目白大学　経営学研究
第18号　2020年　77−90

ビッグデータを活用した障害予測に関する実験的検証

An Experimental analysis for Detection of Hardware Failure Using Big Data

伊藤　利佳　　藤田　直行
（Rika ITO　Naoyuki FUJITA）

【要　約】

近年の計算機環境の大規模化に伴い，ハードウェアシステムの障害の影響も年々大きくなっている。そのため，ハードウェアの障害を事前に予測するシステムの構築が求められている。研究所や企業などにおいてハードウェア障害が発生すると，管理者は原因の究明と復旧作業などの対応に追われ，円滑なコンピュータ利用のサービスが妨げられる。しかしながら，障害の起きる原因は多様であり，パフォーマンスの低下やトラフィック状況などのハードウェアの内部情報を監視しているだけでは障害の予兆を捉えることは難しい。そのため，コンピュータの内部情報やシステムの設置状況などの外部情報を包括的に精査することによって障害の予兆を捉えるための研究を行っている。しかし，障害を起こしたハードウェアの状況把握のための内部情報を収集することは容易ではない。そこで，本研究では，ハードディスクに備えられているS.M.A.R.T.（Self-Monitoring, Analysis and Reporting Technology）情報を用いて機械学習を実施することによって，ハードディスク障害の予測に関する解析を行い，その結果を報告する。

キーワード：異常検知，機械学習，ビッグデータ，S.M.A.R.T.情報，ハードウェア障害

【Abstract】

The large-scale expansion of the computing environment in recent years has also seen an increase in hardware system failures. The construction of a system that would predict such hardware failures beforehand is therefore in demand. When hardware failures occur not only is the administrator pressed with tasks such as investigating the cause and recovery procedures, these failures also hinder the smooth services used by the computer users.

However it is difficult to detect failure symptoms by only monitoring the hardware internal information such as performance degradation. Hence, the system is needed that detects these failure symptoms by comprehensively examining the internal information of computers and external information such as the statuses of system installations. In this study, we performed an analysis on the prediction of hard disk failures by applying machine learning to the S.M.A.R.T. data generated from hard disk system and report the results.

Keyword： Anomaly detection, Machine learning, Big Data, S.M.A.R.T. information, Hardware failure

いとうりか：目白大学経営学部経営学科教授
ふじたなおゆき：宇宙航空研究開発機構　研究領域主幹
令和元年10月 4 日 受付
令和元年11月30日 改訂
令和元年12月 2 日 採択（紀要編集委員会）

1．はじめに

コンピュータシステムに障害が起きると，管理者は対応に追われ，ユーザはサービスが利用できなくなる。そのため，一般的にコンピュータシステムは死活監視システムやSNMP（Simple Network Management Protocol）と呼ばれる監視システムによって監視されており，問題があれば即座に管理者に通知されるしくみを取り入れていることが多い。そのため管理者はSNMPのふるまいを監視することによって，ハードウェア障害に対して迅速な対応ができるよう常に注意を払っている。近年はクラウドコンピューティングの普及により多数のマシンを抱える機関の増加や，多数のサーバを仮想的に集約した仮想サーバなどの普及により，1つの障害が甚大な影響をもたらすような環境も増加しており，事前の障害予測の意義・必要性は以前にも増して高まっている。そのためコンピュータのベンダーやソフトウェアメーカーにおいては障害を検知するためのソフトウェアの研究や開発もおこなわれている[1][15]。

ハードウェア障害の先行研究としては，現在のシステムとは異なるが，OSがエラー回数などの履歴を取得するしくみがあり，これらの情報の分析あるいは手法を検討することによってハードウェア障害の兆候を捉えるための研究がされている[6][9][15]。また，インターネットサービスを用いて，サーバの状態を恒常的に更新することによってハードウェア障害や通信障害などを回避しようという取り組みもなされてきた[12][14]。一方，近年仮想化マシンが急激に増加しているが，仮想化マシンは一台のハードウェアが故障するとそれに引っ張られて他のマシンにも影響が及ぶなど，予期しないふるまいが発生することもある.そのため，あえて仮想的な障害を発生させ，障害発生時の動作を評価する研究なども行なわれている[7]。また，ハードウェア障害に関する研究に限らず，異常検知という観点では，不正アクセスや機器障害など異常状態の検出などの様々な手法が提案されてきている[8][10]。

しかし,異常が起きた後に検知するための取り組みに対し,起きる前に予測することは簡単ではない。そのため,本研究では,予測に焦点をあてて提案および実験を行ない,その結果を報告する。

1－1．研究の背景

個々のハードウェアに対し，障害の予測が可能になれば突然のハードウェア障害に慌てて対応することもなくなり，ユーザも不本意にサービスを利用できなくなることはなくなる。また，障害が起きそうなハードウェアに対して，事前に計画的な準備をすることも可能となるだけでなく，原因究明のための手間や時間，人的コストも省略することができる。しかしながら，障害の起きる原因は多様であり，負荷のかかり方やパフォーマンスの低下，トラフィック状況などのハードウェアの内部状態情報を監視しているだけでは障害の予兆を捉えることは難しい。そこで，コンピュータ内から発せられる電磁波の微弱な変化などの内部物理情報だけでなく，システムの設置環境から得られる外部物理情報を包括的に精査することによって障害の予兆を捉えるなどの多面的な研究が重要となる。しかしながら，ハードウェア障害に関する研究の難しさは，そもそもデータを集めるのが難しいことから始まる。ハードウェアに障害が起きた際に，オペレーティング・システムがエラーメッセージを出し，それらのメッセージはログに記録される。これらのログ情報は障害が起きた後に検証用のデータとして利用することができる。しかし，障害はいつ発生するかが不明であり，現実に障害が起きるのを待っているとなると，データ解析に必要とされるボリュームのデータを集めるには非常に多くの時間が必要となる。また，コンピュータに負荷をかけて故意に障害を誘発することも可能ではあるが，現在運用中のシステムに対してこのような方法を用いることはユーザにとっての影響が大きく不可能である。また，別に用意した実験用のコンピュータにあえて負荷をかけて障害を起こさせたとしても確実に記録が残るとは限らない。なぜならば，ハードウェア障害が起きた場合，その障害に引きずられて電源が同時に落ちてしまうことも多く，このような場合にはオペレーティング・システムがエラーメッセージを出す前に電源がOFFになってしまうため，障害の

瞬間はもちろん，障害の前後の記録も残らないことも多い。このような状況から，ハードディスクの障害に関する十分な量のデータを収集することは容易なことではない。

そこで本研究では，既に公開されているS.M.A.R.T.（Self-Monitoring, Analysis and Reporting Technology）情報を用いて解析を行うことにした。S.M.A.R.T.情報とはハードディスクの状態をハードディスク自身が発行する状態情報のことで，障害の早期発見・故障の予測などを目的としてそれぞれのハードディスクにもともと搭載されている機能である。例えば，システム上に発生した各種エラーの発生頻度や発生したエラーのワースト記録，電源のON/OFFの回数，電源投入時間などあらゆる情報がハードディスク本体に記憶されている。大手のハードディスクベンダーはこれらの情報をもとにさまざまなツールを構築しており，ハードディスク障害診断ツールを公開している場合もある。しかし，これまでの研究により，個々のS.M.A.R.T.情報のパラメータを監視しているだけでは実際の障害予測に対して十分な情報とは言えないことが指摘されている[5]。そこで，本論文ではS.M.A.R.T.情報を集めたS.M.A.R.T.情報群に対して，機械学習をさせることにより，S.M.A.R.T.情報群から障害の予測の有無や障害が起きる時期の予測を行うことを目的として実験を行い，検証を実施したので，その結果を報告する。

２．S.M.A.R.T.情報

２－１．S.M.A.R.T.情報におけるパラメータ

S.M.A.R.T.情報は，ハードウェアにもともと備え付けられているハードウェア診断のためのシステムであり，ハードウェアを多種類の要素から診断し，その項目（パラメータ）ごとに状態を数値で示してくれる機能である。S.M.A.R.T.が提供する情報には，電源投入回数やプログラムエラーなどのシステムの使われ方や状態などの内部情報と，温度などの外部情報も含まれている（表 1 ）。したがってS.M.A.R.T.情報を確認することでハードウェア内部の状態を知ることができる。現在製造販売されているハードウェアには，このS.M.A.R.T.機能が備わっている。表 1 はS.M.A.R.T.情報におけるパラメータの例の一部である。各パラメータはそれぞれに値を持っており，それによって機能の状態を表している。それぞれのパラメータ

表 1　S.M.A.R.T.情報におけるパラメータ例

No	属性	内容
1	Temperature	温度
2	Power-On Hours	通電時間の合計
3	Device Power Cycle Count	電源投入回数
4	Write Error Rate	書き込みエラー率
5	Read Error Rate	読み込みエラー率
6	Erase Fail Count	消去エラー回数
7	Program Fail Count	プログラムのエラー回数
8	Spin Retry Count	スピンアップ再試行の数
9	Unexpected Power Loss	予期しない電力損失
10	Command Timeout	タイムアウトによるコマンドの中止回数
11	Throughput performance	スループット性能
12	Hardware ECC recovered	ハードウェアエラー訂正による回復回数

出所：Acronis HPに基づき筆者作成
https://kb.acronis.com/content/9123（閲覧2019/9/20 閲覧）

の属性には，現在の値，最悪値，しきい値などが表示され，システムの内部の状態が現在どのようになっているのか，またこれまでにどのように使用されてきたかがわかるようになっている。現在値などのそれぞれの表示は以下の意味を有している（図 1 ）。

・現在の値（Current）…大きいほどハードウェアの状態は良い。
・最悪値（Worst）…計測した中で最も悪い状態の値。大きいほどハードウェアの状態が安定している。
・しきい値（Threshold）…メーカーが設定した限界とされる値。現在の値および最悪の値がこの値を越して下回った場合には故障を疑う必要があるとされる。
・データ（Raw value）…リアルタイムデータ。

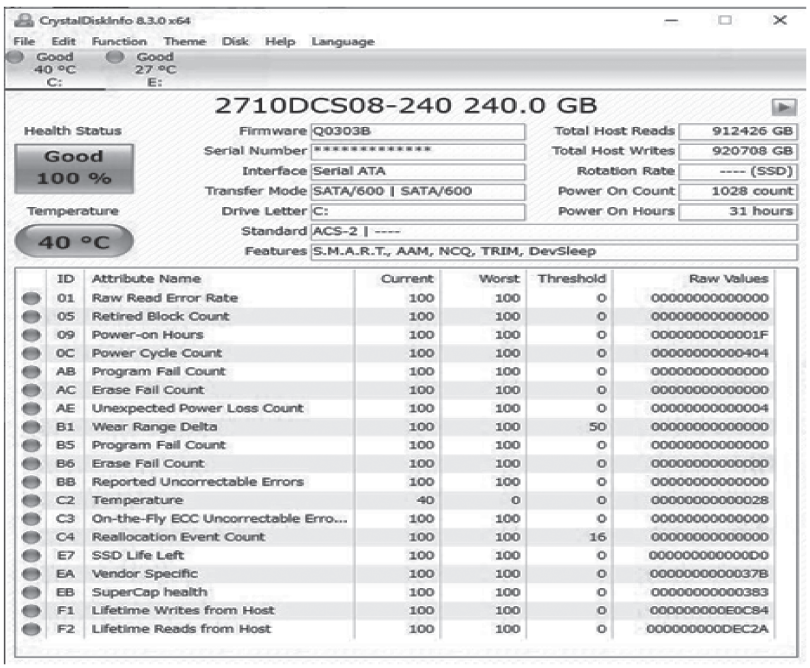
これらの情報をもとにして，ハードウェアベンダーの中には，独自に設けたしきい値を使い，しきい値を超えた場合は危険であると判断してユーザに通知するサービスを提供しているベンダーもある。このように，S.M.A.R.T.情報はハードウェアの内部の様態を知るのに有用な情報であるが，ハードディスクメーカーごとに仕様や記録されている情報内容が異なるため，全体としての比較検証は難しい。一方で，ハードウェアの状態を一覧で表示されるソフトウェアが提供されている [4]。

２－２．S.M.A.R.T.情報を用いた実験について

実験には多数のデータ量が必要なため，実験データはクラウド関連企業が提供している約80,000台のS.M.A.R.T.情報を利用する [2]。これらのデータはクラウド関連企業において実際に使われていた機械で，Web上で一般に公開されている情報である。これらのS.M.A.R.T.情報の中には，壊れたシステムの情報だけでなく壊れなかったシステムの情報も混在している。さらに，多種のモデルのハードウェアが含まれているため，実験前に準備を行なう必要がある。

〈S.M.A.R.T.情報の実験準備〉

１台のハードウェアからは１日１回データが記録されるため，１台のハードウェアに対して多くのサンプルデータが生成される（図 2 ）。



画像：CrystalDiskInfoから提供されているソフトウェアを用いて筆者作成.

図 1　ハードウェアとS.M.A.R.T情報の概要

一台のハードウェアに対して数年分のデータがある。ハードウェアとデータを区別するために，これらのデータをここでは標本データと呼ぶ（図２）。

これらの標本データを使用するために，次の準備を行う。

１）80,000個のデータを，それぞれ型番（モデル）ごとに分類する（図３）。

２）次に各モデルについて，それぞれ故障したハードウェアと故障しなかったハードウェアを２つのグループに分類する。故障したハードウェアの台数は故障しなかったハードウェアの台数と比べてかなり少ないので，グループの大きさは異なるが,それらをFグループとNFグループに分類する（図４）。

FとNFについては以下の意味となる。

F：Failure（壊れたグループ）

NF：Not-Failure（壊れなかったグループ）

３）ここからは各ハードウェアに対応する標本データの分類となる。実験では，機械学習のホールドアウト検証を用いて実験するため，学習のトレーニングと学習後のテスト用データをあらかじめ分割しておく。

各モデルのFグループのハードディスクを80％と20％の割合でランダムに分ける（80％=トレーニング用，20％=テスト用）。ここで，標本データを分類するのは，機械学習の学習に用いるデータと，学習後の検証に使うデータが重ならないようにするためである。

次にFグループの20％のハードディスクから抽出された標本データ数と同数の標本をNFグループからランダムに抽出し，テスト用としてとっておく。両方のグループから同数の標本データを取り出しておくことによって，テストデータ群には，故障と非故障がそれぞれ50％ずつが混在するデータ群となる（図５）。

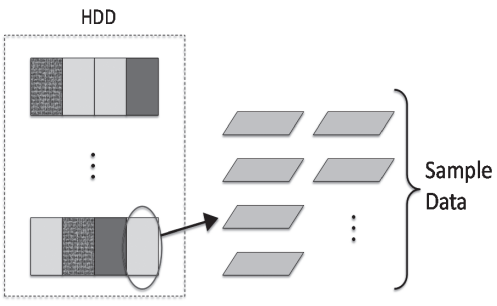


図２　ハードウェアと標本データ

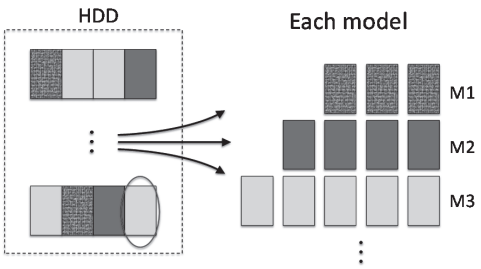


図３　データの分類

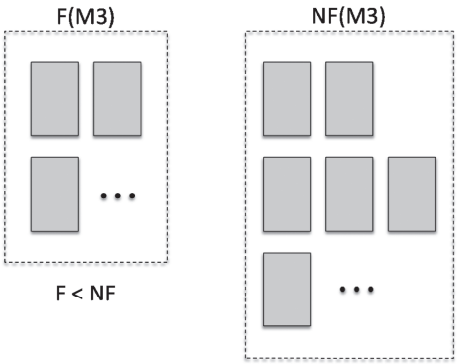


図４　データの分類

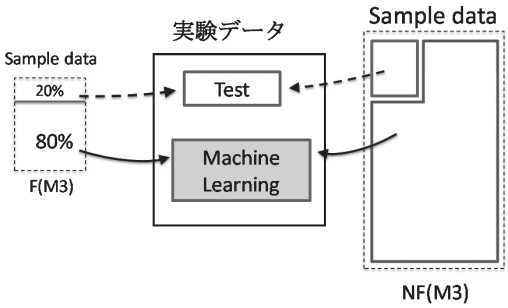


図５　実験データの生成

３．実験のアプローチ

本章では，実験の手法について説明する。

できるだけ精度の高い障害予測を達成するために，S.M.A.R.T.情報単体ではなく，標本データをS.M.A.R.T.情報群として機械学習させ，検証を行った。機械学習にもいくつかの代表的な学習方法が存在するが，前回我々が行った異常

検知に関する予備実験および検証の結果，いくつかある機械学習の中でサポートベクターマシンが良い結果が示したことから，今回の実験ではサポートベクターマシンを使用する[6]。

３－１．サポートベクターマシン

コンピュータに過去のデータを学習させ，将来どのような結果が出るのかを予測させる機械学習を用いた研究が近年盛んに行なわれており，急速に精度も高まってきている。例えば画像による数字認識や顧客分析による顧客予測，写真の顔認証など，身近なところに機械学習は広く活用されている。手法に関してもさまざまな手法が開発されている。その中で，サポートベクターマシンは，パターン認識モデルの一つで，ある集団を分類するための分類問題に適用できる[3]。サポートベクターマシンが実験によく用いられているのは，実験での高認識率，理論的基礎，実現のしやすさという 3 つの要素を兼ね備えているためであると言われている[1)]。パターン認識手法とは，人間によってすでに識別済みの訓練サンプルに基づき，識別関数のパラメータを決定する方法である。サポートベクターマシンは線形識別器の一つで，以下のテストサンプル $x=(x_1, ..., x_d)^T$ の識別関数は，次の式で表される[13]。

$$f(x) = \sum_{j=1}^{d} w_j x_j + b \qquad (1)$$

ここで，w_j は，線形識別器の重みと呼ばれるパラメータで，b はバイアス項と呼ばれる。この識別器の $f(x) = 0$ を満たす点の集合は，$d - 1$ 次元の超平面となる。サポートベクターマシンでは，未知のテストサンプルを正しく識別するために，訓練サンプルの 2 つのクラスの真ん中を通る超平面を探す。すなわち，訓練サンプルと超平面と訓練サンプルとの最小距離を評価関数として用い，これを最大にするように超平面を決定する。そのように評価関数の値を大きくすれば，2 つのクラスへの距離がバランスされ，真ん中に超平面が位置することになる[13]。すなわち，サポートベクターマシンは，このようなマージン最大化という考え方を利用した手法である。そのため，他のアルゴリズムに比べてシンプルでありながら認識性能に優れていることが知られており，また，未学習データに対して高い識別性能を得るため，未知のテストデータに対する予測に対しても高い認識精度が期待できる手法である[11]。

３－２．提案法

セクション 2 で準備した個々のテストデータには，対応するハードウェアのある時点での情報が書き込まれているため，ひとつのハードウェアに対して，時系列的に多数の標本データが存在する。情報としては，温度，ハードウェアの電源のON/OFF回数など多岐にわたるが，言うまでもなく特に重要な情報は故障の有無である。故障の有無が機械学習の重要な教師データとなるため，標本データにステータスが付加される。

各行を 1 標本として扱うため，最終的に故障したハードウェアに対応する標本データは行の数だけ存在する.各行には 1 日のS.M.A.R.T情報とハードウェアステータス（good or failure）が記録される。したがって，各ハードディスクが図 6 のような形式で記述される。

それらの標本データをトレーニングする際には,故障した場合には「１」のステータスが付加され，故障しなかった標本には「０」のステータスが付けられる。学習段階においては，これらのデータが混在した状態で，学習が実施される。

既にステータスが付加されているため，これらの故障のステータスを教師データとすることができる。このようにそれぞれのハードウェアは多数の時系列的な標本データを持っている。

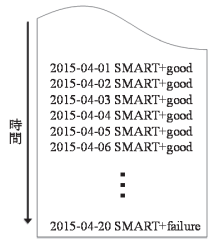


図 6　各ハードウェアのデータ

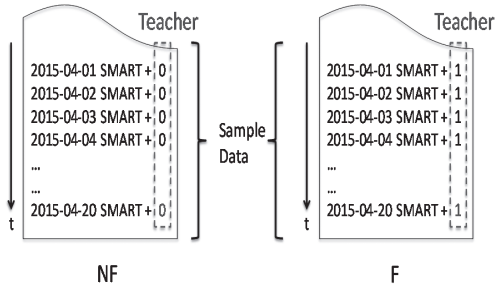


図７　データの教師信号

実際の学習局面ではこれらを分類して学習する（図 ７ ）。

ハードウェアのS.M.A.R.T情報はbackblaze[2)]より取得したものを利用する。

３－３．実験のアルゴリズム

前述のように，アルゴリズムとしてはサポートベクトル法を利用し，プログラミング言語としてはR（version 3.31）を用いた。実験は，２つのフェーズから成っている。まずは標本の学習段階で，次がテスト段階である。

（1） 学習段階：まず，学習をする前に，学習のための標本データの準備を行う必要があるので，この方法について説明する。準備では，ハードウェアごとに標本データを分類し，ハードウェアの壊れた日を起点として，過去に遡った日数で区切って学習をさせる。

ハードウェアは使い始めてから壊れるまでの期間，膨大な内部データを生成している。故障時期の予測を実施するため，これらのデータを一定期間で区切ってグループ化する。例えば，12月31日に壊れたハードウェアの場合，12月31日を０地点とし，それより前のデータを30日ごとのグループにする。壊れる60日前から31日前までのデータを標本ボックスに入れる。この作業を全てのハードウェアに対して行い，故障した日時から遡った日数が同じものを集めてひとつの標本データグループを作る。壊れていないハードウェアについては起点となる日にちがないため，ランダムに日程を抽出して標本データとして取り込んでいく（図 ８ ）。

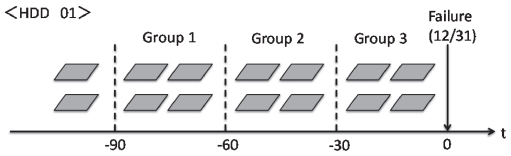


図８　データのグループ化の流れ

表２　判定器の生成条件

標本取得条件	設定値
データ取得開始日	障害の前日
データ取得終了日	ケース：障害から480日前

この操作を繰り返し，30日ごとに480日まで16個の学習用データ群を作成する。

データ群の準備ができたところで，これらのデータを学習させることによって，16個の独立した障害の判定器を生成する。

（2） テスト段階：これらの判定器を用いてどの程度予測ができるか別のデータを用いて検証をする。

テスト用データも学習用データと同じように故障の日から遡って30日ごとで分ける。トレーニングにおいては，30日ごとのデータを用いて，16個の判別器を生成したが，テストに関しては実験の際の計算コストがかかりすぎるため，本論文では次の４つのパターンのみで実験を行った。すなわち以下の４パターンである。

１）１日－30日
２）31日－60日
３）61日－90日
４）150日－180日

グループ化のところで前述したとおり，壊れた日にちまでの日数をもとに，データを標本ボックスに入れる。この作業を全てのハードウェアに対して行い，故障した日時から遡った日数が同じものを集めて標本

データグループを作る。したがって，壊れるまでの日数が同一期間に入るデータは，同一グループとして扱うことにする（図８）。

３－4. モデル別分類

実験には，black blaze社が公開しているある国内メーカーの３種類のハードウェア（同一メーカー）のデータを用いる。それぞれのモデル別に機械学習を行って，予測精度の調査を行なった。

【実験に使用したモデル】

（１）モデル１：M１
（２）モデル２：M２
（３）モデル３：M３

トレーニングデータ用ハードウェアと試験データ用ハードウェアは完全に分けており，試験データ用ハードウェアのS.M.A.R.T情報から障害が起きるハードウェアか，障害の起きないハードウェアなのかを予測し，その評価を行なった。前述のとおり，機械学習トレーニングの段階で，トレーニングするデータセットを30日ごとに区分けしてトレーニングをする。すなわち，障害発生の１日前から障害発生の30日前までのデータのみでトレーニングした機械，障害発生の31日前から障害発生の60日前までのデータのみでトレーニングした機械，というように30日ごとに区切ったデータでトレーニングをした機械でテストを行った。これに合わせてテスト用データも30日ごとに区切ってテストをおこなった。テスト用データをこのように分類して検証した理由は，大きく２つある。ひとつめは，故障する日にちが近づいてくると，より検出力が高くなるか否かを確認したいと考えたこと。２つ目は故障の時期と学習用標本データの関連性について検証したいという考えからである。

４．実験および評価指標

トレーニングによってあらかじめ生成した合計16個の判定器に対して，トレーニングには使用されなかった試験用の標本データを入力し，それが壊れるハードウェアか否かの判定をSVMアルゴリズムで行った。ハードウェアのモデルは，S.M.A.R.T.情報の関係から４種類を対象に実験した。１つのモデルには，多数の異なるシリアル番号のハードウェアが属するが，分けずにそのまま扱う（図９）。

データの期間は2012年１月１日から480日分のデータを使用する。

実験結果は表３のような行列形式で表され，行列のそれぞれの要素には個数が記録される。

例えばAであれば,故障しないと予測されたハードウェアのうち実際に故障しなかった台数

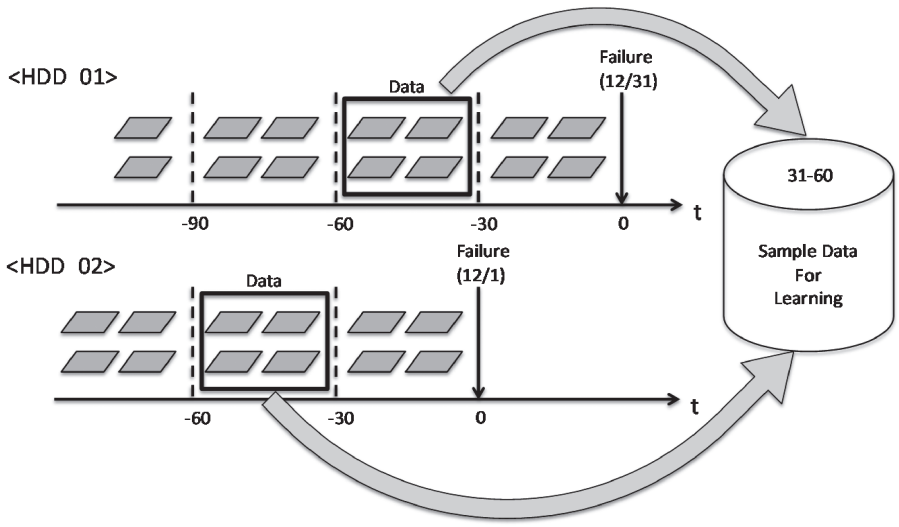


図９　トレーニングデータの作成方法

が示される.本論文においては，評価基準として一般的に用いられている基準に注目をしてデータを取得した。これらのハードウェアからコードでトレーニングデータ，および試験データが生成される。モデル別ハードウェア数は表４に示す。表３の行列Bは「壊れると予測していたハードウェアが実際には壊れなかった」ということを意味しており，行列Dは，「壊れると予測していたハードウェアが壊れてしまったことを意味している。

基準として用いたのはPRE（precision）と呼ばれる値である。PREは，壊れると予測したものの中で，実際にはどの程度の台数が壊れたかを示す数値である。この値が高ければ，壊れる予測をしたハードウェアについて，高い精度で実際に壊れたということになる。

【評価指標】

PRE（precision）：精度

D/（B+D）　（2）

表３　実験の評価指標

	予測（壊れない）	予測（壊れる）
実際（壊れてない）	A（真陰性）	B（偽陽性）
実際（壊れた）	C（偽陰性）	D（真陽性）

表４　データ取得のハードウェア台数

モデル	障害ハードウェア	正常ハードウェア
M１	113	4519
M２	58	2625
M３	61	975

４－１．実験結果

実験を行った結果を図10～図23に示す。実験は３種類のハードウェアの型番（モデル）について実施した。

M１～M３はモデル１～モデル３を意味する。方法は，故障発生までの期間によって16種類に区分けされたテストデータをそれぞれに対して16種類の判別器を用いて，故障するか否かを予測する形で実施し，予測がどの程度当たっているかを百分率で評価したものである。縦軸は予測の正解率で横軸はそれぞれ16種類の判別器である。例えば,図10のグラフは，モデル１の４つの日程パターンのうち，30日以内に壊れたハードウェアをテストデータとして，16種類の判別機による判別を行い，精度をプロットしたグラフである。このようにして各モデルに対し期間ごとの４パターンで実験をおこなった。

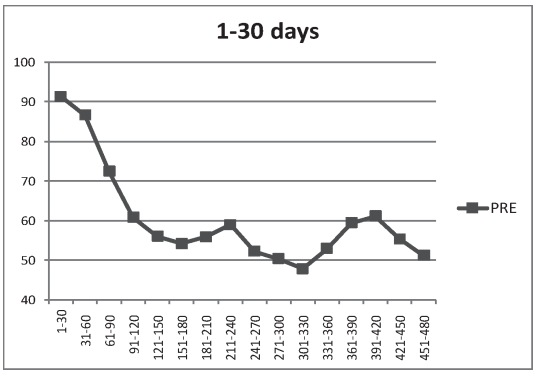


図10　PRE of M1（１－30 days）

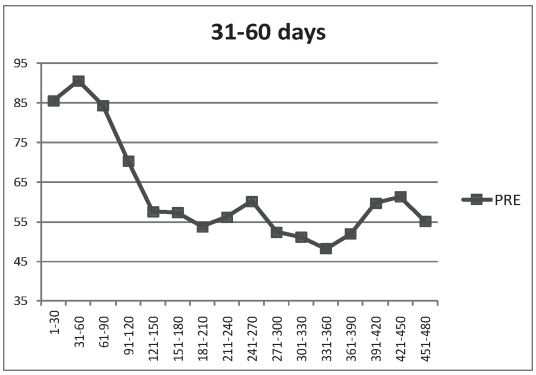


図11　PRE of M1（31－60 days）

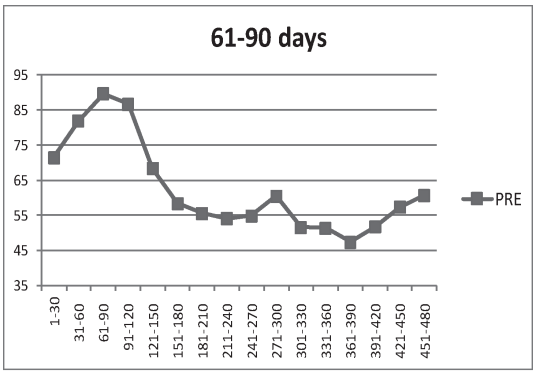


図12　PRE of M1（61－90 days）

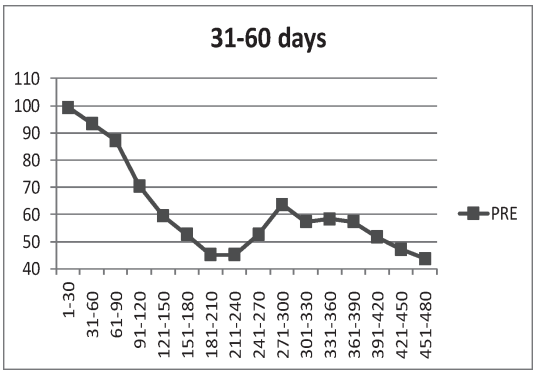


図15　PRE of M2（31－60 days）

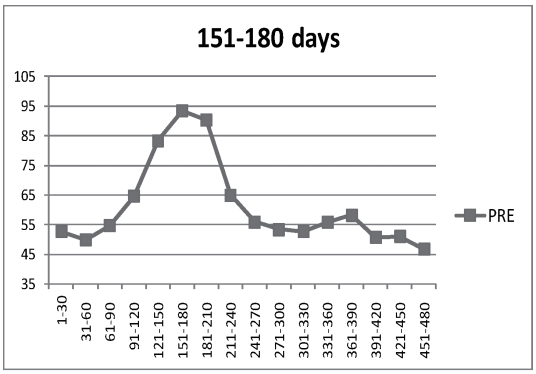


図13　PRE of M1（151－180 days）

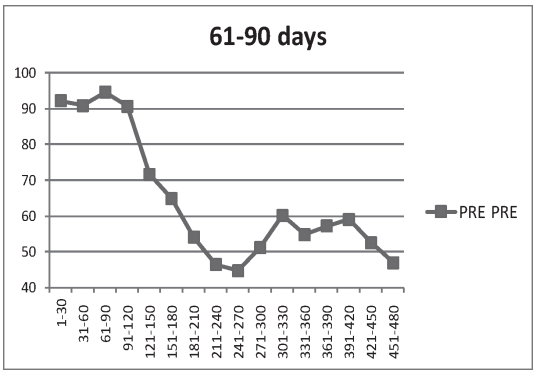


図16　PRE of M2（61-90 days）

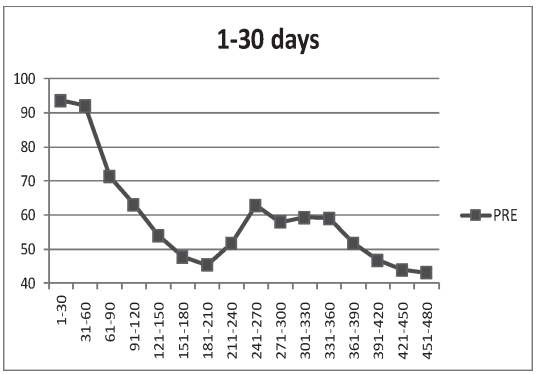


図14　PRE of M2（1－30 days）

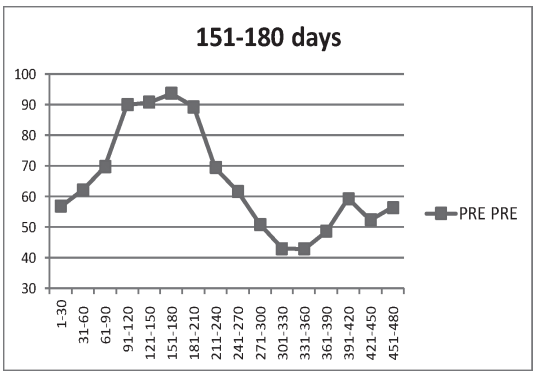


図17　PRE of M2（151－180 days）

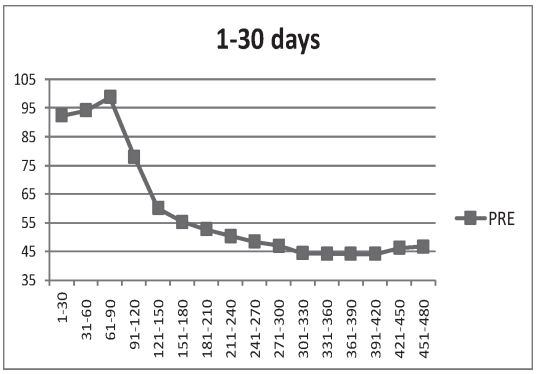


図18　PRE of M3（1－31 days）

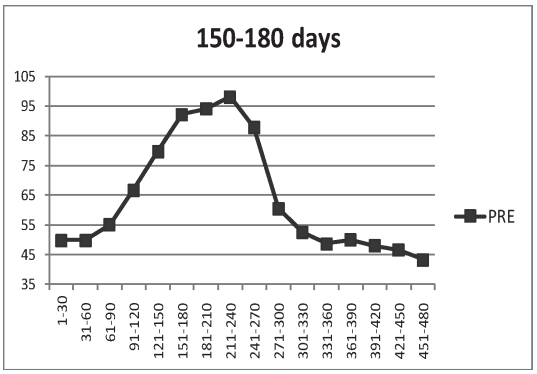


図21　PRE of M3（151－180 days）

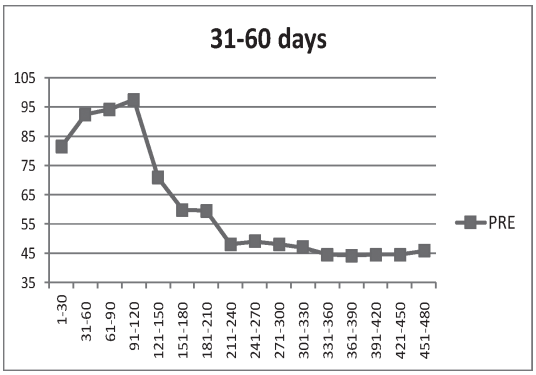


図19　PRE of M3（31－60 days）

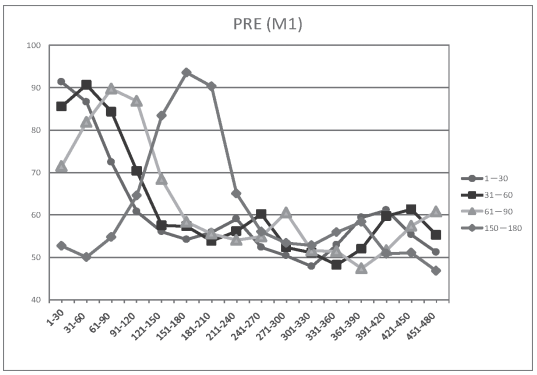


図22　PRE of M1

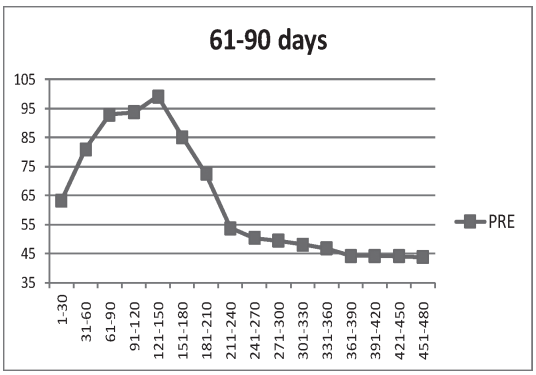


図20　PRE of M3（61－90 days）

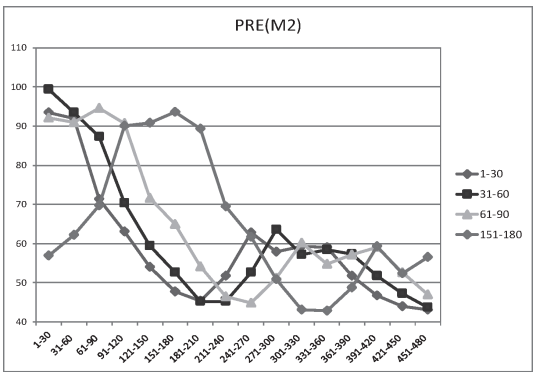


図23　PRE（M2）

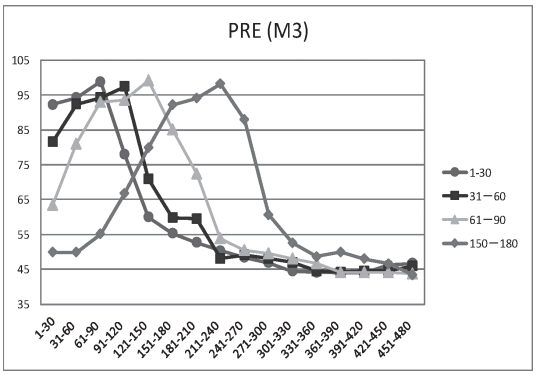


図24　PRE（M3）

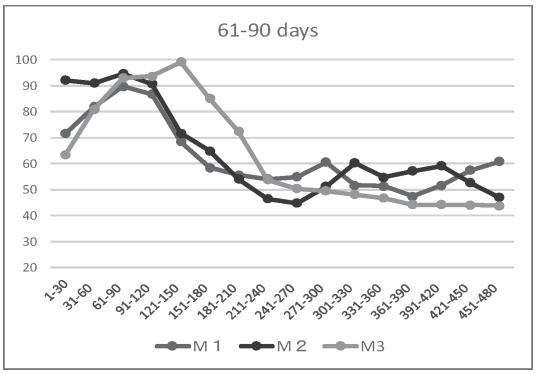


図27　PRE（61－90 days）

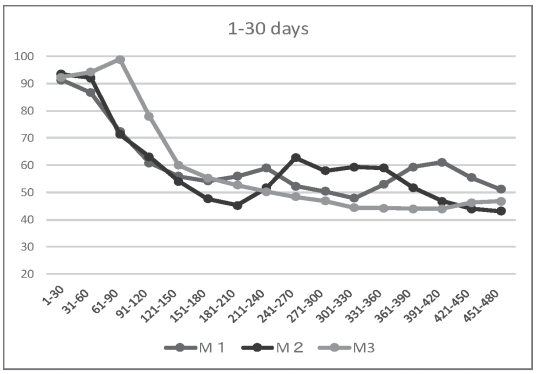


図25　PRE（1－30 days）

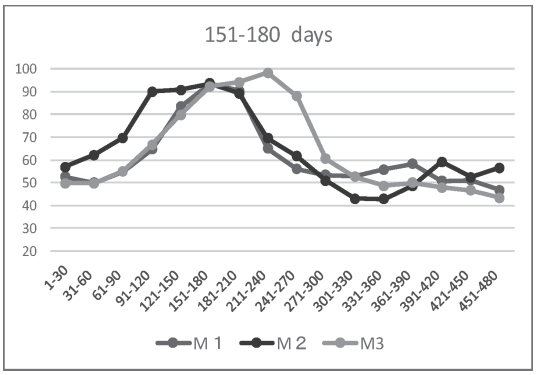


図28　PRE（151－180 days）

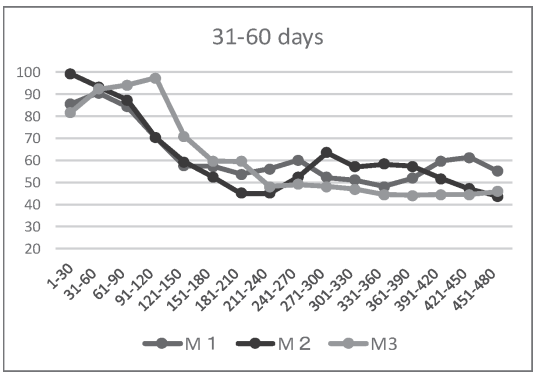


図26　PRE（31－60 days）

4－2. 実験結果の考察

図10～図21は各モデルのPREをプロットしたグラフである。前述のとおり，PREは壊れると予測した台数に対して，実際にどの程度の台数が壊れたかという割合を示す数値である。図22～図24は 4 タイプの日数を比較したグラフで，図25～図28は 3 種類のハードディスクモデルを比較したグラフである。

図10～図21を見ると，PREの値はかなり大きく変動しているが，全体的に故障に対して近い日数の判別機を用いたところの精度が高い傾向がみられる。すなわち，故障するまでの時間が短い場合の精度が高く，故障までの時間が延びるにつれて精度が下がる傾向にある。これは故障が近くなれば，ハードウェア上に何らかの故障の予兆が出て，それがS.M.A.R.T.情報とし

て記録され，予兆を捉えやすくなった結果であると考えられる。逆に，壊れるまでに時間がかかった（当分壊れなかった）学習データを用いた場合には判別精度が低くなってしまうことが示されている。さらに詳しく検証すると，テストデータの日数と判別器の関係性が見えてくる。

図22～図24を見ると，判別器に呼応してPREの精度が変化している傾向が見られる。つまり，テストデータの日数によって，判別器による判別精度のピークが移動していく様子がわかる。すなわち，学習データと故障までの期間という観点から見ると，テストデータのグループの日数と近い期間の判定機を用いると判定精度が高くなるということがわかる。例えば，壊れる前30日以内のハードウェアに対しては，1 − 30 days，31 − 60 days，61 − 90 days付近の判定器を用いた判定精度が高く，151 − 180日以内に壊れるハードウェアに対しては,151 − 180 days，181 − 210 days，211 − 240 days付近といった比較的長い期間の判定器を用いた場合に，PREの判定精度が高くなっていることがわかる。

この傾向は，モデル別に比較しても同様の傾向が見られる。

図25～図28は，日数ごとにＭ１ ～ Ｍ３ のモデルを比較した図である。これらの図を見ても，モデルごとにばらつきは見られるものの，判別精度は学習データの判別器に応じて変化していることが示されている。

以上のことから，壊れる日が近ければ近いほど，PREに影響を与える特有の兆候が表れると考えられると同時に，学習に用いたデータの中に，壊れるまでの期間に応じて，その期間特有の傾向が見られるのではないかと考えられる。したがって，データに応じて学習データを適切に使い分けることで，全体として，より判別精度の高いシステムの構築が期待できることが改めて確認された。

５．おわりに

本稿では，障害予測を目的として，S.M.A.R.T.情報のデータ群をもちいる手法を提案した。具体的には，故障の有無を教師信号としてS.M.A.R.T.情報のデータ群を機械学習させることによって，障害予測をさせるための実験を行なった。その結果，比較的高い精度で障害検知が可能となることが明らかとなった。また，故障するまでの期間ごとの判別器を用いることで，予測の精度が変化することが明らかになった。

今回得られた知見をもとに，各期間における特有の兆候についての知見を深めるとともに，PREを向上させる方法についての研究を行なうことが今後の課題である。

【参考文献】

［1］AOS Technologies, Inc（ハードウェア障害ツール）（2018/09/10）

［2］backblaze,“https://www.backblaze.com/b2/hard-drive-test-data.html（2018/09/10）

［3］N.Cristianini, J. Shawe-Taylor（2000）“An introduction to support Vector Machines: and other kernel-based learning methods”, Cambridge University Press New York, NY, USA.

［4］CrystalDiskInfo, https://crystalmark.info/ja/（2018/09/10）

［5］P.Eduardo, Wolf-Dietrich Weber, and Luiz André Barroso. “Failure Trends in a Large Disk Drive Population.” FAST. Vol. 7. 2007.

［6］S.Felix et al.（2007）“Using Hidden Semi-Markov Models for Effective Online Failure Prediction”, Proceedings of 26th IEEE International Symposium on Reliable Distributed Systems, pp.161-174.

［7］國分俊介, 片山吉章, 樋口毅 他（2008）“仮想化環境におけるハードウェア障害模擬とHAクラスタシステム試験への適用”,電子情報通信学会技術研究報告. DC, ディペンダブルコンピューティング：IEICE technical report 108（181）, 1-7.

［8］國吉賢吾, 森井昌克（2009）“端末監視によるホームネットワーク異常検知システム”, 情報処理学会研究報告,（17）, 39-46.

［9］織田英夫　北村士守　安保進（1996）“ハードウェア障害情報収集分析システムの概要”, 情報処理学会　全国大会講演論文集（システム）, 385-386。

［10］及川 達也, 和泉 勇治, 加藤 寧 他（2002）“統

計的クラスタリング手法によるネットワーク異常状態の検出”, 電子情報通信学会技術研究報告 : 信学技報 102（349）2002.10.1 p.83～88
［11］C. A. B.Scholkopf（1996）, “Advances in Kernel Methods - Support Vector Learning”, The MIT Press.
［12］鈴木与範,（2005）“Sustainable Service の実現構想”, 情報処理学会研究報告. OS,［システムソフトウェアとオペレーティング・システム］99, 9-14, 2005-06-22
［13］K. TSUDA（2000）“Overview of Support Vector Machine”,電子情報通信学会誌　83, 460-466, 2000.
［14］Y. Watanabe et al.（2011）“Online failure prediction in cloud datacenters by real-time message pattern learning”, Proceeding of CloudCom, IEEE 4th International Conference pp.504 –511.
［15］L.Yu, Z. Zeng, Zhiling Lan, and Susan Coghlan,（2011）“Practical online failure prediction for Blue Gene/F: Period-based vs eventdriven“,Proceedings of IEEE IFIP 41st International Conference on Dependable Systems and Networks Workshops. pp. 259-264。

【注】

1）K. TSUDA “Overview of Support Vector Machine”, 電子情報通信学会誌 vol.83, p460.2000

2）2013年よりデータセンターのハードドライブに関する統計情報などを毎日アップデートして公開している。これにより，通常では収集することができない多数のハードディスクドライブのデータを閲覧したり，独自に評価したりすることができる。これらのデータは,多くの業界やベンダーなどにとっても貴重な公開情報として捉えられている。
blackblaze,“https://www.backblaze.com/b2/hard-drive-test-data.html