Curran and King, J Comput Eng Inf Technol 2019, 8:1
DOI: 10.4172/2324-9307.1000215

Journal of Computer Engineering & Information Technology

Research Article

A SCITECHNOL JOURNAL

Predictive Maintenance for Vibration-Related Failures in the Semi-Conductor Industry

Kevin Curran[1]* and Robert King[2]

Abstract

Predictive maintenance has proven a cost-effective maintenance management method for critical equipment in many verticals. The semi-conductor industry could also benefit. Most semiconductor fabrication plants are equipped with extensive diagnostic and quality control sensors that could be used to monitor the condition of assets and ultimately mitigate unscheduled downtime by identifying root causes of mechanical problems early before they can develop into mechanical failures. Machine Learning is the process of building a scientific model after discovering knowledge from a data set. It is the complex computation process of automatic pattern recognition and intelligent decision making based on training sample data. Machine learning algorithm can gather facts about a situation through sensors or human input and compare this information to stored data and decide what the information signifies. We present here the results of applying machine learning to a predictive maintenance dataset to identify future vibration-related failures. The results of predicted future failures act as an aid for engineers in their decision-making process regarding asset maintenance.

Keywords

Predictive maintenance; Machine learning; Semi-conductor

Introduction

The semiconductor industry includes companies who have been at the forefront of data analytics. Despite this however, very few semiconductor manufacturers have directly applied data analytics to their fab operations [1]. The manufacturing of electronic chips and more specifically wafers is a highly complex operation that can involve hundreds of individual industrial and quality control processes, which can take months of intensive processing from start to finish. Improving yield results is a commitment each manufacturer in the semiconductor industry seeks to fulfil. As faulty equipment can lead to overexposure or underexposure for specific processes, which can ultimately result in undesirable wafers that need to be recycled and for the semiconductor material to start its life as a wafer once more. Sub-components can also be the cause of faults within the fab, as some of the most complex tools used within wafer fabrication can comprise of more than 50,000 parts, acquired from numerous different suppliers [2].

It is important then that underlying mechanical problems are rooted out before the mechanical failure occurs. Especially when manufacturers within the semiconductor industry face some of the fiercest commercial market share competition in the economy. Therefore, it is important that these manufacturers can reliably deliver adequate product quantity and quality, to ensure that their prices remain low and those they can maintain or attain market share [3]. This signifies the importance of mitigating and eliminating the probability of current and future equipment failures within the semiconductor industry, with the objective of minimising unscheduled downtime as much as possible. The predominant maintenance strategy within the semiconductor industry is preventative maintenance, through time-based or a variation of wafer-based maintenance activities. Preventative maintenance however has been proven to be less cost effective and less reliable as a maintenance management method compared to predictive maintenance.

Industry 4.0, also known as the "Industrial internet of things" or "smart manufacturing", refers to the latest technological advancements in industrial production, and the overall transition into the newest industrial revolution known as the "fourth industrial revolution" (Table 1). The German government first coined the term, when an initiative named "Industrie 4.0" was announced in 2011, by an association of representatives gathered from Germany's business, political and science sectors [4]. The aim of the association was to strengthen the competitiveness of Germany's manufacturing industry. Although Germany still leads the charge today, companies around the world have been contributing to deliver the platform of industry 4.0.

Yet some professional and academic experts claim that the digitalisation of production is simply a continuation of the third industrial revolution, others argue that there are distinctive differences between Industry 4.0 and the other industrial revolutions. Those differences being that technological developments are growing exponentially, compared to the linear growth of technology in previous industrial revolutions. It is speculated that this is due to the exceedingly interconnected world, through technology such as the Internet of Things, and that through the introduction of new technology, newer and ever more efficient technology can be developed [5]. Industry 4.0 has conceived the concept of "Smart factories", which refers to the combination of Operational Technology (OT) with Information Technology (IT). It also seeks to build upon the computerization of the third industrial revolution through the importance of digitisation, by introducing Cyber-Physical Systems (CPS) and the Industrial Internet of Things (IIoT) to traditional production lines. CPS can be a collection of sensors, machinery or IT systems that can communicate with other CPS by using standard Internet-based protocols brought by the Internet of Things [6]. While the IIOT refers to the utilisation of the Internet of Things, and specifically the integration of big data and machine learning technology in "Smart factory" manufacturing. Which has introduced the ability to monitor the condition of individual machines, and ultimately has stemmed the functionality of predictive maintenance programmes.

Figure 1 shows the results of a PricewaterhouseCoopers survey regarding Industry 4.0 and whether firms were currently capable of making data driven decisions [7]. Approximately half of the firms engaged in data driven decision-making in 2016. However, by 2020, the expected participation in data driven decision-making doubles for almost every specified industry. So clearly there is a desire to engage in

***Corresponding author:** Kevin Curran, School of Computing, Engineering and Intelligent Systems, Magee Campus, Ulster University, Coleraine, Northern Ireland, Tel: +44 780 3759052; E-mail: kj.curran@ulster.ac.uk

Received: March 20, 2019 **Accepted:** April 04, 2019 **Published:** April 11, 2019

Table 1: Summary of industrial revolutions.

Industrial revolutions: 1.0 - 4.0		
1.0	1784	The transition from craft production to the use of machinery through industrialization, following technological advancements in water and steam power.
2.0	1870	The popularization of mass production, following the division of labour and the use of production lines by employing electrical energy.
3.0	1969	Utilising computerised systems & information technology to automate mass production.
4.0	Today	The digitisation of production through the implementation of Cyber-Physical Systems, brought by the Internet of Things.

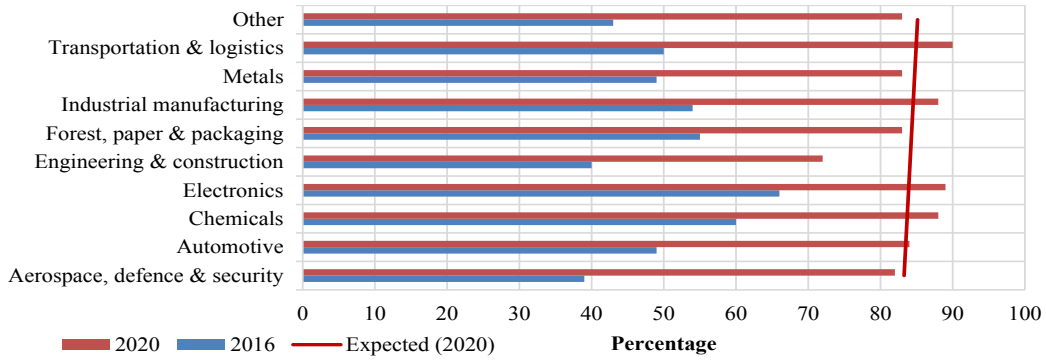


Figure 1: Expected data analytics participation by industry.

data analytics for those firms who do not already engage in data driven decision-making. Notably, the firms that expressed interest and those that already make data driven decisions, all come from industries that involve specialist processes and equipment that would highly benefit from utilising a predicted maintenance programme for their day-to-day operations. So as manufacturers invest in new technology to meet industry 4.0 standards, predictive maintenance stands as one of the most critical functions for a 'Smart factory' to be considered 'smart'. As each manufacturer relies on specialist equipment used for processes unique to their product or industry, and for that equipment to operate at a high efficiency. Most importantly however, is for their specialist equipment to function uninterrupted of technical problems, so that their equipment can attain the highest uptime possible.

Predictive maintenance aims to reduce downtime of assets which it achieves through performing asset maintenance when it is required and to prevent unscheduled downtime due to asset failure. It also aims to monitor the performance and "health" of assets and pinpoint and eliminate the root cause of asset performance degradation. These objectives are achievable through using sensor data generated from monitored assets, to produce predictive models that can aid in the decision-making process for asset maintenance.

Methodology

This paper discusses the application of machine learning in a predictive maintenance solution to assist in the decision-making process of maintenance management to predict future vibration-related failures. The solution calculates the Time to Failure (TTF) for equipment, by using variable data generated by equipment sensors to predict future equipment failures in the form of a predictive model by using the TTF variable.

Predictive maintenance

Predictive Maintenance (PdM) is a proactive maintenance approach that emphasises the forecast of how and when equipment will fail through data analytics, and to perform maintenance precisely before the total failure occurs. This is achieved through the detection of possible failures by monitoring and analysing various equipment operation variables, by using an assortment of diagnostic sensors and other monitoring instruments. For example, monitoring equipment changes in; vibration, temperature, pressure or voltage, to name a few. The outcome then, is that maintenance will only be scheduled when a failure has been detected, rather than when equipment is perceived to require maintenance. A good PdM programme should set out to improve production efficiency through decreased equipment downtime, and maintenance effectiveness by eliminating unneeded maintenance. The result is lower overall costs involved in maintenance management while retaining high equipment uptime, since equipment wear can be analysed during production operations by using sensors. PdM has both high implementation and operating costs in addition to requiring a high level of skill to be able to interpret and act upon received sensor data.

Maintenance strategies

Preventive Maintenance (PM) or "Preventative maintenance" is the maintenance philosophy of performing maintenance tasks at predetermined intervals using triggers. These triggers can be derived by a specific amount of calendar days or when a tool has elapsed a defined amount of runtime [8]. Ideally, the expected result is a reduced likeliness for equipment failure to occur, as equipment should receive needed maintenance through these maintenance intervals before mechanical failures or malfunctions can occur. However, the most apparent issue in utilising PM is that, although failure due to equipment "wear" is considerably reduced, it is notably inefficient in terms of asset management, and can often result in unnecessary maintenance being performed, in what could be considered "operational insurance". This is because maintenance schedules are determined by the "Mean-time-to-failure" (MTTF) statistic, which refers to the average lifespan or duration a specific tool is expected to operate, uninterrupted of technical issues, until the duration ends, and a failure eventually occurs. The MTTF can be derived from a manufacturer's recommendation, the use of industry statistics or from the analysis of previous maintenance history from an asset within the same classification. Despite this, PM offers little

assurance for catastrophic failures and the need of "unscheduled" maintenance, and they are still just as likely to occur when adopting this maintenance strategy.

Run-to-failure Maintenance (RTF), also known as "Breakdown" or "Reactive" maintenance, is the most straightforward maintenance philosophy. When a machine failure has occurred, priority is placed on restoring that tool back to its original state of operation. The maintenance strategy embodies the saying; "If it ain't broke, don't fix it" and as you might expect from the name, no emphasis is placed on any proactive maintenance management. The RTF ideology is about maximising the use of resources in production, rather than increasing the usability of resources through improved lifespan of assets. This view on maintenance management anticipates that the repair cost and lost operation time involved in equipment failure is lower in value, than the cost required to implement and maintain any ongoing maintenance management network. The biggest flaw in this approach is that it is unpredictable, and its success is highly dependent on the type of machinery used with this approach.

Reliability-Centered Maintenance (RCM) is a hybrid maintenance philosophy, combining the use of PdM, PM and RTF. RCM acknowledges that not all equipment has the same level of importance and takes a systematic approach to identify the right strategy, for the right need. Despite this, the predominant strategy is PdM, which is applied to the most critical systems of operation, while RTF is utilised for the least.

Failure rate

To be able to adequately assess each maintenance strategies effectiveness, it is important to understand why failures occur. The bath-tub curve depicts the failure rate for most mechanical equipment and presents the three different stages of reliability throughout a machines lifespan (see Error! Reference source not found.). The "Break-in" phase, illustrates high failure rate expectancy for new equipment such as defective components and poor instalments (Figure 2) [9].

Over the course of a few weeks or months, machine reliability will be proven, and the tool will enter the "Useful life" phase, where it will spend most of its lifetime. During this stage equipment will maintain a constant probability of failure. As the tool operates uninterrupted from physical human involvement or worn out parts. Finally, the machine will unavoidably reach the "Wear-out" phase, representing an ever-increasing escalation of failure, as the equipment ages and wear becomes much more detrimental through extended use, until the failure inevitably occurs. This shows that predictive maintenance is most effective when deployed during the break-in and wear-out phase, where failure indication is strong. As the constant failure rate of equipment during the useful life phase is random in nature, and predictions made within the phase are less reliable. The objective of proactive and planned maintenance is to prolong the lifespan of equipment once it has reached the wear-out phase therefore, maximising the effective use of said equipment. But each maintenance programme has its own impact on the bath-tub curve concept, as shown in Error! Reference source not found.

The lifespan of equipment is prolonged and renewed through multiple maintenance iterations or "spikes". PdM portrays short iteration cycles representing the early detection and resolution of technical issues before they arise into mechanical failures through using data analytics and machine learning. While PM also prolongs the lifespan of assets, most notably, the maintenance cycles are considerably greater as maintenance is scheduled at predetermined intervals, thus unsuspected problems may be in circulation longer, resulting in a greater failure rate risk. There is no prolonging effect when utilising an RTF strategy as no priority is given to maintenance methods until the failure has occurred, therefore the failure rate gradually increases until the failure occurs.

Machine Learning

Machine learning algorithms can be categorised into three distinctive learning model types; supervised learning, unsupervised learning or reinforcement learning. The use case of each type of learning model is highly dependent on the input used, and the expected output to be generated by the learning algorithm. Supervised learning is by far the most commonly used type of machine learning algorithm. The premise of supervised learning is; by training the machine learning algorithm to map the correct input to the correct output using historical data, the model in theory should be able to map previously unseen data to a predicted output value as well. The desired situation is that through training the predictive model, the machine learning algorithm should be able to predict the value Y, by using never seen before input data X. Supervised learning earns its name through using training data, which is used to improve the algorithms mapping accuracy, and test data, which is used to score the accuracy achieved by the machine learning algorithm. Meaning that to achieve the full potential of supervised learning, the desired output must already be known, and the dataset to be used with the supervised learning algorithm must be fully completed and labelled as well. Unsupervised learning refers to machine learning algorithms that are used to observe and extrapolate the context of unlabelled input data from highly data driven datasets. Unlabelled data is data that has no defined category, other than the value itself. A person can understand a list of unlabelled temperature values, even when not given the context. But a machine is only able to understand the data's raw value; therefore it is the job of the unsupervised learning algorithm to deduce the context of the value without being explicitly told. Unsupervised learning algorithms achieve this through a process called Clustering, which attempts to find underlying patterns within the dataset, so that similarly valued data can be grouped together to identify data similarities. With the end goal of mapping the unlabelled data into labelled outputs. The main distinctive difference between unsupervised learning and the other learning models is that there is no specified assessment period used to verify the results of the unsupervised learning algorithm (Figure 1). This is because unsupervised learning algorithms are specifically used when a dataset contains unlabelled data. This means that there is no need to verify the prediction results or accuracy of the algorithm, as there is no test data to use in the first place. Therefore, observations made by the unsupervised learning algorithm can be best summarised as logical assumptions, rather than direct estimates that can be proven

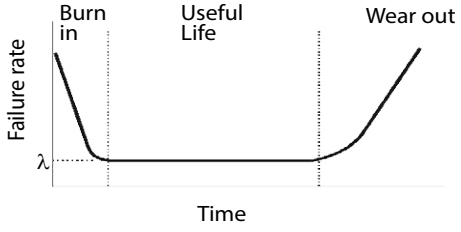


Figure 2: Effect of maintenance strategies on bath-tub curve.

with specific test data. The main applications of unsupervised learning algorithms are predominantly pattern recognition-based. This includes; object and face recognition, voting analysis, image segmentation and more recently network security for behaviour-based detection [10].

Reinforcement learning is used when the result of the machine learning algorithm is an action, rather than a prediction. Or alternatively, when emphasis is placed on the behaviour of the algorithm within a specific environment. The machine learning algorithm takes form as an agent, which is used to observe and learn how to map scenarios (input) into actions (output). Reinforcement learning is very similar to supervised learning, but the key differentiator between the two learning models is that, while supervised learning uses specified test and training data to verify the predictions made by the machine learning algorithm, Reinforcement learning employs a reward function that gives feedback to the algorithm (agent). A common and simplified example of reinforcement learning is the act of a child learning how to walk. In this scenario, the algorithm takes form as a child (agent) and the environment would be the ground. The algorithm observes the state of the environment – e.g. a child is taught how to walk by their parents. The algorithm then manipulates the environment and maps it into an action – e.g. the child tries to walk alone or not at all. The environment then provides feedback through a positive or negative reward based on the completion criteria. For example, if a child can walk alone, their parent rewards them. While if the child does not attempt to walk, no reward is given. It is trial, and error based. Another notable difference between reinforcement learning and supervised learning is the use of a 'supervisor'. In a supervised learning model, test data is set aside with the specific purpose of verifying the results of the algorithms prediction. This test data acts as a supervisor in this scenario, as it is given knowledge of the environment and it shares that knowledge with the machine learning algorithm to verify the predictions made by the algorithm. In contrast, reinforcement learning is designed to overcome problems that have too many resolutions. In this type of environment, creating a supervisor to verify the results with every resolution in mind is impossible. Therefore, reinforcement learning algorithms rely on learning from past experiences using the feedback given from the environment.

The learning algorithm type best suited for a predictive maintenance prototype is a supervised learning model as the predictive maintenance dataset is fully labelled and largely complete and emphasis should be placed on the reliability of predictions made by the learning algorithm, as unreliable maintenance will likely result in unnecessary maintenance being performed, which contradicts the value of predictive maintenance.

Supervised learning algorithms

With supervised learning as the selected learning model, consideration must now be given to the available supervised learning algorithms that will be used to map the predictive maintenance input and output. But first, supervised learning algorithms are prone to two specific modelling errors called 'Overfitting' or 'Underfitting' a predictive model. Furthermore, supervised learning algorithms also suffer a modelling problem called the 'Bias-variance trade-off'. These two specific circumstances need to be discussed, in the future event that either of these modelling errors occurs in the development of the predictive maintenance model.

The bias-variance trade off, sometimes known as the bias-variance dilemma, is a specific modelling problem for supervised learning algorithms. The difficulty of the problem is through the act of simultaneously mitigating the effects of errors caused from high model bias and high model variance. Predictive models that can be described as having high variance, low bias, are referred to as 'Overfit' models. Overfitting will be the topic of discussion in the next subsection, but the key drawback of an overfit model is that the model cannot distinguish random "noise" from relevant data within the specified training set. Likewise, models described as low variance, high bias are referred to as 'Underfit' models. Underfitting will also be discussed in the next subsection, but the defective trait of an underfit model is that the predictive model makes too many assumptions, due to the high level of bias present within the model. This results in a predictive model failing to identify relevant correlation between dataset features and intended model outputs. The total error caused by high bias or high variance is the same. Only when the variance and bias are near equal in influence, is the expected total error for the predictive model reduced. Overfitting also known as "Overtraining", describes a model that corresponds to the training data too well, and therefore fails to generalise to new data, to create reliable predictions. Overfitting is the result of a predictive model with high variance, which causes the algorithm to react to "random noise" in the training data, therefore negatively impacting the "predictive" effect of the predictive model. Regression models are particularly prone to overfitting, and in extreme situations the 'trend line' will unrealistically plot through each plotted point. In comparison, underfitting also known as "Undertraining" is the scenario when a model can be described as having too few variables, which can be a by-product of using an incomplete dataset [11]. This will result in a predictive model that performs poorly and ultimately fails to predict reliable future observations. Underfitting occurs when a predictive model has a high bias that causes the algorithm to make too many assumptions, and therefore fails to identify relevant correlation between the independent and dependent variables to achieve reliable predictions. Both overfitting and underfitting models can result in models with poor predictive performance, which places emphasis of a balanced approach. This will allow the predictive model to progressively predict future occurrences within an acceptable margin of error.

Regression analysis is the process of examining the relationship between; a dependent variable and one or more independent variables. Through examining and utilising historical data, the relationship correlation between the two or more variables, if any, can be revealed, which will allow the forecast of subsequent data. For the development of a predictive maintenance solution, a regression-based model is required to use with the time-to-failure variable calculated within the analysis chapter, to forecast the events of possible future vibration-related failures. As the name implies, linear regression is a statistical process used to identify if a linear relationship exists between; a dependent variable (y), and one or more independent variables (X). The function of linear regression is relatively simple, by understanding the relationship condition between of variables y and X, the value for X can be predicted for future occurrences of y. Linear regression models can be further divided into two categories. In the scenario where only one independent variable x exists, the model is referred to as a 'Simple' linear regression model. On the contrary, if multiple independent variables exist, the model is regarded as a 'Multiple' linear regression model. Logistics regression modelling is primarily used to find the probability between two events. For example, the events may be success and failure, or broken and unbroken. For logistic regression to work as intended, the dependent

variable (y) must be a Boolean or binary value which represents the state between the two events. Although it is a regression modelling tool, it is widely used for classification problems, simply due to the nature of independent variables being grouped to one or the opposite state. Polynomial regression is a regression modelling tool like linear regression, as it too is used to identify the relationship type between a dependent variable and one or more independent variables. However, the main differences between the two modelling tools is that the relationship between X and y is modelled to the nth-degree polynomial. This produces a curvilinear relationship between the n-th power of y and X [12]. Polynomial regression only uses the operations addition, subtraction and multiplication.

A Predictive Maintenance Model

Microsoft has published a predictive maintenance dataset in the past, which is designed to be used within their Azure platform as a learning device [13]. This dataset satisfies the need to be relevant and usable, for the development and training of a predictive maintenance model, as it has been created with that purpose in mind. However, caution should be taken when directly comparing the results of this project with other predictive maintenance solutions, as although this 'simulated' dataset provided by Microsoft is publicly available, there is no credibility that this dataset will represent realistic data values. But nonetheless, it is a valid dataset for the development of a predictive maintenance algorithm.

Dataset description

The dataset available from Microsoft's GitHub repository comprises of five.csv files regarding machines, errors, maintenance, telemetry and failures. The breakdown of each file's purpose is as follows;

Machines: Contains individual attributes of each machine that helps to differentiate one another. This includes each machine's age and model type.

Error: A log of errors thrown during the operation of specified machines. Microsoft themselves state that these errors should not be considered mechanical failures by themselves. However, these errors may suggest the event of future failures [14].

Maintenance: A log of both scheduled and unscheduled maintenance transactions and outcomes. Unscheduled maintenance is the result of total failure by an individual machine, whereas scheduled maintenance only refers to regular planned inspections.

Telemetry: A log of metrics collected by each operating machine in real time. Operation metrics monitor changes in; voltage, rotation, pressure and vibration through sensors. The data for each variable is then averaged over an hour and is stored within these logs.

Failures: A log of machine failures designed to be used alongside the maintenance log. This failure log stores the specifics of each failure that has caused unscheduled maintenance. Data stored within these records include; the machine ID, the failure type and the date/time in which the required maintenance was performed.

The data within each file is fully labelled and is void of empty records. Therefore, not much data preparation will be required other than calculating the remaining useful life attribute for equipment. The next stage is to analyse the patterns of each operation variable, and to identify which variable is the most viable for determining potential future failure risks.

Variable selection

A popular rule within the data science community is the "No free lunch theorem". The rule, which when simplified, emphasises that no specific machine learning algorithm can adapt and perform well to multiple complex problems, and that the purpose of every machine learning algorithm should be as specific as the problem they are designed to solve [15]. Therefore, the predictive maintenance algorithm should be developed with a specific type of analysis in mind. Which means selecting a suitable variable that will form the basis of the supporting hypothesis.

Figure 3 shows the average values for each of the four operation variables found within the telemetry file. Analysing these values will be imperative for selecting the right variable, which will be used to identify variable-related potential failures for the machine learning algorithm to predict and verify. Analysis of each variables viability can be found below:

Pressure: Pressure values of equipment seem to fluctuate anywhere between 80-130 as normal operating parameters. However, most pressure activity appears to be between the values of 90-105, and there are a few occurrences of values outside this normal threshold

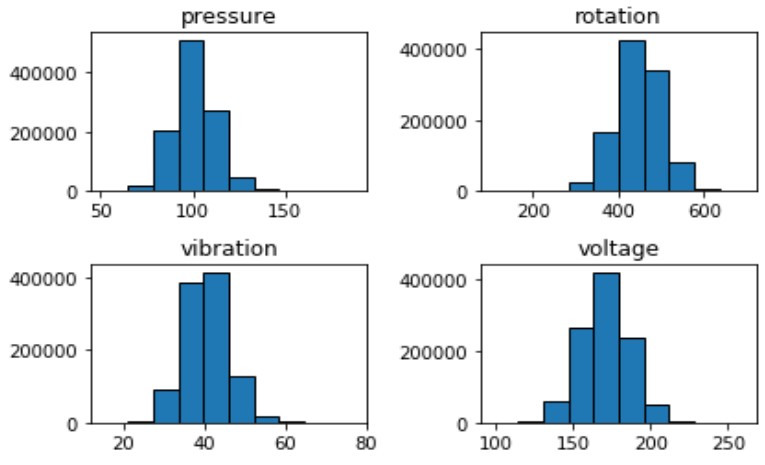


Figure 3: Histogram of dataset variables.

(under 80 or over 130). In terms of potential failures, low pressure may suggest that a failure has already occurred in the form of a leak, while extreme highs could suggest that pressure is about to implode, which would also result in a failure. While these scenarios are plausible, there appear to be too few cases in the dataset available to be able to test and validate pressure-related risks when developing the machine learning algorithm. Therefore, if pressure is selected as the chosen variable for a pressure-related hypothesis, it may not actually provide any meaningful pressure analysis due to how little data exists to test.

Rotation: Figure 3 shows that rotation values are diverse and can range anywhere from 250 to 600, however most rotation values seem to be equal to 400 or above. There are very few cases of 'extreme' low rotation values (under 250). Although there are records of values much higher than the average rotation value, which could include values outside the normal operational threshold. However, different tools are likely to have different rotation values depending on the process they were designed for. Furthermore, the result of high rotation would not necessarily be a clear indication of imminent or even eventual failure. As rotation-related failures are often the results of total failure by another component. For example, as rotating tools are prone to vibration as bearings age, vibration analysis would result in more meaningful research than rotation analysis. Therefore, rotation will not be considered for further analysis, as rotation values can be indicative of a failure that has already occurred but cannot be the sole determining factor when identifying potential future failures that have not occurred yet.

Vibration: Vibration appears to be an ideal candidate for vibration-related predictive maintenance, as the 'normal' parameters for vibration seem to be within the 36 to 46 range. High vibration values can suggest misalignments, loose connections, unbalance or vibrations too high for the intended systems tolerance, which given time, could result in connections being severed, which could cause total failure for the individual machine. However, lower than average vibration values are not indicative of anything. As it is hard to distinguish if extremely low values are the by-product of a machine performing well, or a machine not working at all. It requires prior knowledge of the process being used within the dataset which is not available. Therefore, if vibration analysis is chosen for the predictive maintenance solution, only higher than average vibration values will be of any use for identifying potential vibration-related failures.

Voltage: Voltage, regardless of the values, requires prior knowledge of the process being used, as different processes require different tools which will have different operating parameters. Some equipment may require a very low voltage current, while others may require a high amount of voltage. It is highly situational, which is the problem, as the dataset does not reveal which processes it is simulating, as it was never designed to. However, there is an argument to be made that fluctuating voltage for individual machines is a tell-tale sign of a faulty power supply, which is plausible, but may add unnecessary complexity to the predictive model, which may have an adverse effect for the predictive maintenance solution.

Vibration appears to be the most viable variable for further analysis, as it appears that large amounts of vibration values are closely grouped together. Vibration also has a substantial amount of records that are significantly higher than the average, which could be machines with potential vibration fatigue or vibration-related issues.

Vibration analysis

With vibration selected as the most viable variable to simulate failures within the available dataset, further vibration analysis is required to identify the causation of vibration-related failures. Through further understanding of the vibration data within the telemetry file, it will be easier to form a feasible hypothesis which will calculate the remaining useful life of assets. This remaining useful life variable can then be used by the predictive maintenance algorithm to identify future vibration-related failures. However, the first step in vibration analysis is to visualise the vibration values stored within the dataset.

Figure 4 shows the result of visualising the vibration activity for four randomly selected machines throughout the course of one year. The machines randomly selected within the dataset were machines; 12, 24, 72 and 112, which represent example A, B, C and D respectively. Each sampled machine above displays occurrences of both extreme high and low values, relative to the estimated average vibration value of 40. However as discussed during the variable selection argument, not many problems derive from experiencing low vibrations therefore, further analysis is required.

The next step was to reduce the scope of the vibration activity and to visualise the vibration values for randomly selected months for each sampled machine (Figure 5). Sampled machines A, B and C have been highlighted as they have each displayed higher than average vibration levels for a substantial period. Whereas example machine D only shows abnormal vibration activity for short periods, before de-escalating back down to normal values. It was made abundantly clear that machine D would act as the control variable for future analysis. After discovering this abnormal activity for three of the sampled machines, comparisons were made to the datasets included failure log. When looking for failures within the same month as the three sampled machines, it was discovered that all three sampled machines experienced total failure during or shortly after the abnormal vibration activity occurred. While machine D, having displayed no abnormal activities, had no failures occur within the same month of July, and in fact the next failure machine D experienced was in mid-September. Therefore, the next action to take was to analyse the vibration activity for machine D within the month of September when the next failure occurred.

Figure 6 shows the results of further analysis, which revealed the same abnormal vibration activity that had appeared for the other three sampled machines. It also appears to have concurred just before the machine D experienced total failure on the 15th of September. This appears to confirm that when machines experience this abnormal vibration activity, total failure follows shortly after. Unfortunately, since there are no yield metrics included within the dataset, no comparison of yield performance during these abnormal vibration activities can be made. However, this still proves valuable, and can be used to form the basis of a vibration-related hypothesis. Therefore, the next step will be to determine how to calculate the remaining useful life for equipment before an expected vibration-related failure occurs.

RUL calculation and hypothesis

To be able to determine when a specific machine is likely to fail, a special variable called the Remaining Useful Life (RUL) mainly used in prognostics needs to be calculated. The RUL is as simple as it sounds and will be used to store the specific number of days until

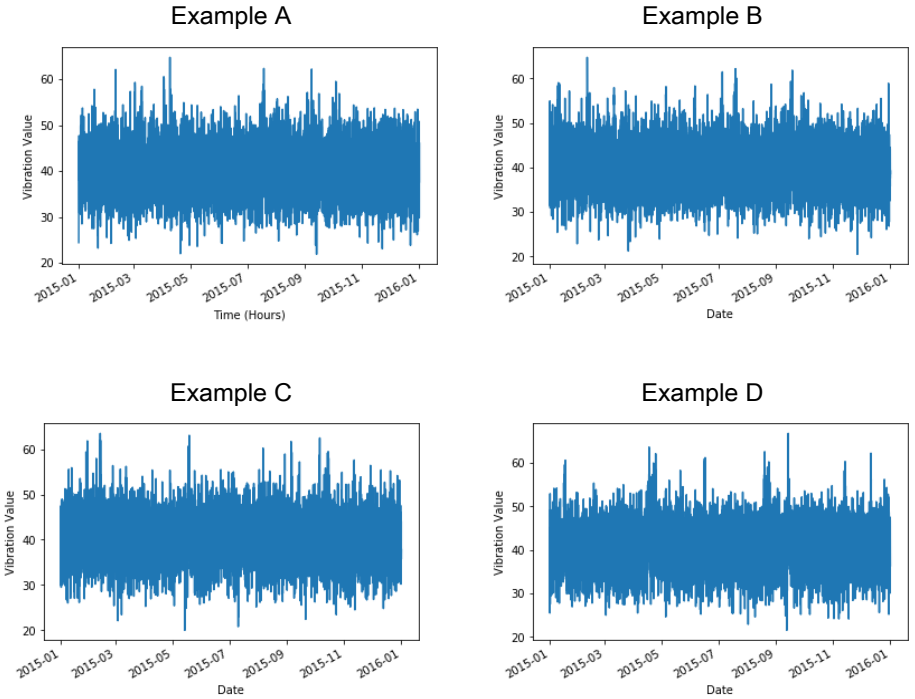
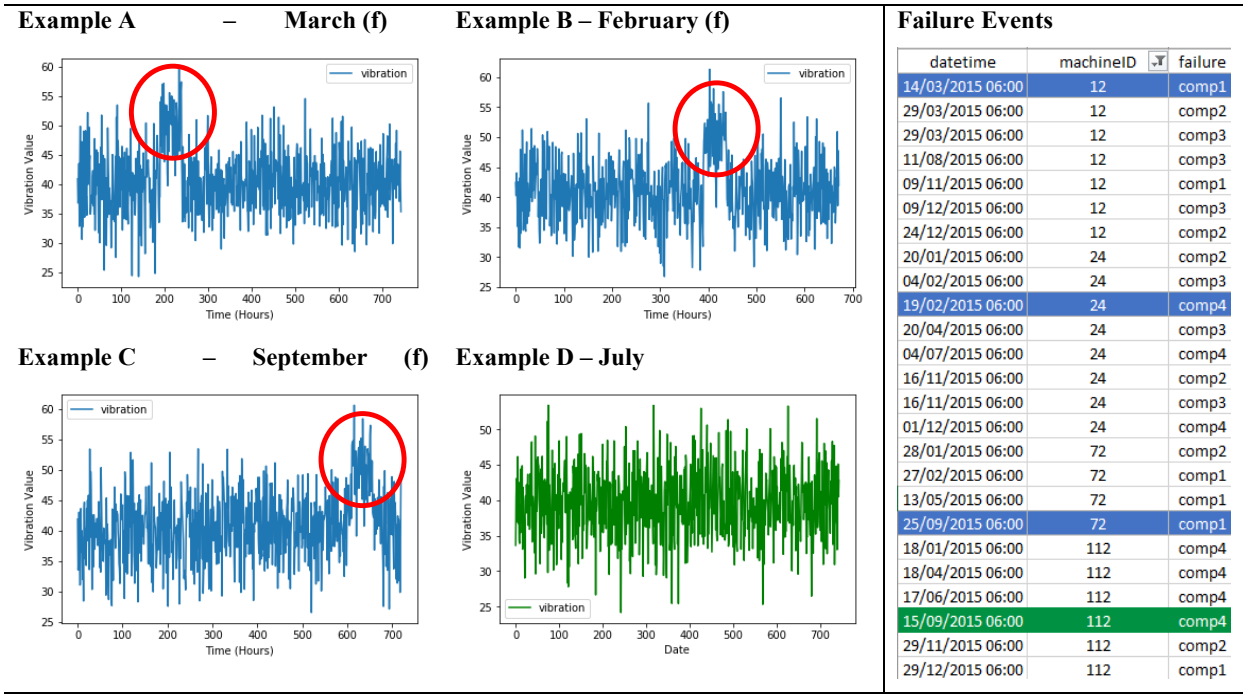


Figure 4: Time series of vibration activity by randomly sampled machines.



Failure Events

datetime	machineID	failure
14/03/2015 06:00	12	comp1
29/03/2015 06:00	12	comp2
29/03/2015 06:00	12	comp3
11/08/2015 06:00	12	comp3
09/11/2015 06:00	12	comp1
09/12/2015 06:00	12	comp3
24/12/2015 06:00	12	comp2
20/01/2015 06:00	24	comp2
04/02/2015 06:00	24	comp3
19/02/2015 06:00	24	comp4
20/04/2015 06:00	24	comp3
04/07/2015 06:00	24	comp4
16/11/2015 06:00	24	comp2
16/11/2015 06:00	24	comp3
01/12/2015 06:00	24	comp4
28/01/2015 06:00	72	comp2
27/02/2015 06:00	72	comp1
13/05/2015 06:00	72	comp1
25/09/2015 06:00	72	comp1
18/01/2015 06:00	112	comp4
18/04/2015 06:00	112	comp4
17/06/2015 06:00	112	comp4
15/09/2015 06:00	112	comp4
29/11/2015 06:00	112	comp2
29/12/2015 06:00	112	comp1

Figure 5: Time series of vibration activity that correspond with failures within the dataset failure log.

the total failure is expected to occur. To calculate the RUL, we find a position A which represents the point of time in which a machine has been diagnosed of performance degradation. We also identify a point B which represents the minimum level of acceptable performance of the individual machine. The RUL of a machine is then calculated by estimating the distance between points A and B. However, because the dataset only gives a snapshot of vibration variables for a limited time, nor is there any yield metrics to measure, it is not possible to determine if machines are affected by performance degradation for the dataset being used. This means no minimum acceptable performance

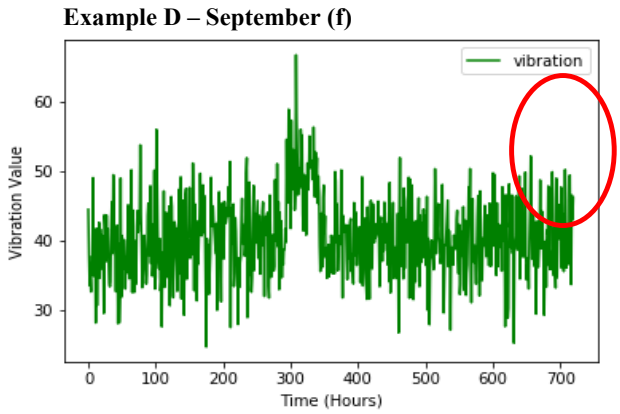


Figure 6: Time series of vibration activity for machine D in September.

can be established. Therefore, the RUL will still be calculated, but the term TTF will be used to replace the concept of the RUL. Both serve the same purpose, but TTF more accurately describes the situation of identifying vibration-related failures with no performance related metrics considered. The pattern of which vibration escalates needs to be identified and calculated to be able to predict vibration-related failures instead.

Calculating that vibration pattern is carried out with the formula maxWeekVib – AvgWeekVib = trend. By taking the maximum recorded vibration of a specified week and reducing it against the overall average of vibration values for the same entire week, will result in what will be referred to as the vibration 'trend'. This trend will represent the increase in movement for vibration-related activity. The next step is to divide the trend against the current vibration value. RUL calculation for vibration analysis is shown in Equation 1.

$$(\text{currentVib}/(\text{maxWeekVib}-\text{avgWeekVib})) \times 7 = \text{TTF(Days)} \quad (1)$$

By dividing the last recorded vibration for the current week with the vibration 'trend' identified earlier, this will calculate the estimated TTF value in week format for individual machines. Therefore, the result of the above calculation must be multiplied by seven, to attain the true TTF value in a day format, which will be used for the predictive maintenance algorithm. Multiple data science libraries were used to create the machine learning algorithm including Pandas, NumPy, Matplotlib and Sklearn.

Results, Discussion and Evaluation

As the failure logs contains all failures for every tool, it was decided very early in the development that the best verification of the algorithm's prediction accuracy would be through measuring the algorithms predictions using the Time-to-failure hypothesis, with the actual failure logs. As the dataset includes the vibration related information and failure information for the entire year of 2015, the test plan shown in Table 2 was created:

To verify accuracy, the algorithm was trained using the data generated from one of the training months. Those predictions would be plotted with the actual failure date for each machine within the two test months. On the chance that a machine did not fail within either of the two specified test months, then the record for that machine was deleted rather than letting it risk the scale of the test. Figure 7 shows that the verification data for the first test is generated using variables from January's production data. The predicted Time-to-failure values for each machine is tested and verified with the actual failure dates of the same machines found within the failure logs. Although some machines may have no failure dates within the specified test months, in that event the machine record was deleted. Only the training data from the first 4 weeks will be considered in the calculation of the Time-to-failure variable. As although this may provide estimated values, especially for months like October that have 31 days within the month. The calculation of the first 28 days per month should provide a balanced standard as each month in the year has at least a minimum of 28 days. It also means that the Time-to-failure calculation equation created in chapter 2 during the hypothesis case study does not need to be tweaked for individual months. This may result in underestimated values being calculated for the Time-to-failure variable, and therefore the predictions made, but these potential calculation errors should be within an acceptable margin of error and will be considered when evaluating the test results of the prototype.

Verification

The following show the prediction results measured against the actual failures of machines using the test plan.

The predicted results from the January-February-March test in Figure 7 showed promise, but also showed signs of overfitting.

The predicted results from the April-May-June test shown by Figure 8 removed the appeared overfitting.

The predicted results from the July-August-September test shown by Figure 9 showed that more accurate predictions were being made.

The predicted results from the October-November-December test shown by Figure 10 showed the best results with multiple accurate predictions being made, but still too few cases. When verifying the results, the accuracy of the predictions made by the algorithm using the Time-to-failure variable was significantly lower than expected. With only few occurrences of accurate predictions, it is not certain whether those predictions are in fact accurate, or simply just down to luck.

Figure 11 shows the extent of this situation:

This may be the result of not all failures within the failure log being directly related to vibration-related failures. Therefore, failures that have been caused by another component failure have not been identified, which renders the predicted Time-to-failure value useless for non-vibration related failures which may be obscuring the prediction results. Another possibility is that the predictive model simply cannot generalise to the new data given, as telemetry and failure logs are separate. It seems the algorithm suffered from overfitting, as predictions were made that the TTF for certain equipment would be over 250 days. However, in the following tests, there were occurrences of actual TTF values from the failure log with values that large, as seen from Figure 9. Unfortunately, the root problem cannot be determined regardless, as the dataset simply does not include enough information required to determine which failures in the dataset are explicitly vibration-related failures. Regardless of the few accurate

Table 2: Test plan.

Training	Test
January	February, March
April	May. June
July	August. September
October	November, December

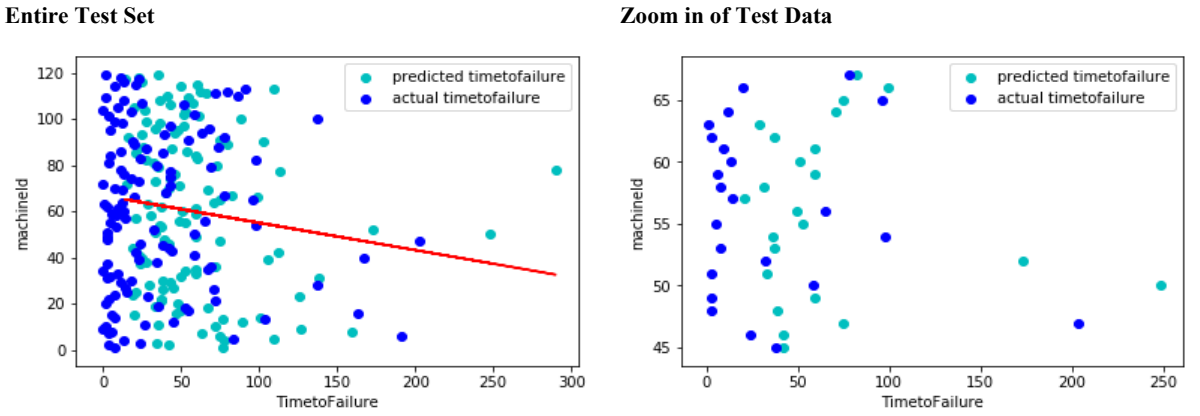


Figure 7: January-February-March predictions.

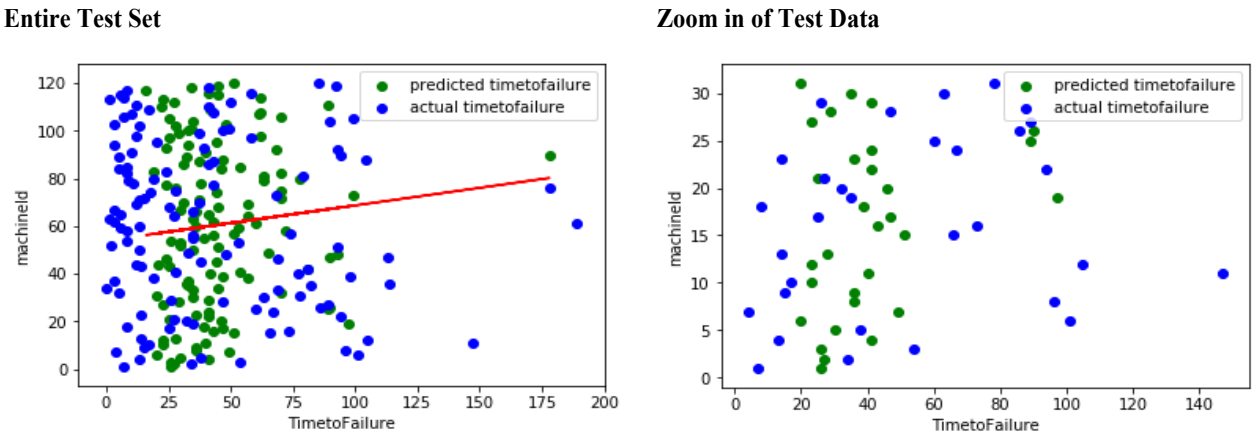


Figure 8: April-May-June predictions.

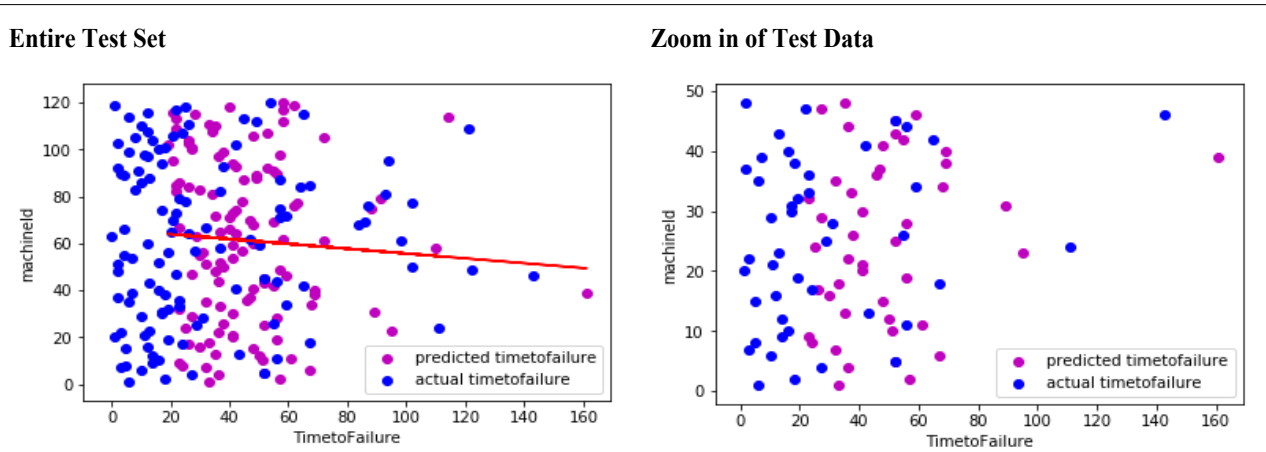


Figure 9: July-August-September predictions.

predictions made by the algorithm, the overall accuracy over the four tests is underwhelming at best. Within its current state, the prototype would not be suitable as a predictive maintenance solution, as it is completely transparent that the prototype requires additional work outside the project plan.

Conclusion

Predictive maintenance requires predictions that are reliable. We achieved this through linear regression which analysed the relationship between vibrations and failures. Semiconductor manufacturers care about yield results. Therefore, in a live environment consideration

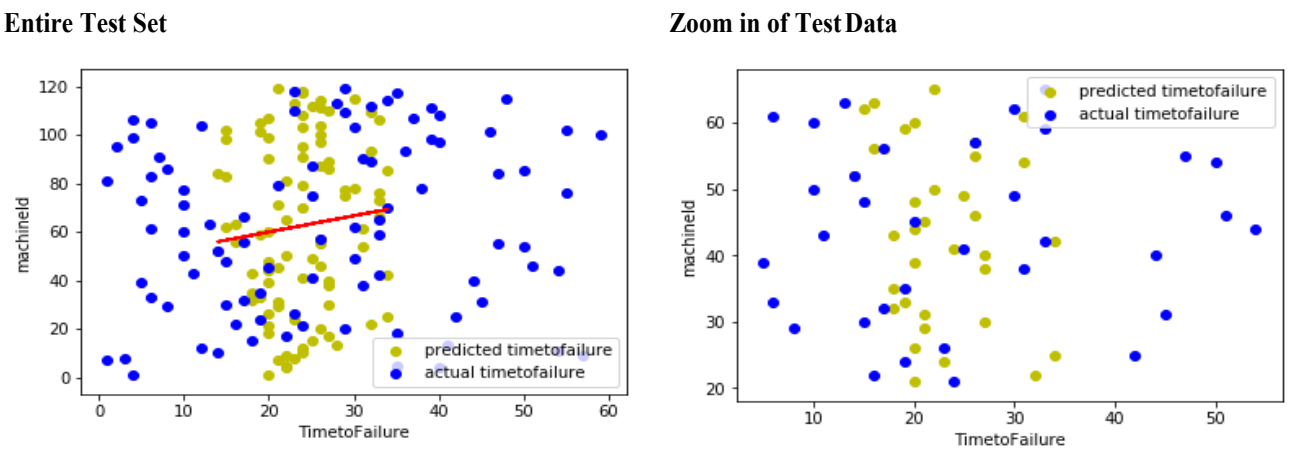


Figure 10: October-November-December predictions.

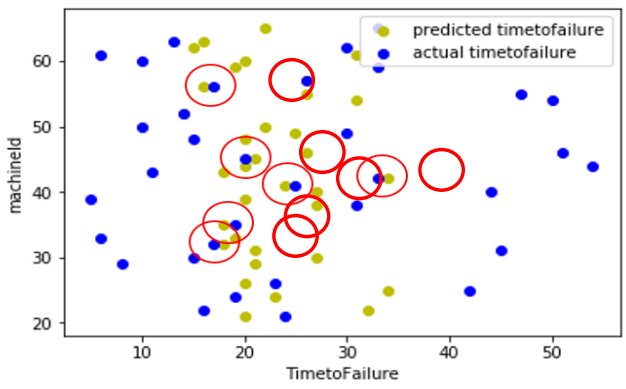


Figure 11: Count of accurate predictions within manageable margin of error for Oct-Dec predictions.

must be given to the yield output by each machine and would likely be the sole variable for measuring performance degradation in semiconductor equipment. Our future work will employ multi-class classification to verify if an asset is likely to fail outside of a predicted time window. For example, you would have labels for scenarios when the asset indeed fails within the time window and labels for assets that fail before or after their predicted failure window date to further help improve the accuracy of the developed predictive models [16].

References

1. Batra G, Jacobson Z, Santhanam N (2016) Improving the semiconductor industry through advanced analytics. McKinley Company.
2. Lapedus M (2017) Why fabs worry about tool parts. Manufacturing, Packaging & Materials.
3. Vesely E (2018) Functional and non-functional requirements for an IIOT predictive maintenance solution.
4. Kagermann H, Lukas W, Wahlster W (2011) Industry 4.0: With the internet of things on the way to the 4th industrial revolution. VDI news 13.
5. Schwab K (2016) The fourth industrial revolution. Crown Publishing Group, New York.
6. Rüßmann M, Lorenz M, Gerbert P, Waldner M, Justus J, et al. (2015) Industry 4.0: The future of productivity and growth in manufacturing industries. Massachusetts: The Boston Consulting Group, Boston.
7. Scheffer C, Girdhar P (2004) Practical machinery vibration analysis and predictive maintenance. Elsevier.
8. Geissbauer R, Vedso J, Schrauf S (2016) '2016 Global industry 4.0 survey', industry 4.0: Building the digital enterprise.
9. Mobley RK (2002) An introduction to predictive maintenance. Butterworth-Heinemann, Amsterdam.
10. Engel G (2016) 3 Flavors of machine learning: Who, what & where. Dark Reading.
11. McQuarrie ADR, Tsai CL (1998) Regression and time series model selection. World Scientific, Singapore.
12. Pečkov A (2012) A machine learning approach to polynomial regression. Jo˘zef Stefan International Postgraduate School, Ljubljana, Slovenia.
13. Mathew J (2016) SQL-server-r-services-samples. GitHub Repository, USA.
14. Ehrlinger J (2017) Predictive maintenance for real-world scenarios. Microsoft Azure.
15. Wolpert DH, Macready WG (1996) No free lunch theorems for search. Technical Report SFI-TR-95-02-010 (Santa Fe Institute), Santa Fe, New Mexico.
16. Zhang Y (2015) New predictive maintenance template in azure ML. Microsoft.

Author Affiliation

Top

[1]*School of Computing, Engineering and Intelligent Systems, Magee Campus, Ulster University, Coleraine, Northern Ireland*

[2]*Faculty of Computing, Engineering and Built Environment, Magee Campus, Ulster University, Coleraine, Northern Ireland*