこのツールを通じて分析のノウハウを習得することを期待している。

## 3. システム概要

センサプラットフォームを使用した分析フロー、またハードウェアとソフトウェアの詳細に関して記述する。システム図を図 1 に示す。センサプラットフォームは下記の 3 つの部分により構成されている。

- (1) センサデータの取得と送信を行う Edge Device
- (2) データの受信/加工/可視化/保存を行う Host Machine
- (3) Edge Device と Host Machine の通信接続のための Gateway

ユーザーは無線経由で Edge Device の取得データを Host Machine へ送信し、可視化結果を Host Machine 上でリアルタイムに確認することができる。また任意のタイミングでデータの保存が可能であり、保存したデータに対して Jupyter Notebook 上で記述されたコードに則って前処理/可視化/モデリングを行うことができる。加えてモデルの予測機能を組み込んだシステムをデータ取得環境と同等の環境で構築し、検証実験を行うことができる。これらの機能により、ユーザーはプラットフォーム上で 9 軸センサのデータ取得からモデリング、実際の工場現場にモデルを適用した際の検証実験までを一気通貫で行うことができる。

Edge Device は小型であること、I2C でのセンサモジュールとの接続が可能であること、Wi-Fi による接続が可能

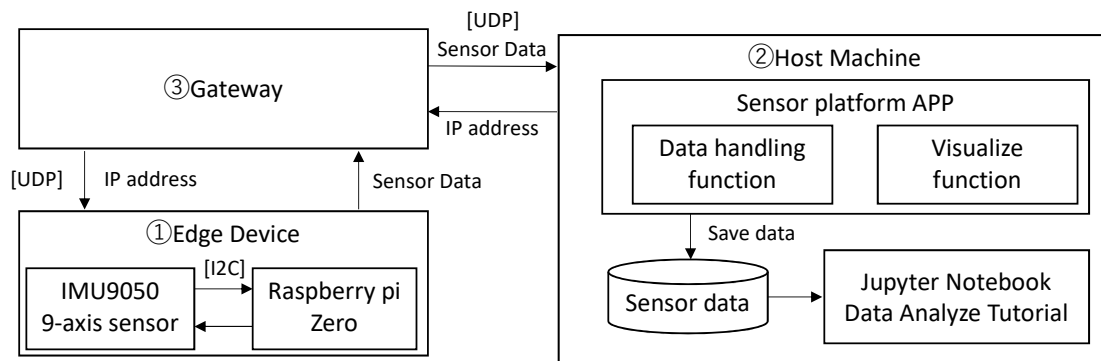


図 1 システム図  
Fig. 1 System diagram.

であることを要件とし、シングルボードコンピュータの Raspberry pi Zero WH を用いて構築した。ボードのサイズは 65mm × 30mm ,Broadcom BCM2835 の SoC で構成されている。I2C 経由で接続した MPU9250 から 3 軸加速度, 3 軸ジャイロ, 3 軸コンパスデータを取得し, UDP のソケット通信により 180Hz 前後のセンサスピードで Host Machine へデータを送信する。図 2 に Edge Device の外観を示す。LiPo バッテリーによる給電で起動することができ, 110mAh のバッテリーで 15 分, 500mAh のバッテリーで 1 時間以上の連続した駆動が可能である。また USB による給電で動作することも可能である。

Host Machine は Python 環境が構築されたミドルスペック程度のコンピュータでの動作を想定している。4 章と 5 章の検証実験では Windows10 を搭載した Lenovo の ThinkPad X260(Core i5-6200 を搭載, 実装メモリ 16GB) を使用した。Host Machine 上での可視化画面の一例を図 3 に示す。ジャイロセンサ, 加速度センサ, 磁気センサそれぞれの 3 軸方向の実値, 折れ線グラフによる時系列変化, センサスピードが表示されている。ユーザーは必要に応じてカスタマイズを行い, 独自の可視化画面を開発すること

ができる。リアルタイムに可視化結果を確認することで, ユーザーはデータの Feasibility を早い段階で確認することができる。これにより効率の良いデータ分析のフローを実現し, 分析プロセスにおける試行回数を増やすことを意図している。

ユーザーはキーを押すことでデータの保存を行うことができる。GUI 上の仮想的なボタンではなく物理的なキーによる操作は, 押し間違いなどの失敗を防ぐこと意図した設計である。高速なデータのハンドリングが可能であること, またデータ分析業務従業者が親しみやすいデータ構造であることから, データのハンドリングは二次元の Numpy Array で行っている。保存したデータは予め指定したフォルダに CSV 形式で出力される。

保存後のデータに対する前処理や後工程の処理は, ユーザーが取り組むプロジェクトの内容に依存する。そのためセンサプラットフォーム自体の機能としては, 必要となるであろうデータの前処理や後工程のプロセスを想定し, Jupyter Notebook 上で記述されたコードを Tutorial という形式で提供する。これによりユーザーは, Jupyter Notebook 上で Tutorial のコードを参照しながら分析作業

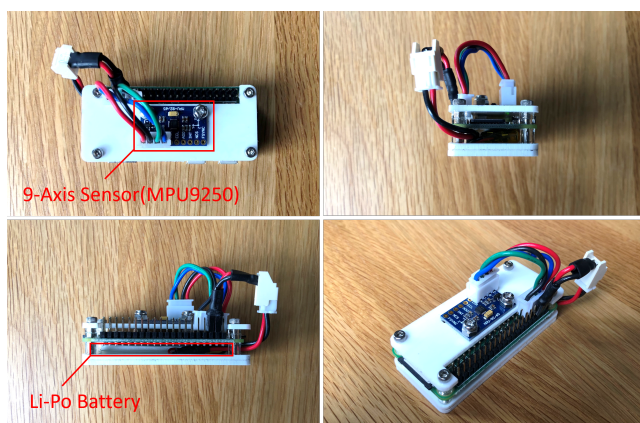


図 2 Edge Device の外観  
Fig. 2 Appearance of Edge Device

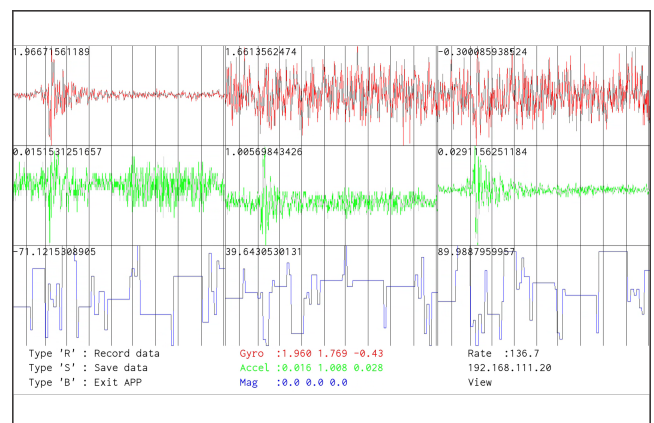


図 3 リアルタイムの可視化画面  
Fig. 3 Real-time Visualizing

を行うことができる。

本プラットフォームは Gateway をハブとしたネットワーク構成を採用している。これは複数の Edge Device を用いたデータ取得環境やアクチュエータを含んだシステムの構築など、様々なシステム構成に対してプラットフォームを柔軟に対応させることを意図している。

加えてこのプラットフォームのソースコードをリファレンスとして、生成モデルによる予測を機能として組み込んだシステムを構築することができる。その際、データ取得時とシステム実装時のシステム構成の差異はモデルの評価を行う上でノイズの要因となることが懸念される。そのため Edge Device, Host Machine, ネットワーク構成などのシステム構成は、可能な限りデータ取得環境と同等の構成で構築することを意図している。また機械学習エンジニアが検証システムを構築することを意図して、センサプラットフォームはデータ分析業務で一般的に使用される Python を用いて開発した。

## 4. 検証実験

### 4.1 目的

センサプラットフォームを使用することで、物理的事象を解釈するためのデータの取得が可能かどうか、また一連の Feasibility Study をある程度簡易に行うことができるかの検証を本実験の目的とする。検証実験として、工場の工作機器からデータを取得する状況を想定し、実際の工作機器からデータを取得して可視化やモデリングなど一連の分析を行った。

### 4.2 検証実験の概要

工作機器 (RYOBI の卓上電動系鋸装置) を「停止」「低出力で駆動」「高出力で駆動」の 3 種類の状態で 1 分間駆動させ、工作機器の筐体外部取り付け付けた Edge Device から各動作状態における 9 軸センサデータを取得した。3 種類のラベルで分類された 9 軸センサデータを学習データとし、工作機器の状態を「停止」「低出力で駆動」「高出力で駆動」の 3 種類に分類するモデルを生成した。学習データと同様の環境で 30 秒間データを取得し、これをテストデータとしてモデルの精度を評価した。このようにトレーニングデータとテストデータを取得の段階で分けた理由は、取得したデータからトレーニングデータとテストデータをランダムに分割してモデリングを行うクロスバリデーションの手法では、時系列データを扱う場合にデータリークが生じる可能性があるためである。またデータ取得は 120Hz 前後のセンサスピードで行った。

分析のプロセスは探索的データ分析に則って行った。モデリングのアルゴリズムや特徴量への変換処理を、可視化及び集計作業行いながらアドホックに定義する分析プロセスである [17]。データの素性が不明な場合に有効なため、

データ分析業務において広く使われる手法である。

### 4.3 データ取得と可視化結果

データ取得時の様子を図 4 に示す。系鋸が動くと同時に波形が変化し、工作機器の状態により 9 軸センサーの値が変化している様子が確認できた。工作機器の駆動状態を反映する情報がデータに含まれることが期待できる。

取得したデータの可視化結果を図 5 に示す。青が「停止」、緑が「低出力で駆動」、赤が「高出力で駆動」のデータである。加速度の x 軸が分類のために有用な情報が十分に含まれていることが可視化結果から確認できる。この結果を踏まえて今回は学習に使用する特徴量を加速度の x 軸のみに絞り込むことに決定した。

### 4.4 特徴量

可視化結果から、特徴量に関しては各時間の瞬時値を扱うよりも、時系列方向の情報を加味した特徴量を作ることが、駆動状態の予測に対しては有効であると考えられる。これを踏まえ、任意の時刻  $t$  における任意のラベル  $C$  の特徴量  $F(t, C)$  について定式化する。センサーのサンプリング周波数を  $N$ 、時刻  $t$  におけるセンサー値  $A(t, C)$  とする。信号波形に対しサイズ  $\Delta t$  の窓を考え、時刻  $t$  におけるセンサー値集合  $X(t, \Delta t, C)$  を、

$$X(t, \Delta t, C) = \{A(\tau, C) | t - \Delta t < \tau \leq t\} \quad (1)$$

と定義する。ただし  $\tau$  はサンプリング時刻を表す。このようにして定めた集合  $X(t, \Delta t, C)$  に対する 7 つの統計量 (平均, 標準偏差, 最小, 25%点, 中央値, 75%点, 最大) を  $t$  における特徴量  $F(t, C)$  とする。今回の検証では  $\Delta t = 1$ ,  $N = 120$  で学習データを作成した。

### 4.5 モデリングと結果

今回は解釈性の高さを重視して、決定木の数を 100, 決定木の深さを 3 とした Random Forest でモデリングを行った。結果を表 1 に示す。正解率は 100%であることから、物理的事象をある程度解釈可能なデータの取得が可能であ

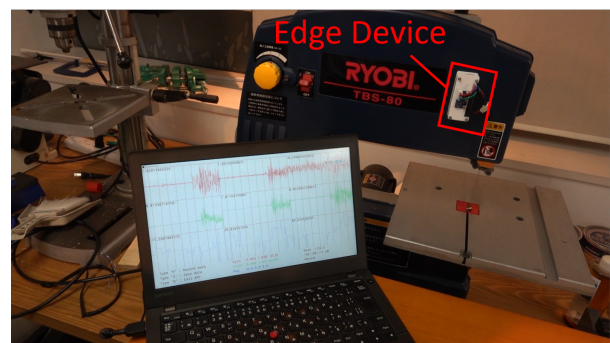


図 4 データ取得時の様子  
Fig. 4 Collecting data.

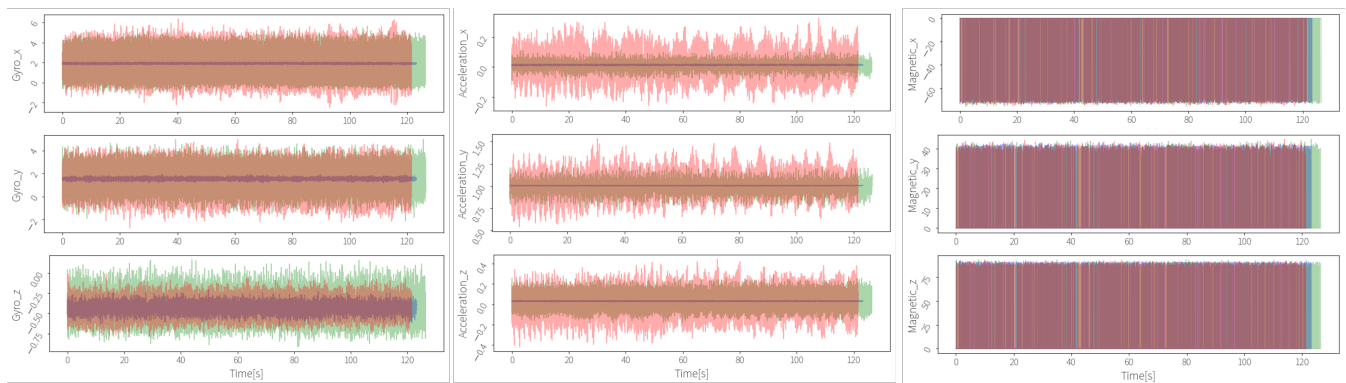


図 5 左から、ジャイロデータ、加速度データ、磁気データ  
Fig. 5 Gyro data, Accelerate data, Magnetic data

ると言える。またデータの取得、前処理のコーディング、モデリングの評価までの工程を数時間程度で終わることができた。実環境である工場現場の検証においても、本検証実験と同様に簡易に Feasibility Study を行うことができるのではないかと考えられる。

## 5. 追加実験

### 5.1 目的

4 章では工作機器の動作分類を想定課題として検証実験を行なった。一連の分析フローから、可視化の有用性、また一連の分析フローを短期間かつ低コストで行うことができることを確認した。しかし 4 章の実験ではモデルの精度が 100% であるため、センサーデータに基づいて作られたモデルが物理的事象をどの程度正確に分類できたかは明確ではない。そこで、目的変数である工作機器の駆動状態を「停止」「低出力で駆動」「高出力で駆動」の 3 パターンから、「停止」「最大出力で駆動」の間を 12 パターンに段階分けしたものに变更し、これを目的変数として同様の実験を行った。想定課題の判別の難易度を上げることで、センサープラットフォームがどの程度繊細な物理的事象を分類可能かを明確にすることを目的としている。

### 5.2 概要

ラベルの数が 3 種類から 12 種類に変更したこと、学習に使用するセンサーデータを 1 軸方向 (加速度の x 軸) に限定せず 6 軸方向 (ジャイロの 3 軸と加速度の 3 軸) を使用したこと、学習のアルゴリズムとして Random Forest, Support

Vector Machines(SVM), Multilayer Neural Network の 3 種類を用いたこと以外は 4 章で行った検証実験と同様であるため、詳細な実験条件は割愛する。

### 5.3 結果

ヒートマップとして出力した分類結果を図 6 に示す。左図が 4 章と同様に 1 軸方向を特徴量として Random Forest で学習したモデルの結果である。正解率は 79% だった。Random Forest のパラメータに関しては Grid Search でチューニングを行い、決定木の数を 100、決定木の深さを 12 として学習を行った。右図が 6 軸方向を特徴量として Random Forest で学習したモデルの結果であり、正解率は 99.9% だった。3 種類のアルゴリズムを用いてスコアリングを行ったがアルゴリズムによるスコアの差は見られず、精度は特徴量の種類に依存した結果となった。1 軸方向のセンサーデータを特徴量として使用した場合は 80% 前後、6 軸方向のセンサーデータを特徴量として使用した場合は 98%~100% の精度だった。

### 5.4 まとめと考察

4 章の検証実験を踏まえ物理的事象をどの程度正確に分類できたかを調べるための検証実験を行った。取得データからモデリングを行った結果、12 種類の工作機器の駆動状態を 1 軸方向 (加速度の x 軸) のセンサーデータのみで 80% 前後、6 軸方向 (ジャイロの 3 軸と加速度の 3 軸) 方向のセンサーデータを用いて 98%~100% の精度で分類することに成功した。

1 軸方向のみの特徴量を Random Forest で学習したモデルの精度が低い理由は、共振によるノイズが原因であると考えられる。駆動レベル 9 の加速度センサのデータを図 7 に示す。数十ミリ秒の小さな周期に加えて 2 秒間隔の周期が観測できる。同様の共振は駆動レベル 8 でも観測できた。特に誤答率が高い駆動レベル 8 及び 9 のデータに関しては図 7 の様なうなりが学習のノイズとなり正答率の低下を引き起こしたと考えられる。対策として、右図の様に

表 1 モデルの精度  
Table 1 Accuracy

|      |        | 真の結果 |      |        |
|------|--------|------|------|--------|
|      |        | Stop | Weak | Strong |
| 予測結果 | Stop   | 3904 | 0    | 0      |
|      | Weak   | 0    | 3482 | 0      |
|      | Strong | 0    | 0    | 3907   |

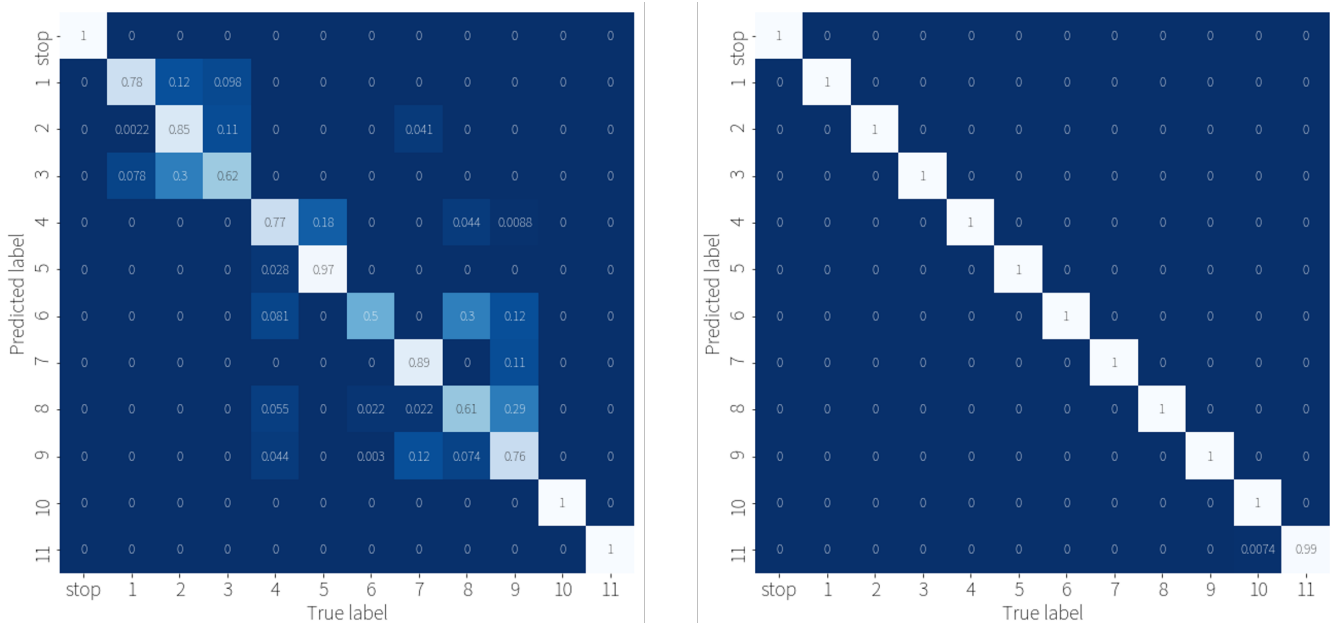


図 6 12 種類の分類の結果．左図が 1 軸方向，右図が 6 軸方向のデータで学習  
 Fig. 6 Results of 12-labeled classification test. Training with 1-axis(left), 6-axis(right).

共振の影響がないセンサデータの特徴量を学習に組み込むこと，また今回は 7 つの特徴量を用いて学習を行ったが，これに加えてノイズの影響に強い統計量を組み込んで特徴量を作ることが考えられる．

本実験で観測できた物体の共振はセンサプラットフォームのリアルタイムの可視化によりデータ取得段階で観測することができた．実際のデータ取得環境でも今回のようなノイズになり得るデータの特徴が発生することは珍しくない．当然一つ一つのデータに対し可視化処理を行い具さに確認することで事前に対処できるノイズだが，高次元なデータや長時間取得したデータなど可視化が難しい場合は対処することができず，モデルの精度下げる原因になり得る．このようなデータの癖をデータ取得時に確認できるのは，実際のデータ分析業務において有用であると考えられる．

## 6. まとめ

本論文では 9 軸センサのデータ取得からモデリング，現場における予測モデルの検証実験までを一気通貫で行うことができるセンサプラットフォームを提案した．工作機器や作業員に取り付けた小型の Edge Device により 9 軸センサデータを取得することで，物体の振動や移動の簡易な計測が可能であり，短期間かつ低コストで予測モデルの Feasibility を検証することができる．システムの検証実験として，センサプラットフォームを実際に用いて工作機器の動作予測モデルを生成し，12 種類の工作機器の駆動状態を 98%~100%の精度で分類することに成功した．また簡易なデータ取得が可能だったこと，可視化やモデリングな

ど一連の分析フローを短時間かつ低コストで行うことができたことから，実際の工場現場においても有用なフレームワークとして活用できるのではないかと考えられる．

Feature Work として，より高度な可視化機能の実装，ユーザスタディによる可視化効果の評価，カメラ映像と組み合わせたマルチモーダルなデータを取得する機能，及び複数の Edge Device からデータを取得する機能の開発を予定している．現場の要望に応じられるシステム実装が完了した段階で，実際の工場現場での活用を開始する．

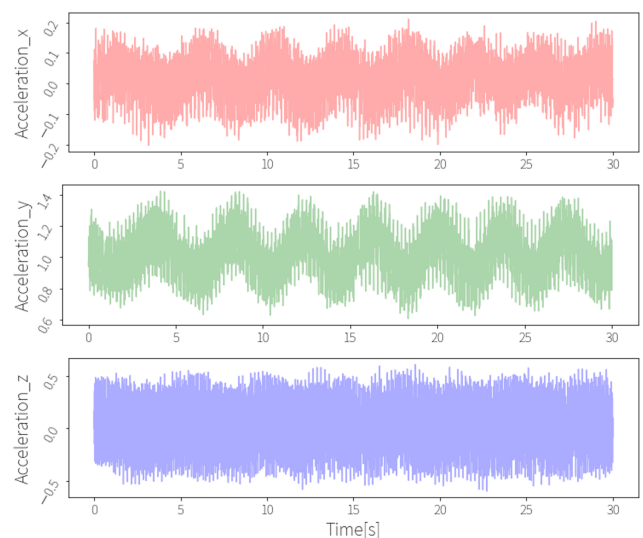


図 7 駆動レベル 9 の加速度データ  
 Fig. 7 Accelerate data with level 9

## 参考文献

- [1] Bishop, C. M.: *Pattern Recognition and Machine Learning*, Springer (2006).
- [2] Tolomei, Gabriele, et al. "interpretable predictions of tree-based ensembles via actionable feature tweaking." *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 2017.
- [3] Rodriguez, Alberto, et al. "Failure detection in assembly: Force signature analysis." *2010 IEEE International Conference on Automation Science and Engineering*. IEEE, 2010.
- [4] Luechtefeld, Thomas, et al. "Machine learning of toxicological big data enables read-across structure activity relationships (RASAR) outperforming animal test reproducibility." *Toxicological Sciences* 165.1 (2018): 198-212.
- [5] AlphaFold: Using AI for scientific discovery, 入手先 (<https://deepmind.com/blog/alphafold/>) (参照. 2019.8.19).
- [6] What is the Plattform Industrie 4.0?, 入手先 (<https://www.plattform-i40.de/PI40/Navigation/DE/Home/home.html>) (参照. 2019.8.19).
- [7] Amirat, Yassine, et al. "Hilbert transform-based bearing failure detection in DFIG-based wind turbines." (2011).
- [8] Garcia-Segura, Raul, et al. "Electric Arc Furnace Modeling with Artificial Neural Networks and Arc Length with Variable Voltage Gradient." *Energies* 10.9 (2017): 1424.
- [9] Alibaba Cloud IoT Platform, 入手先 (<https://jp.alibabacloud.com/product/iot>) (参照. 2019.8.19).
- [10] FUJITSU Cloud Service for OSS, 入手先 (<https://jp.fujitsu.com/solutions/cloud/k5/function/paas/iot-platform/>) (参照. 2019.8.19).
- [11] Intel IoT Platform, 入手先 (<https://www.intel.co.jp/content/www/jp/ja/internet-of-things/white-papers/iot-platform-reference-architecture-paper.html>) (参照. 2019.8.19).
- [12] Primesap, Live Trac, 入手先 (<http://www.primesap.com/livetrac.html>) (参照. 2019.8.19).
- [13] Moff band, <https://jp.moff.mobi/> (参照. 2019.8.19)
- [14] Young, Thomas S., Robert J. Teather, and I. Scott MacKenzie. "An arm-mounted inertial controller for 6DOF input: Design and evaluation." *2017 IEEE Symposium on 3D User Interfaces (3DUI)*. IEEE, 2017.
- [15] Nagayama, Tomonori, and Billie F. Spencer Jr. *Structural health monitoring using smart sensors*. Newmark Structural Engineering Laboratory. University of Illinois at Urbana-Champaign., 2007.
- [16] Minamizawa, Kouta, et al. "TECHTILE toolkit: a prototyping tool for design and education of haptic media." *Proceedings of the 2012 Virtual Reality International Conference*. ACM, 2012.
- [17] Tamraparni Dasu, Theodore Johnson (2003). *Exploratory Data Mining and Data Cleaning Wiley*. Series in Probability and Statistics
- [18] RAPID, 入手先 (<https://rapids.ai/>) (参照. 2019.8.19)
- [19] Talbot, Justin, et al. "EnsembleMatrix: interactive visualization to support machine learning with multiple classifiers." *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2009.
- [20] Tableau, 入手先 (<https://www.tableau.com/>) (参照. 2019.8.19)
- [21] MatrixFlow, 入手先 (<https://www.products.matrixflow.net/>) (参照. 2019.8.19)
- [22] AutoML, 入手先 (<https://cloud.google.com/automl/>) (参照. 2019.8.19)