White Paper

Predictive Maintenance Using Normal Behavior Modeling

ABSTRACT

Normal behavior modeling (NBM) is an approach to process, system, and equipment management and maintenance enabled by recent advances in the fields of artificial intelligence (AI) and machine learning (ML). This white paper describes the evolutionary forces that have led to this new capability set, its applications (current and prospective), its benefits, and details of how the technology works. In addition to the explanation of NBM's operation and implementation, several examples are provided that highlight the benefits of this technology.

INTRODUCTION

This white paper provides an in-depth introduction to the topic of NBM so that corporate and governmental decision-makers will understand its capabilities, requirements, and potential limitations, thus enabling them to make informed decisions about the technology's applicability to their particular operational challenges. Much has been written about AI in recent years, particularly its ability to identify anomalies in the otherwise normal operation of equipment and processes. But there remains an air of mystery about the technology, especially concerning the ways in which specific alerts and notifications are generated and how they should be responded to (if at all). There is, in short, a bit of a 'black box' aspect to AI and techniques like NBM, and it is our goal to provide a peek inside that box.

In addition to providing insight into how NBM works, a further objective of this white paper is to identify specific use cases in which it delivers tremendous business value. NBM is applicable to a wide range of operational fields—in practice, any in which 'normal' operation can be quantified from available data—and we will identify several of them. They include everything from maintaining the reliability of production assets in manufacturing plants and oil and gas facilities to reducing carbon emissions from refineries and optimizing the operation and maintenance activities of aviation and renewable energy firms.

Modern industrial equipment routinely costs millions of dollars to purchase, maintain, and operate. The goal of any industrial concern is not only to employ that equipment to maximize production (and hence revenue) but also to manage ongoing expenses by minimizing routine or unscheduled maintenance and increasing the useful life of these expensive capital assets. Historically, these goals have been pursued using condition-based monitoring (CBM) solutions or OEM-provided asset management tools incorporating physics-based models. However, in a world where equipment is increasingly reliable and failure data is scarce, traditional approaches are often far less effective than the NBM approaches described here. The goal of NBM is to facilitate the achievement of these goals, i.e., minimizing operational costs while maximizing revenue generation and equipment lifetime, in a more proactive and effective manner. But before diving into the details of how NBM accomplishes these things, it's worth taking a moment to consider how we have gotten to where we are today in the field of NBM-based anomaly detection.

EVOLUTION OF MAINTENANCE APPROACHES

Normal behavior modeling is an AI-enabled analytical technique that can be applied to an entire end-to-end process or one particular piece of equipment, such as a hydraulic pump or wind turbine. However, long before the term NBM came into existence—since the invention of the earliest mechanical devices—equipment and process operators have been tasked with keeping them running for as long and as continuously as possible.

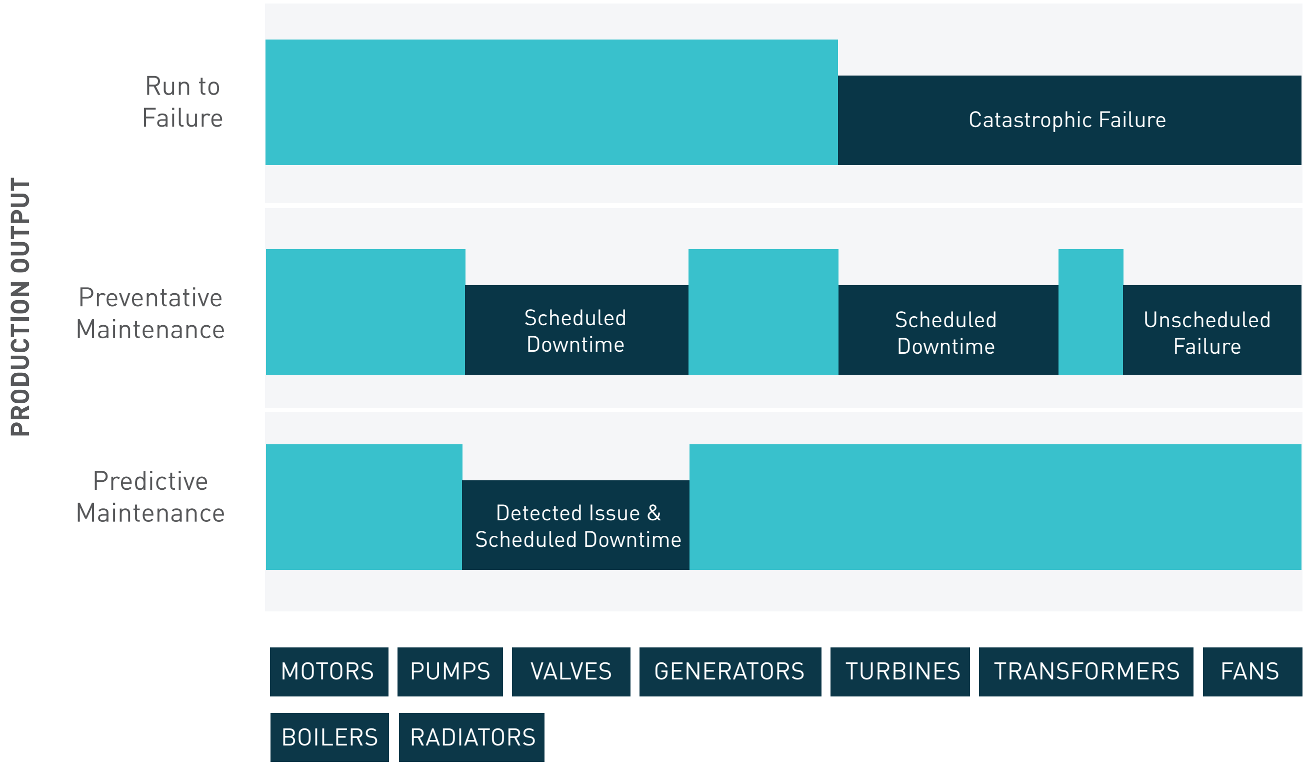
In the earliest days, this meant nothing more sophisticated than repairing a piece of equipment when something failed (run-to-failure approach). If a wheel on your oxcart broke, you either repaired it on the spot or you took it to a wheelwright who fixed it for you or sold you a new one. This approach is ideal for assets/machines that are very cheap to replace and the asset failure will not cause catastrophic damage to adjacent assets, endanger human lives, or result in significant revenue losses from downtime.

As time and technology progressed, operators got better at identifying conditions suggesting that a failure was imminent, and they became equally creative at coming up with maintenance actions that could be taken in advance to prevent an all-out failure (preventative maintenance approach). With a better understanding of how pieces of the machine could be expected to degrade over time, experts would set expiration dates for critical components and replace them periodically. The downside of this approach was that because expiration dates were somewhat arbitrary or based on each component's average expected life, perfectly good parts were often replaced, incurring unnecessary costs and installation-related downtime.

By the time of the industrial revolution in the late 1800s, with the arrival of textile and paper mills, munitions factories, etc., the equipment had become quite complex and frequently temperamental to operate, meaning that factory bosses had to employ experts whose sole purpose was to keep everything running smoothly. The expertise required to do this job was often no more complex than listening to a machine's sounds, feeling its vibrations, or staying alert to unusual smells. Eventually, though, mechanical gauges arrived on the scene and industry had its first sensors, a new capability that still plays a critical role to this day. With the ability to monitor quantifiable measures like temperature, pressure, and flow rate, and to track these measures over time, the possibility emerged of understanding what qualified as 'normal' behavior for a piece of equipment, predicting when something was about to go wrong, and doing something about it in advance (predictive maintenance approach).

Fast forward to the middle of the 20th century, and operations experts were applying statistical techniques to interpret the growing quantities of data they were collecting (often manually) about their

Figure 1—Production Output vs Maintenance Approach



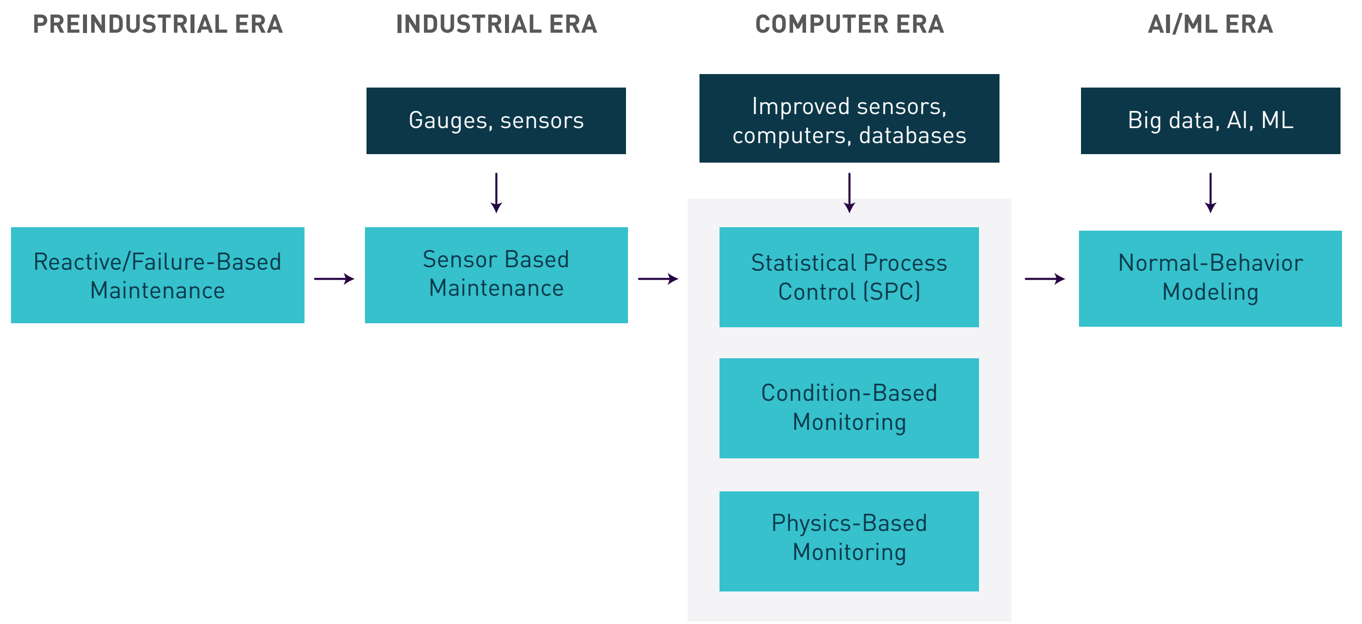
systems. Increasingly advanced mathematical analyses were applied to operational data, culminating in the 1970s and 1980s with the emergence of statistical process control (SPC), a new capability that identified upper and lower tolerances for a specific performance measure and automatically determined when a tolerance was exceeded, often an early indication of an incipient problem. There were, though, important limitations of these techniques. First, this approach was applied to the time-series data of a single metric, e.g., the pressure of a pump, the temperature of a reactor, etc. The ability to combine numerous values into a single holistic view of system performance was still very much in the future, let alone properly triggering anomalies driven by the interplay between multiple, interdependent variables (i.e. multivariate). Further, the threshold alerts provided by SPC systems were frequently indicative of a problem that was already occurring, meaning it was late in the game to do anything proactive about it.

Around this time, condition-based and physics-based performance modeling began taking precedence. In the former case, operators conducted continuous monitoring of numerous sensor outputs, based either on their own prior knowledge of system operations or on guidelines provided by original equipment manufacturers (OEMs) of these systems. By evaluating equipment performance in real-time, operators became aware of problems, though typically with little or no advance notice, rendering repairs after-the-fact situations.

Physics-based models, on the other hand, relied on subject matter experts to model and simulate the operation of a system or piece of equipment by defining it with a series of mathematical equations. These equation sets could do a good job of defining a system's operation under predictable conditions but faced challenges as system complexity grew and in the face of dynamic conditions over time, particularly exogenous events like weather and age-based changes in the operating environment. Especially challenging for physics-based models is their tendency to degrade over time due to operations, wear and tear of equipment, and other changes that cause the equipment to deviate from specifications originally provided by the OEM. Keeping up with these changes requires periodic re-tuning of the models, a difficult and time-consuming task. Despite the limitations of these methodologies, many system operators today still maintain their equipment using some combination of physics- or condition-based maintenance approaches.

With the turn of the 21st century and the arrival of big data, high-bandwidth data transmission, and petaflop computing capabilities, a new universe of possibility came into being. Operators had understood for a long time that a system's overall performance-–its ability to do its job—depended not only on the correct values of individual performance measures right now but also on the collective achievement of many measures working in concert over extended time periods. And it is this need for comprehensive monitoring and analysis of increasingly complex systems that normal behavior modeling satisfies with often unmatched precision.

Figure 2—Evolution of System/Equipment Maintenance



NBM ADVANTAGES

Holistic system understanding

There are several direct advantages of NBM, but most significant is its ability to continuously monitor and provide proactive alerts of impending maintenance problems for a complex system of components, basing these alerts not only on the values of individual parameters but on a holistic understanding of the system in its entirety. This is unique compared with simpler methodologies such as univariate statistical process control (SPC) and condition-based monitoring (CBM), each of which bases its alert outputs on the value of a single variable. In the simplest implementation, for example, univariate SPC establishes statistically-determined maximum and minimum threshold values for a time-series data stream, then provides alerts when one of these thresholds is exceeded for a predefined period of time or number of instances. CBM works similarly, using threshold values typically provided by the OEM of the piece of equipment being monitored. There are numerous flaws with both of these approaches to system monitoring, but the primary one is failure to recognize the inherently interconnected nature of complex systems, both to other components in the system and to the external environment.

Multivariate statistical process control approaches can be employed to address some of these limitations, but their cost and difficulty of implementation grow quickly with the complexity of the system and number of variables in question. A workaround for this is to use prior knowledge to keep the number of surveillance variables to a minimum, but the tradeoff is a reduction in system visibility and the ability to alert on new problems (i.e. unknown unknowns) that have rarely or never before been encountered. This carries significant risk in today's industrial environments where complex operations are enabled by interconnected systems consisting of an ever-increasing mix of sophisticated assets from a wide array of vendors.

Another common approach to system monitoring is physics-based modeling, which uses a complex series of mathematical equations to describe how a system or component should perform under normal circumstances. This approach is implemented by subject matter experts (SMEs) who use prior knowledge to implement a rules-based model or simulation. Key advantages of these models include interpretability and predictive power. However, similar to the other approaches cited above, they are challenged by increasing system complexity and the computational requirements required by large-scale models. The dynamic nature of modern industrial environments also poses a problem where models require the ability to adapt quickly as operating parameters evolve with time—no small task for a more rigid rules-based approach. Maintaining model performance can be daunting as assets grow older since mathematical equations are fixed and do not age. An important manifestation of this problem is an increased rate of false positives with the passage of time.

NBM, by contrast, evaluates the system in a holistic manner, recognizing explicitly that the status values of one element of the system can be directly affected not only by the status values of other elements of the system and of the external environment but also by characteristics such as the age of the system's components and changes in what comprises 'normal' for the system. Stated differently, what qualifies as normal on the day a system is first turned on may bear little resemblance to what qualifies as normal a few months or years later. And this is the key difference between NBM and the other discussed approaches: traditional physics-based models, CBM, and SPC methods work best on day 1 of operations whereas data-based approaches like NBM continue to improve with time.

In many ways, considering NBM vs. a physics-based approach as an either/or proposition is a false dichotomy. In fact, an NBM approach can be complementary to a physics-based tool, augmenting the overall solution in significant ways. In addition, the data utilized by NBM is generated from underlying physical processes. Instead of relying on SMEs to define the equations that correspond with those underlying physical processes to drive physics-based modeling and simulation, NBM leverages the power of machine learning to infer those rules automatically from the data itself, even as the system continues to evolve over time.

Unsupervised learning

Another important advantage of NBM is its ability to detect and provide alerts on impending failure modes without having been exposed to those failure modes in advance. This raises the notion of so-called supervised and unsupervised performance modeling. In supervised learning, the system is trained on historical data sets to recognize a predefined subset of output conditions explicitly. In the case of image recognition, for example, supervised models are frequently trained to recognize a specific outcome (say, identification of a cat in a photo) by being shown large numbers of images of that outcome, i.e., photos containing cats. In such instances, each piece of data (each photo) is said to be labeled, i.e., either 'cat' or 'not cat.' In the analogous predictive maintenance scenario, a supervised system would learn about the various failure modes of, say, a hydraulic pump by being shown in advance all the various ways in which a hydraulic pump can fail along with data sets containing labels that correspond to these failure modes. The obvious weakness of this approach is that it renders the system incapable of identifying and alerting on failure modes not foreseen in advance by trainers. And, of course, this approach also suffers from the additional requirement for someone to take the time to identify and codify all of the failure scenarios that will be provided to the system, a challenge made even more daunting by the increasing reliability of modern machines, a result of which is reduced access to failure data for use in training.

Unsupervised learning, on the other hand, requires none of this. The system is fed a large data set from the normal operating equipment. Once the model has learned what comprises 'normal,' it autonomously provides alerts on situations that deviate from that state. This approach also continuously revises its understanding of 'normal' for the system, meaning that its knowledge is dynamic, a key benefit whose importance will be explored more in subsequent sections.

Data agnosticism

Another important distinction between NBM and other forms of system monitoring is the notion of data agnosticism. An NBM model is unconcerned with what kind of equipment it is monitoring. Whether wind turbine, fuel pump, or nuclear reactor is of no relevance to the NBM model. The model evaluates an input data stream, develops its understanding of normality, and triggers alerts whenever it perceives that normality has been violated.

Other advantages

Besides the holistic system understanding, unsupervised learning, and data agnostics capabilities described, NBM yields additional advantages:

- More effective in monitoring (internally or externally) dynamic environments.
- Provides alerts on anomalies BEFORE failures occur, saving money and time.
- Strikes an ideal balance between alert fatigue from too many false positives and missed problems as well as from too many false negatives (i.e., alerts that should have taken place, but did not).
- Adaptable and scalable as system complexity increases since it's relatively straightforward to add new or retrain existing normal behavior models as new assets (and their associated sensor inputs) are added.

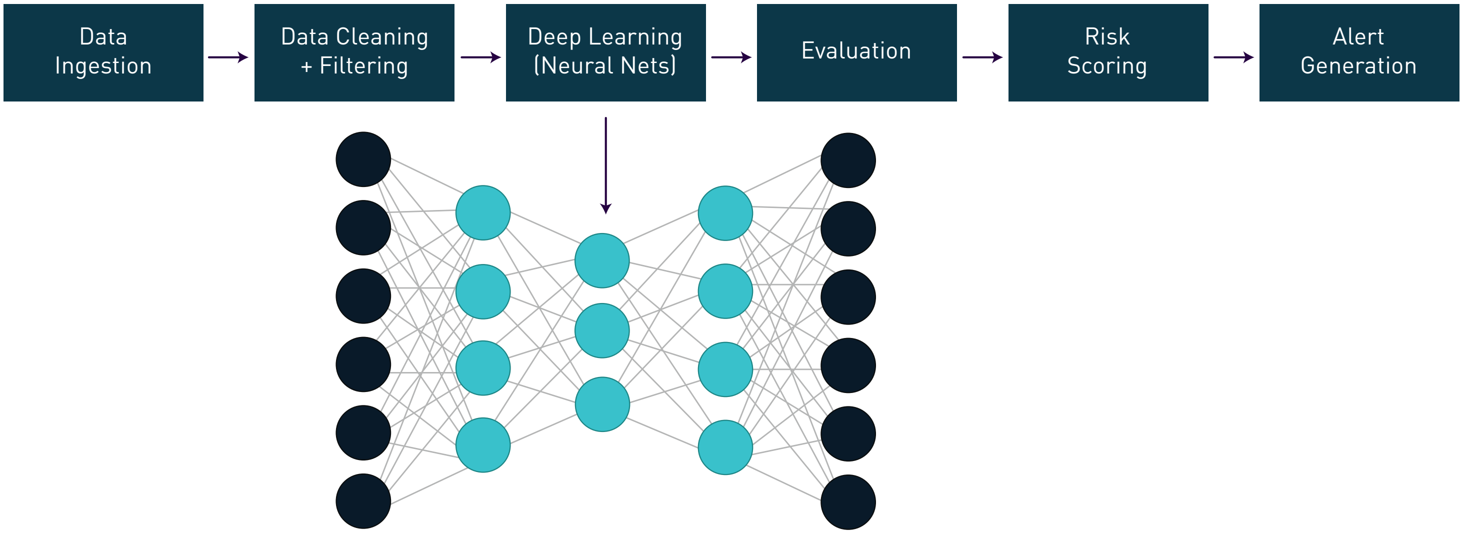
THE TECHNOLOGY OF NBM

Normal behavior modeling is an automated AI/ML-enabled anomaly detection methodology for evaluating and describing the behavior of a system or piece of equipment under normal operational and environmental conditions. NBM models ingest large volumes of quantitative time-series data (temperature, pressure, flow rate, etc.) from multiple sensors, both initially for training purposes and continually thereafter for ongoing monitoring and periodic retraining of the system. Once trained to understand the quantitative characteristics that define 'normal' for the system in question, the model continues to evaluate the incoming sensor-provided data stream and generates alerts whenever an out-of-normal condition is detected. Managers and technicians can then use these alerts to undertake maintenance and repairs of the system more proactively than doing so only upon system failure, thus saving time and money and improving the overall productivity and safety of the system, the facility in which the system operates, and the workers who interact with it.

NBM models are used in a wide variety of capacities, but their applicability falls primarily into the predictive maintenance field. They are used to make more effective and efficient the monitoring and maintenance of complex physical or virtual systems, comprising either a single complex device (e.g., a jet turbine), an interconnected series of physical devices (e.g., the equipment on an oil platform or refinery), or a complex process (e.g., sales of a product on an online platform). To employ NBM modeling, the only requirements are a continuously operating system comprised of multiple components, status and performance data from sensors attached to those components, and one or more quantifiable outputs from the system.

The process described in the following sections is summarized in Figure 3, including all steps from initial data ingestion/cleaning/filtering through feature extraction, weighting, risk scoring, and alert generation.

Figure 3—NBM End-to-End Process



Data ingestion, cleaning, and filtering

As with any data-driven analytical exercise, the quality of data provided to the NBM model will determine the quality of output results (in this case, the veracity of alerts). This includes standard data-management best practices, such as handling spurious or out-of-range data or entirely missing data, the latter of which can be dealt with by interpolation or inference based on other related data points. The autoencoder architecture described in the following sections is extremely robust with respect to handling noisy data, enabling it to perform well even when presented with less-than-perfect sensor inputs.

It is a useful first step in any NBM development process to decide on the frequency and granularity of the data that will be used. While it will generally be true that more is better, there will be points beyond which processing times become cumbersome and the value of even more data will begin to result in diminishing returns. Thus, because there exist sensors that provide output data every second and others that provide a reading once each hour, careful consideration should be given to the input data frequency required versus the timeliness of outputs that will be needed. A commonly employed test of these requirements is to create and run an initial NBM model on a large historical data set during which known failures occurred, to gauge the extent to which the model can predict those failures post facto. This is not only an excellent tuning exercise prior to model development, it also goes a long way toward creating organizational comfort with the entire modeling process.

Deep learning and neural networks

Artificial neural networks (ANNs) are a popular form of machine learning loosely inspired by the organizational structure of biological neural networks found in animal brains. In this structure, an input and output layer and one or more internal "hidden" layers are interconnected sequentially. This organizational arrangement, coupled with the particular activation function implemented by each node, gives neural networks the power to learn with high accuracy, even from highly dimensional input data sets.

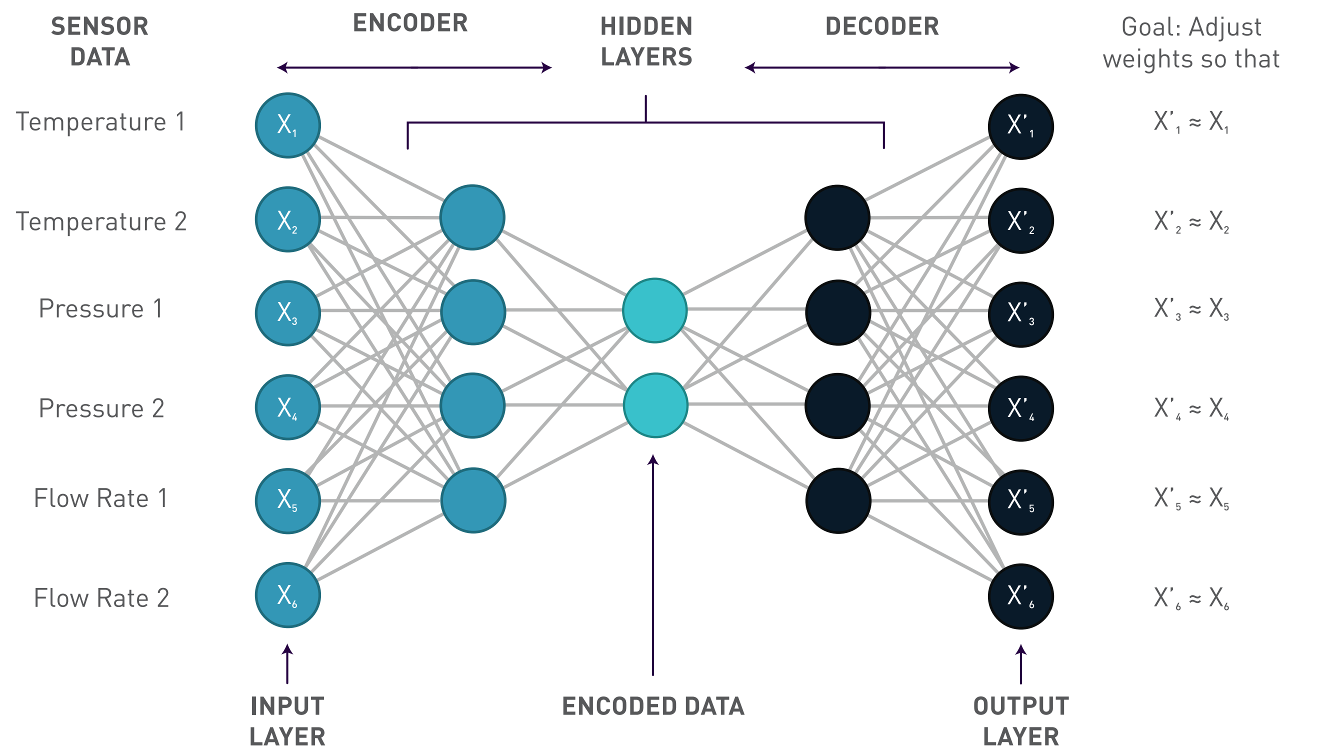
As the depth of inner node layers increases, so does its learning power, given sufficient training data. This is, in fact, where the notion of deep learning comes from, i.e., when multiple hidden layers are implemented between the input and output layers. Deep learning models have tremendous learning capacity and deliver outstanding performance for many tasks where the underlying data set is composed of hierarchically related elements and/or time-series data. During training, the hidden layers learn the important features of the input data set automatically, which is of great benefit in situations where costly manual feature engineering would otherwise first need to be applied.

Normal behavior modeling is an anomaly detection methodology frequently implemented with a form of deep learning architecture known as an autoencoder[1]. An autoencoder consists of two stages– an encoder and a decoder, each comprised of one or more neural network layers (Figure 4). In industrial settings, the encoder's input layer might ingest a continuous stream of quantitative data from equipment sensors (temperature, pressure, etc., shown as Xn in the figure) over time—for example, once each minute. This data is then fed to one or more 'hidden' layers where it is compressed, i.e., reduced in dimensionality, by virtue of using fewer nodes than in the input or output layers in what is commonly referred to as the bottleneck layer. In this way, the input data is encoded into an efficient, lower-dimensional representation known as the latent space representation.

In the decoder stage, the process is reversed and the latent space representation is decompressed and expanded back to the original input dimension at the output layer. All of this is achieved during the standard training phase used in deep learning where the neural network parameters (i.e. weights and bias terms) are determined through an iterative process known as backpropagation in which the model's parameters are continuously tuned until some predefined objective function has been sufficiently optimized. The objective is to reconstruct the original input data at the output layer of the autoencoder with as little difference as possible.

[1] *https://patents.justia.com/patent/11227236*

Figure 4—Simplified Autoencoder Architecture



In practice, achieving this outcome typically requires many thousands of iterations, with the weights[2] tweaked slightly with each iteration in response to measured differences between the input and resulting output layers. When the outputs (X'n in the diagram) have finally achieved parity (or as close to it as possible) with the original inputs (Xn), the model is said to have 'learned' the normal state necessary to deliver subsequent actionable alerts.

The important aspect of this learning methodology is the reduced dimension of the hidden layers versus the input and output layers. This is required in order for the model to learn. If the width (i.e. number of nodes) of the hidden layer was equal to the dimension of the input layer, then the model could simply "cheat" by applying a weight of 1 to every input and instantly recreating the desired output without having learned anything at all about the system or its operation.

Once the model parameters have converged during training, the NBM model is ready to evaluate new input data and draw conclusions about its adherence to the model's learned definition of normal behavior. This is done by using the same objective function employed during training. The reconstruction error is calculated from the output layer as new data is fed into the autoencoder's inputs. The greater the value of the reconstruction error, the more likely the current input data is indicative of an anomalous condition in the system.

Another way to understand the relationship between all variables in a data set is to imagine a vector (Figure 5) representing a point in multidimensional mathematical space that coincides with the weighted combination of all input data elements. Each such vector is a critical element that enables the model to draw its out-of-normal conclusions from the full set of iterating variables rather than from a single variable. In this example, there are only three variables shown, but, in fact, each of these vectors contains as many unique dimensions as there are variables in the input data stream, typically dozens, hundreds, or even thousands.

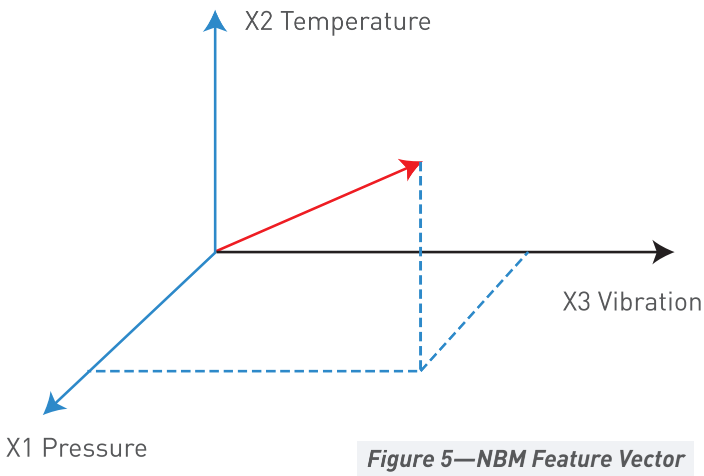
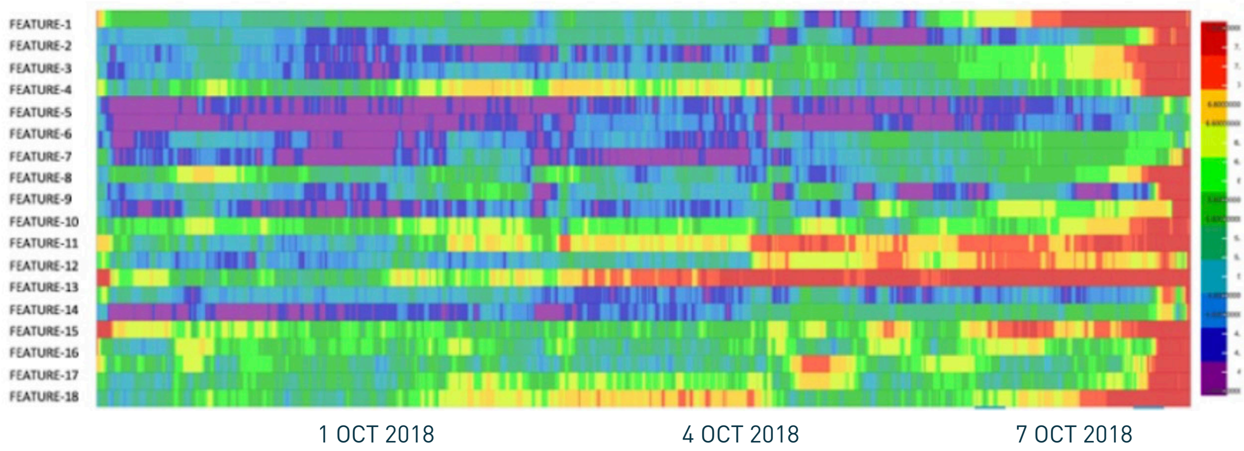


Figure 5—NBM Feature Vector

As these vectors are generated, their results can be visualized using a heat map as shown in Figure 6. The heat map aggregates the degree to which each feature is in or out of tolerance over a period of time (the horizontal axis), with red areas indicating the most out-of-tolerance and green the most in-tolerance. By reordering the feature rows of the heat map to show those with the greatest amount of red (out-of-tolerance) at the top, a snapshot is displayed of the system's status over the selected period of time, providing an intuitive explanation of the most likely drivers of an anomalous event.

[2] *Weight being a coefficient between 0 and 1 by which each node's input value (Xn) is multiplied.*

Figure 6—NBM Feature Heat Map



Evaluation, scoring, and alert generation

One of the key benefits of the NBM approach is that it can reduce many thousands of individual time-series data points to a single, intuitive metric that can be acted upon by SMEs and non-technical personnel. This single point is known as a risk (or anomaly) score. The risk score is the statistical summarization of the extent to which each feature vector value deviates from the mean value for that feature (Figure 7).

Determining whether or not maintenance action should be taken based on the anomaly score is a function of how sensitive we want the output to be. In the most basic version of this technique, we would assign a threshold value for each anomaly score and declare an alert anytime this value is reached. In reality, this approach is likely to make the model overly sensitive and generate more false positives (i.e., alerts to conditions that are, in fact, within tolerance and should not be acted upon) than we want.

Instead, it makes sense to decide in advance upon a statistical band of upper and lower bounds derived from experientially determined standard deviations. Thus, it is only when the anomaly score exceeds the upper bound of the band that we will declare the system to be out-of-normal and generate an alert. Limiting alert generation in this way can be achieved using a sequential probability ratio test (SPRT) or similar customized approach suitable to a given application domain based on prior institutional knowledge.

RETRAINING AND EVOLVING NORMALS

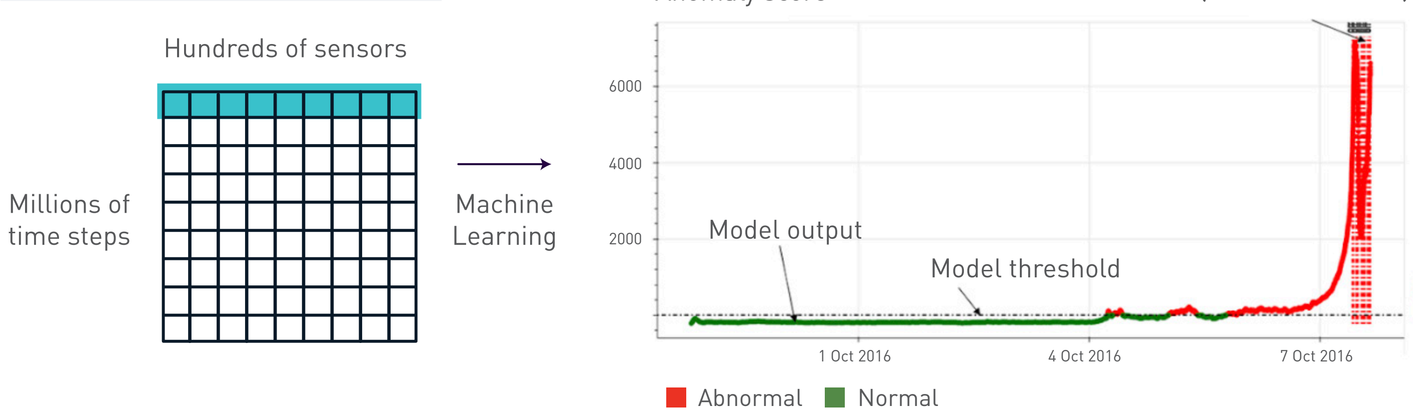
NBM systems, like all systems, evolve over time. And this evolution takes many forms. Equipment ages, maintenance occurs, tolerances change, desired outputs change, regulations change, and externalities like availability of time, people, and money change. As a result, our sense of what is 'normal' for our complex system is highly likely to change and our modeling approach needs the flexibility to adapt as circumstances evolve. Fortunately, NBM is uniquely well-positioned to respond to these inexorable changes, far more so than the CBM, SPC, and other methodologies discussed earlier.

The most straightforward way in which NBM enables this flexibility is by simply retraining it from time to time (monthly is a common choice) to reflect the latest reality. This process occurs in the same manner as initial training, except that results will be improved in subsequent iterations due to the availability of larger and more comprehensive data sets. In the parlance of our original description, it's important to build into the ongoing operation of the NBM process the maximizing of the congruence between the model's inputs and outputs.

EXPLAINABILITY

There is another extremely important element to consider when making decisions about NBM implementation, and that is explainability. One of AI's most persistent criticisms is that users are asked to

Figure 7—Risk Score and Alert Creation



blindly trust the system's outputs without understanding how they were derived. Tools like the heatmap depicted in Figure 6 are important enhancements to the base NBM approach in that they can help identify which specific components are deviating from their normal behavior and provide users with a deeper understanding of where a failure is likely to occur in a complex system. With such enhancements, NBM will not only provide advance warning that a system component is verging on failure but also provide insight into the source of the anomaly.

HUMAN-IN-THE-LOOP AND KNOWLEDGE MANAGEMENT

Many industrial organizations rely heavily on the expertise of key personnel and experienced subject matter experts to keep their operations running smoothly, especially in areas involving upkeep and maintenance of their most critical assets. However, in recent years, companies have faced the increased risk of labor and skill shortages driven by demographic shifts and the lingering impact of the global pandemic. Advanced technologies like artificial intelligence and machine learning provide new opportunities for organizations to elevate their best workers, helping to institutionalize and embed their knowledge into solutions like NBM-based predictive analytics in ways that augment specialists, enhance the rest of the workforce, and increase productivity.

For example, a traditional maintenance team often endures a constant stream of threshold-based alarms from their sensors and relies on the experience of a handful of seasoned operators to filter out the real alerts from the noise. These same SMEs play key roles in making NBM deployment and utilization successful, including identifying failure modes, establishing functional alert thresholds, and determining when model retraining should be undertaken. The most effective NBM solutions employ such human-in-the-loop learning techniques at their core, leveraging the domain knowledge of SMEs to improve underlying model performance through an intentional feedback loop between humans and machines. Because their knowledge is now integrated into the deployed NBM solution—freeing them from the tedious aspects of predictive maintenance activities now handled by the automated solution—they can employ their skills on higher-value activities instead.

There is no replacement for specialized domain expertise, regardless of how much automation a company implements. NBM is, though, a uniquely valuable force multiplier of the skill sets of experienced SMEs, augmenting the workforce and increasing operational productivity at a time when labor shortages and skill gaps pose increasing risks to organizations.

USE CASES

As previously described, the principal purpose of NBM is to define the normal state of a complex system and to then proactively identify and flag instances in which that system is operating outside of normal. Ideally, such identification and flagging will occur with sufficient advance warning to allow maintenance or repair actions to take place that will forestall an outright system failure and all of the revenue loss, repair costs, and safety compromises that typically come with such failures.

There are many examples of complex systems to which NBM techniques can be applied, some of which are physical and others more process-oriented.

- **Production equipment on oil platforms—**Failure can mean millions in lost revenue due to deferred production, safety risks, and environmental catastrophes. By modeling equipment temperatures, pressures, and rotation and flow rates, incipient problems can be identified early, saving upstream operators millions of dollars and significant regulatory exposure.

- **Manufacturing plants—**Out-of-normal operations in manufacturing plants can result in safety hazards, environmental violations, and inferior quality in the products being produced. Proactively identifying process and equipment problems help ensure profitable operations in frequently low-margin businesses.

- **Commercial and military aviation—**Jet engines and other complex airborne hardware are routinely subject to enormous operational stresses. Small problems can quickly cascade into expensive and dangerous situations, risking lives and the loss of immense capital investments.

- **Electric power generation, transmission, and distribution—**Electric power generation equipment, whether renewable or traditional fossil-fuel-powered, requires extremely high-reliability performance, making these assets ideal candidates for NBM-based predictive maintenance methodologies.

- **Financial investments—**The normal ebb and flow of global equity and debt markets occasionally undergo upsets that can produce short-lived investment opportunities or risks that must be quickly and actively mitigated. In an industry characterized by millisecond transaction speeds, knowing about these threats and opportunities before the competition can be the difference between success and failure.

SUMMARY AND CONCLUSIONS

Normal behavior modeling is the state of the art in predictive maintenance of complex systems and equipment. It simultaneously automates complicated performance data monitoring and analysis processes while minimizing alert fatigue from false positives. It facilitates the continuing adaptation of the monitoring system to the evolving notions of what constitutes the 'normal' state of the system as it ages. And it enables alerts to be based on the complex and frequently nonobvious interactions between the many components and parameters within (and sometimes outside of) the system.

Many factors go into successfully developing and implementing an NBM system. These have been discussed throughout this paper, and include:

- Sensor data availability/quality/frequency/features
- Understanding of how 'normal' evolves with equipment age and changing operational practices
- Alert explainability and knowledge management
- Human-in-the-loop learning and workforce augmentation

The concepts described in this paper are intended to give the reader an initial understanding of NBM's capabilities, benefits, and the steps required to make it work in an organization that operates and maintains complex systems.

ABOUT SPARKCOGNITION

SparkCognition's award-winning AI solutions allow organizations to predict future outcomes, optimize processes, and prevent cyberattacks. We partner with the world's industry leaders to analyze, optimize, and learn from data, augment human intelligence, drive profitable growth, and achieve operational excellence. Our patented AI, machine learning, natural language, and visual analysis technologies lead the industry in innovation and accelerate digital transformation. Our scalable software solutions, customizable end-to-end services, and in-depth industry expertise allow organizations to solve critical challenges—prevent unexpected downtime, maximize asset performance, and ensure worker safety and regulatory compliance while optimizing workflows and avoiding zero-day cyberattacks on essential IT and OT infrastructure. Contact us to discuss how SparkCognition Normal Behavior Model technology can unlock the power in your data at info@sparkcognition.com.

APPENDIX—GLOSSARY OF TERMS

Autoencoder (aka neural network encoder)—A neural network that is trained to attempt to copy its input to its output by repeatedly assigning weights in the hidden/latent layer to the inputs and then recursively adjusting those weights until the desired output has been achieved.

Bottleneck—The hidden/latent layer of a neural network that creates in the output layer a representation of the initial input data. The bottleneck layer typically contains fewer nodes than the input or output layers, facilitating the reduction of dimensionality in the input data stream.

Condition-based monitoring—A predictive maintenance technique that continuously monitors the condition of equipment or assets using sensor-derived data that relates information about real-time conditions.

Decoder—The layer of an autoencoder that delivers the output data set after applying the weights developed in the bottleneck or hidden/latent layer.

Dimensionality reduction—Technique employed by an autoencoder's hidden/latent layer to reduce the number of large/complex input features of input data. This technique can better fit the model with less risk of overfitting.

Encoder—The layer of an autoencoder that ingests the input data set from system sensors prior to applying the weights developed in the bottleneck or hidden/latent layer.

Feature—A unique, nonredundant, and measurable (usually numeric, but not necessarily) property of a system that is derived from a set of weighted input data. Note: a feature can be either a unique/native characteristic of the raw input data or the result of combining two or more raw data inputs.

Feature extraction—The process by which unique features are extracted from an initial data set, thus reducing the overall amount of data while providing nonredundant data elements. The process is important to reduce the amount of storage and processing required for subsequent analysis and also to reduce the likelihood of overfitting the model.

Feature vector—Multidimensional mathematical representation of a specific output feature that has had model weights applied to it.

Hidden/latent layer—The central layer of an encoder in which weights are repeatedly applied to input data in an effort to force the output set to match the input data set.

Physics-based modeling—Method of modeling/simulating the operation of a system or piece of equipment by defining all of its characteristics using a series of mathematical equations.

Principal component analysis—An unsupervised statistical learning technique in which underlying patterns are identified in a data set so that it can be expressed in terms of another data set with fewer variables and with reduced dimensionality and complexity but without significant loss of information.

Risk or anomaly Score—Numerical value derived by aggregating all feature output values from the NBM model through statistical analysis. The risk score determines whether or not action is required on the part of maintenance staff.

Tag—The specific name assigned to a unique data element in an input data set to a neural network (e.g., Temp_Pump 37A).

Normal behavior model—An AI-enabled modeling technique in which machine learning is applied to a time series of operational data to identify the characteristics of the data in normal operation.

Supervised learning—NBM training methodology in which known failure modes are included in initial data sets along with the data that preceded these failures.

Unsupervised learning—NBM training methodology in which only normal operating data are included in the initial training data set.

REFERENCES

Brownlee, J. (2020). Autoencoder Feature Extraction for Classification. www.machinelearningmastery.com.

Tiu, E. (2020). Understanding Latent Space in Machine Learning. www.towardsdatascience.com.

Normal Behavior Models Using Autoencoders (2020), https://ebrary.net/194499/engineering/normal_behavior_model_using_autoencoders

Contact SparkCognition today at info@sparkcognition.com