SAP Predictive Maintenance and Service

Data Science and Machine Learning in the Internet of Things and Predictive Maintenance

Contents

Data Science Group
IoT Predictive Maintenance, Products and Innovation, SAP
April 2017

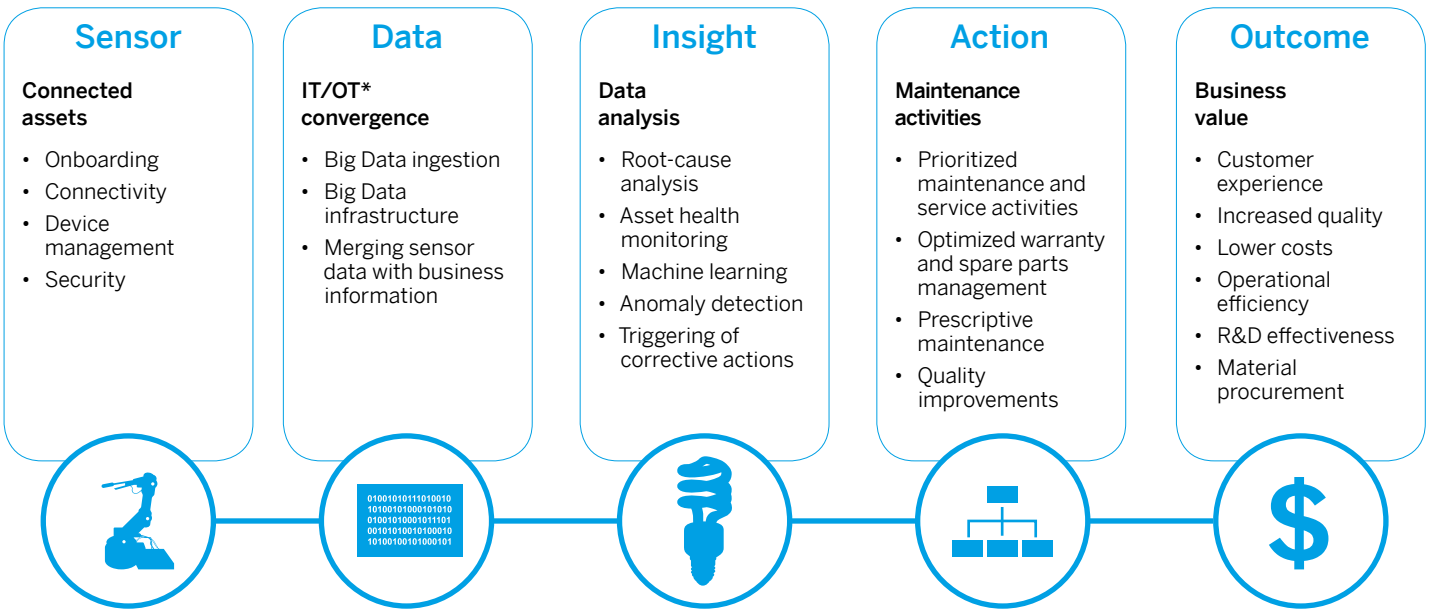
1. IoT and Predictive Maintenance

This paper describes how data science and machine learning are used in the Internet of Things (IoT) and specifically in the domain of predictive maintenance. It is based on many customer projects conducted by SAP's Data Science group in IoT and predictive maintenance. Data science faces many new challenges in the domain of IoT while at the same time, the traditional challenges have not gone away. We describe the architecture and data science components for the SAP® Predictive Maintenance and Service solution, detail examples of the application of data science to IoT and predictive maintenance, and conclude with customer case studies.

The statistics are staggering – our future will involve billions of connected "things," generating trillions of gigabytes of data, in a market of trillions of dollars.

The biggest challenge for the IoT and the "digital enterprise" is turning these huge volumes of data into information and, from that, performing analyses to improve business processes – "from sensor to insight to action," as shown in Figure 1.

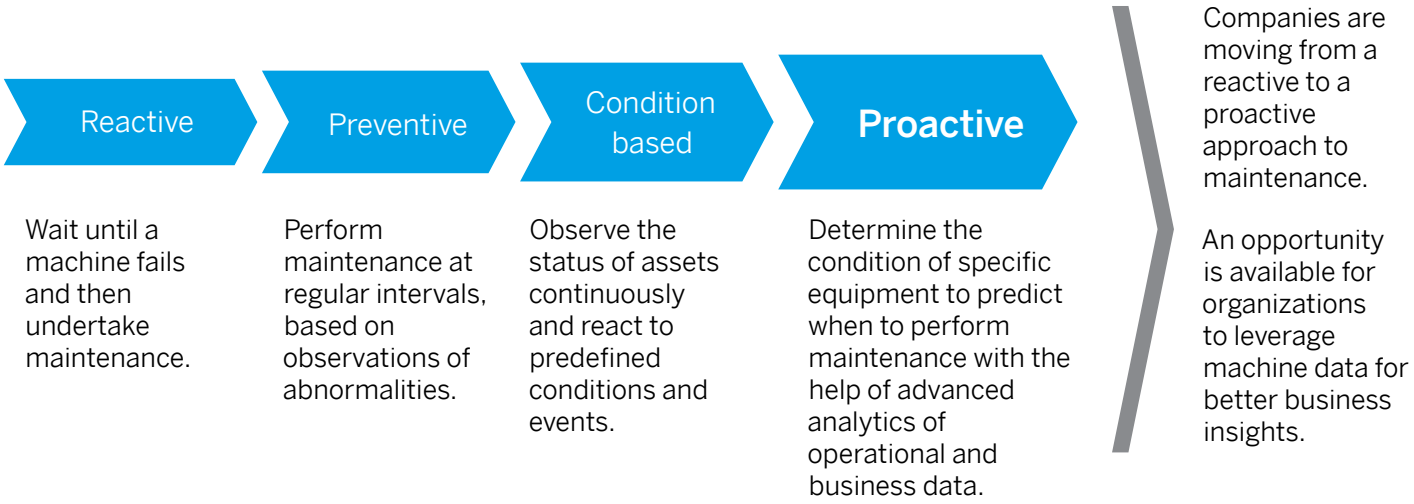
Figure 1: From Sensor to Insight to Outcome



* OT = operational technology

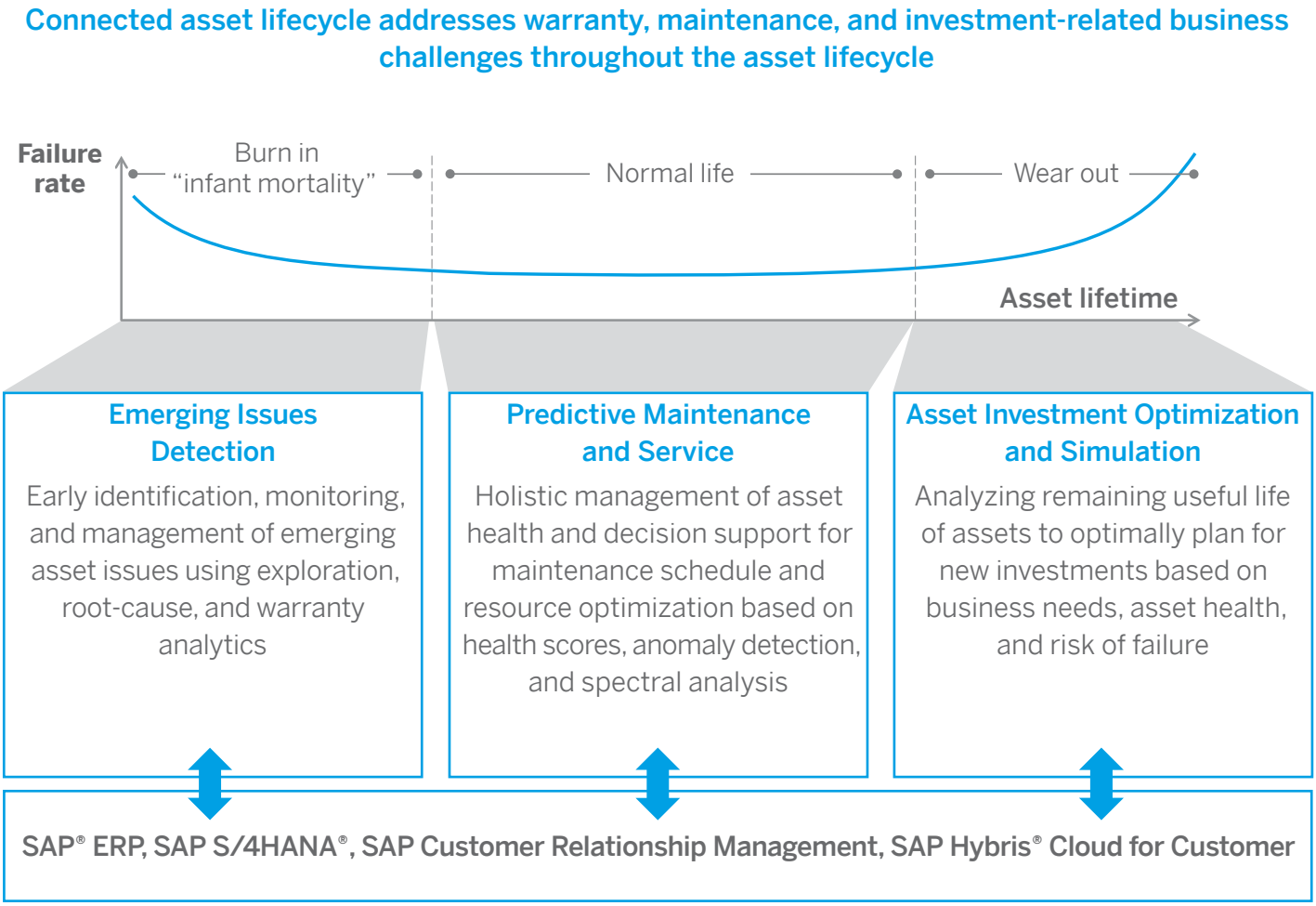
IoT is clearly cross-industry, and across many of those industries, predictive maintenance is a common theme. It is predictive in the sense of changing from a reactive to a proactive approach to maintenance (see Figure 2) – from a rear-view mirror approach to a forward-looking approach.

Figure 2: From a Reactive to Proactive Approach to Maintenance



Predictive maintenance is part of a connected asset's lifecycle that broadly comprises three phases – warranty, maintenance, and investment – as shown in Figure 3. The warranty phase is concerned with the early identification of any emerging issues in the use of the asset and is referred to as emerging issues detection. The main phase in an asset's lifecycle is its "normal life," during which predictive maintenance is performed. This comprises advanced analysis of operational and business data to determine the condition of specific assets in order to predict when to perform maintenance. The third phase of an asset's lifecycle is to optimally plan the development and investment in new and replacement assets.

Figure 3: Connected Asset Lifecycle Management

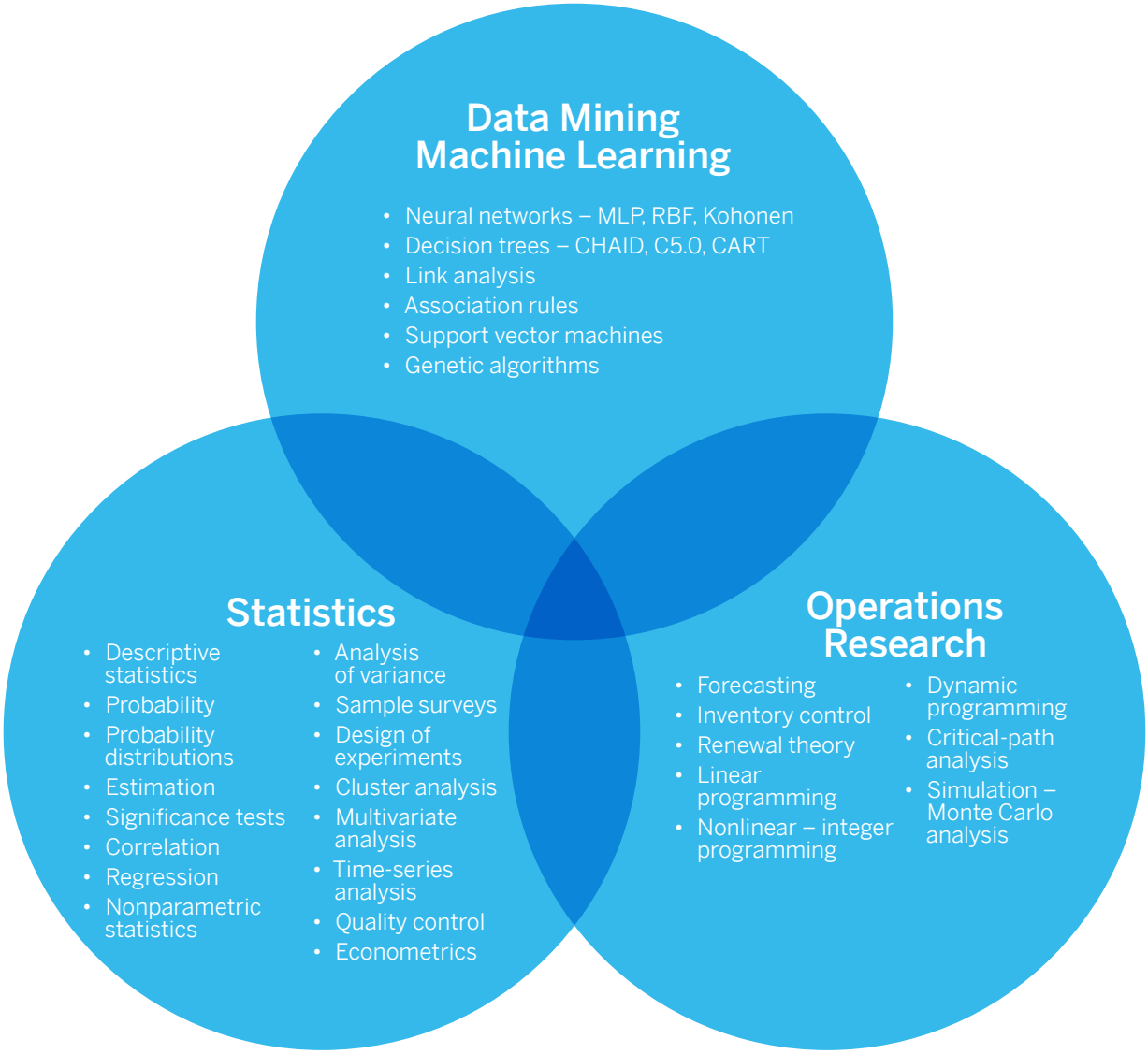


The benefits of the proactive approach to maintenance in the manufacturing industry include a reduction in maintenance costs of factory equipment, a reduction in equipment downtime, and a reduction of capital investment by extending the useful life of machinery. To achieve this requires advanced analytics that come from the discipline of data science.

2. Data Science and SAP's Data Science Process for IoT

Data science is an interdisciplinary field concerned with extracting knowledge or insights from data and is an inclusive term for quantitative methods, such as statistics, operations research, data mining, and machine learning, as shown in Figure 4, along with the main analyses of each discipline. Synonymous terms include knowledge discovery and predictive analytics. Lively discussions exist on the definition and whether one is a subset of another or a superset of others, or whether one discipline is more important than another, more modern, more useful, and so on. However, the key point is that they all have the same objective – the improvement of business processes and, more generally, any process through quantitative analysis.

Figure 4: The Interdisciplinary Nature of Data Science



Data science is not a new subject: Operations research dates back more than 50 years, statistics even further. The term "data mining" appeared more than 25 years ago and likewise for machine learning. What is new is the impact that IoT has had on data science through the vast quantities of data available for analysis, both in terms of the number of data records or observations and the number of variables. The huge increase in data volumes comes from the huge increase in "things" creating data and the refined granularity of the data being produced.

Data science and machine learning have had to address the challenges of IoT and specifically within the context of predictive maintenance:

- Anomaly detection in very large data sets
- The analysis of very rare events
- Multivariate analysis of huge numbers of observations and variables
- Application to streaming data
- Visualization of very large data volumes
- The greater inclusion of business domain expertise

These are significant challenges. Finding anomalies in very large data sets is akin to "finding a needle in a haystack." Analyzing very rare events is by definition very difficult, as there is very little data about the rare event – a scatter plot of a million data points will most likely resemble a solid rectangle.

Although data science has had to adapt to address the challenges of IoT, much of the discipline and challenges remain. Is the data relevant? Is it of good quality? Does the model produced by an algorithm translate from a mathematical relationship to a causal one? These are some of the questions that need to be answered.

Data science has a huge range of applications; however, they can be broadly classified into five groups of analysis that the applications are addressing, as shown in Figure 5, along with examples given within the predictive maintenance domain.

Data science faces many new challenges in the domain of IoT while at the same time the traditional challenges have not gone away.

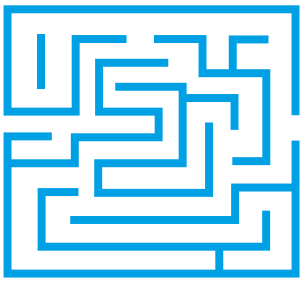
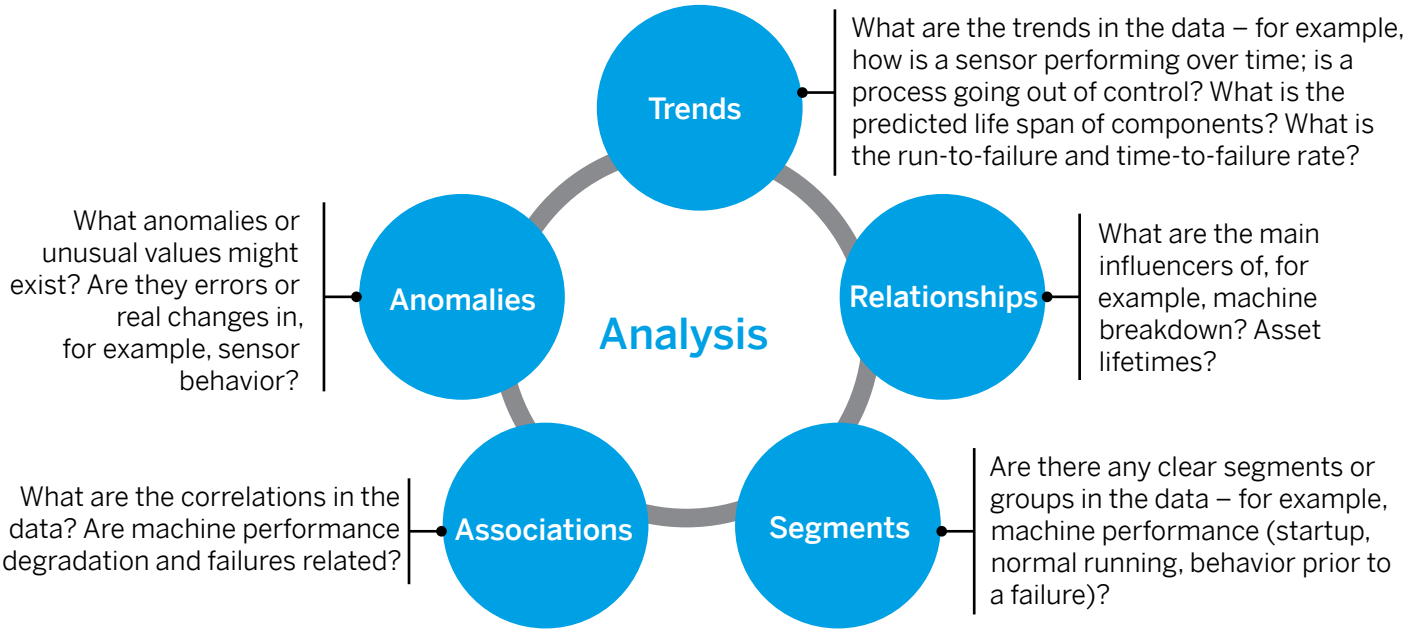


Figure 5: Classification of Data Science Analyses with Examples in Predictive Maintenance



The main applications of data science, both in IoT and in general, are concerned with "relationships" in which we aim to build a model to define the relationships between inputs and outputs. The output (in statistics it is referred to as the dependent variable) is a function of one or more inputs (the independent variables). We use known inputs and outputs to create a model and then use the model to predict or "classify" or "score" unknown values. Data science in IoT and predictive maintenance also focuses on anomaly detection, as it is the unusual or unexpected that can be a precursor to failure.

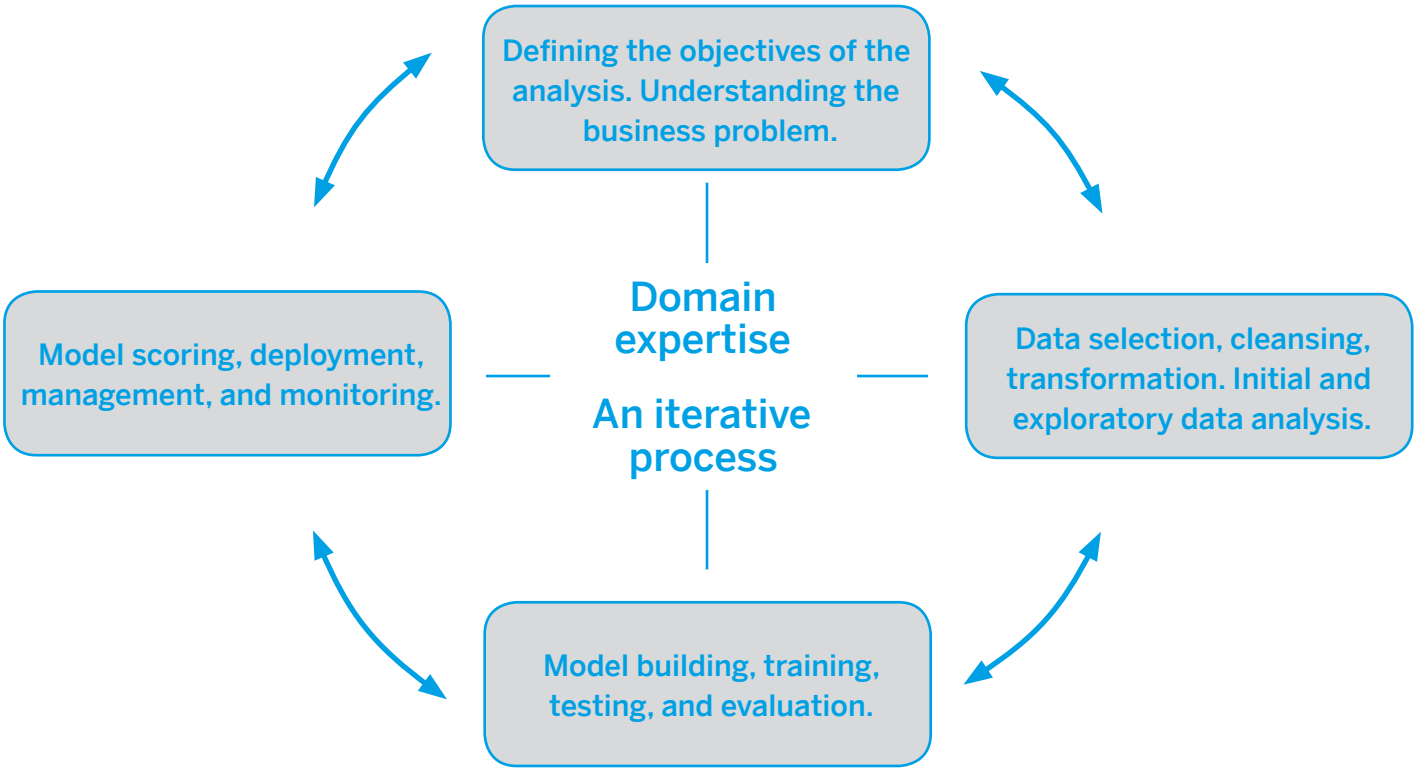
This broad classification expands within the domain of predictive maintenance to include simulation and optimization plus unstructured data analysis, and mining. Other dimensions to this classification for later discussion are batch and streaming data analysis, and automated and nonautomated analysis.

Algorithms are an important part of data science, but, more importantly, they are part of a process for analysis. Over the years there have been several standards proposed for this process with the main one being known as CRISP-DM (Cross-Industry Standard Process for Data Mining). However, our experience in predictive maintenance projects – and IoT in general – has led us to expand on the standard and add two stages to the process we believe are missing, namely, more explicit involvement of business domain expertise and ongoing monitoring when deploying the results of the analysis in business processes. The latter seems to be a conspicuous omission from the standard although in practice most likely done.

Combining domain expertise with data science expertise is especially important for rare event classification problems that are common in predictive maintenance. In these application areas, given that they concern rare events, there is by definition very few data points upon which to base a model. Building useful models in this area is therefore quite difficult. Given the paucity of data then, it is even more important to include the expertise of domain experts and actively include them in all stages of the predictive analysis process. Sometimes it can be a challenge to identify them; however, they should be involved in all stages of the process and, in particular (although not usually done), in the stage of model development and the various approaches to model feature selection and interactive model development. Figure 6 summarizes SAP's process for data science that we have developed though many IoT predictive projects.

IoT is clearly cross-industry, and across many of those industries, predictive maintenance is a common theme.

Figure 6: SAP's Data Science Process for IoT and Predictive Maintenance



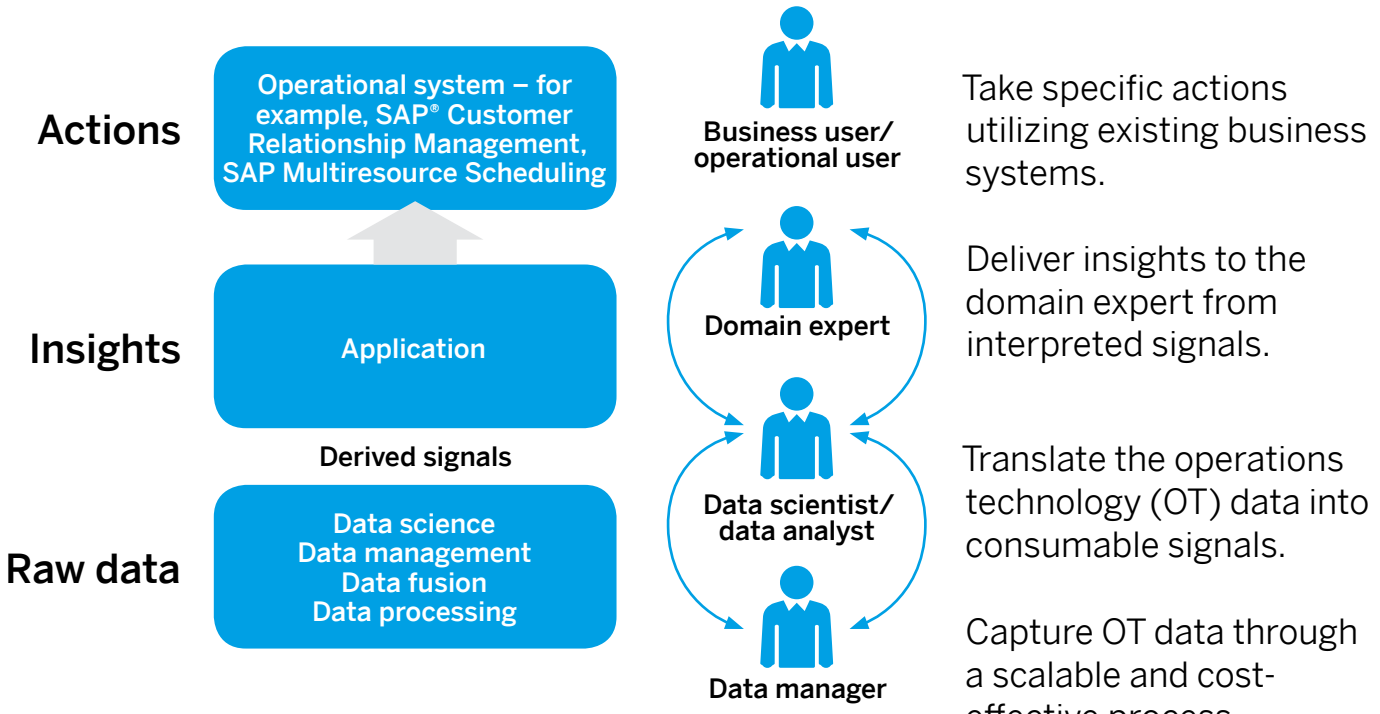
It is important not to simply equate data science with algorithms. The major activity in the data science process is spent on identifying, accessing, and preparing data for analysis. Of course algorithms are important; however, without relevant and quality data, they will provide little insight. Algorithm accuracy is important but so is understanding and clarity of the model. The objective is to improve business processes; however, if the decision makers do not understand why the analysis is suggesting a change, they are unlikely to act upon it.

3. The Data Science Requirements in the IoT and Predictive Maintenance Domain

Requirements are driven by the user personas, the data sources and their characteristics, and the required analyses and their presentation. In Figure 1 we saw the process of going from sensor to insight to action; in Figure 7 we can see the various user personas associated with each step.

Clearly there is a significant role for the data scientist, but it also is important to include data analysts – recently also referred to as citizen data scientists – to reflect a less specialized analyst but one with quantitative skills. Also, as stated earlier, the process must include domain experts.

Figure 7: User Personas in the IoT and Predictive Maintenance



3.1. PREDICTIVE ENGINES

To meet the requirements of data scientists, we need to provide a very comprehensive range of analyses and algorithms – but, at the same time, ones that scale to address the huge volumes of data. We can do this by enabling the most comprehensive collection of analyses and algorithms available by using the R integration for the SAP HANA® platform, thereby giving access to the incredibly comprehensive R open source algorithm library and, furthermore, its extensive data visualizations. For high performance, we have developed the in-database predictive analysis library in SAP HANA.

3.1.1. The R Language for Statistical Computation and Graphics

The R programming language has grown in popularity among data scientists for developing analytic applications and is widely used for advanced data analysis. The integration of the SAP HANA database with R enables the embedding of R code in the SAP HANA database context,[1] whereby the SAP HANA database allows R code to be processed in-line as part of the overall query execution plan. This scenario is suitable when an SAP HANA–based modeling and consumption application wants to use the R environment for specific data science functions. An efficient data exchange mechanism supports the transfer of intermediate database tables directly into the vector-oriented data structures of R. This offers a performance advantage compared to standard SQL interfaces, which are tuple based and therefore require an additional data copy on the R side.

To process R code in the context of the SAP HANA database, the R code is embedded in SAP HANA SQL code in the form of an RLANG procedure. The SAP HANA database uses the R environment to execute this R code, which allows the application developer to elegantly embed R function definitions and calls within SQLScript and submit the entire code as part of a query to the database. A key benefit of having the overall control flow situated on the database side is that the database execution plans are inherently parallel and, therefore, multiple R processes can be triggered to run in parallel without having to worry about parallel execution within a single R process.

3.1.2. Predictive Analysis Library in SAP HANA

If model training times are critical, then in-database algorithms are key, thereby avoiding external data transfers and benefiting from in-memory parallel processing. For this we provide the predictive analysis library (PAL).[2] The library defines more than 70 functions that can be called within SAP HANA SQLScript procedures to execute analytic algorithms. Figure 8 lists the algorithms grouped by the traditional data science categories. From a predictive maintenance perspective, specific algorithms are included for survival analysis, probability distributions, classification and regression, cluster analysis, and time-series analysis. The PAL functions support comprehensive parameter selection – for example, agglomerative hierarchical clustering supports 10 different distance measures and 7 different clustering methods; decision trees support includes split and merge thresholds, parent and leaf number control, maximum depth and branch, pruning, discretization, cross-validation, number of trees, JavaScript Object Notation (JSON), and Predictive Model Markup Language (PMML) output.

Figure 8: Data Science Algorithms in the Predictive Analysis Library

Association analysis
- Apriori
- Apriori Lite
- FP-Growth
- KORD – tTop K rule discovery

Classification analysis
- CART
- C4.5 decision tree analysis
- CHAID decision tree analysis
- K-nearest neighbour
- Logistic regression
- Neural network
- Naïve Bayes
- Random forest
- Support vector machine
- Confusion matrix
- Area under curve (AUC)
- Parameter selection/model evaluation

Regression
- Multiple linear regression
- Polynomial regression
- Exponential regression
- Bivariate geometric regression
- Bivariate logarithmic regression

Cluster analysis
- ABC classification
- DBSCAN
- K-Means
- K-medoid clustering
- K-medians
- Kohonen self-organized maps
- Agglomerate hierarchical
- Affinity propagation
- Latent Dirichlet allocation (LDA)
- Gaussian mixture model (GMM)
- Cluster assignment

Time-series analysis
- Single exponential smoothing
- Double exponential smoothing
- Triple exponential smoothing
- Forecast smoothing
- ARIMA/seasonal ARIMA
- Brown's exponential smoothing
- Croston method
- Linear regression with damped trend and seasonal adjust
- Forecast accuracy measures
- Test for white noise, trend, seasonality

Probability distribution
- Distribution fit /Weibull analysis
- Cumulative distribution function
- Kaplan-Meier survival analysis
- Quantile function

Outlier detection
- Interquartile range test (Tukey's Test)
- Variance test
- Anomaly detection
- Grubbs outlier test

Link prediction
- Common neighbours
- Jaccard's coefficient
- Adamic/Adar
- Katzβ

Data preparation
- Sampling
- Random distribution sampling
- Binning
- Scaling
- Partitioning
- Principal component analysis (PCA)

Statistic functions (univariate)
- Mean, median, variance, standard deviation
- Kurtosis
- Skewness

Statistic functions (multivariate)
- Covariance matrix
- Pearson correlations matrix
- Chi-squared tests
- Test of quality of fit
- Test ofi
- F-test (variance equal test)

Other
- Weighted scores table
- Substitute missing values

3.1.3. Streaming Analytics with SAP HANA Smart Data Streaming and the PAL

Real-time event-stream processing is an integral part of IoT and therefore so are streaming analytics. The PAL includes incremental data science and machine-learning algorithms that learn and update continuously to provide dynamic predictions. Event-stream processing is about reacting to event-driven data in real time as the events happen. Historical data may be of less significance than current data for near-time, "immediate future" prediction. Expected behavior may shift or drift over time. Deploying machine-learning algorithms within SAP HANA smart data streaming provides the ability to immediately incorporate current data into algorithms rather than periodically polling an external data source. This lets companies instantly and progressively adapt to changing conditions and behaviors using machine-learning algorithms that are designed for continuous analytics with low latency.

The PAL includes DenStream, an incremental clustering algorithm that uses the concept of microclusters to summarize clusters of arbitrary shapes and an elaborated pruning technique to detect outliers. It is insensitive to noise, and its novel pruning technique leads to better memory management of streaming data. The PAL also includes Adaptive Hoeffding Tree, an incremental decision tree algorithm that uses limited samples to choose the best tree node-splitting attribute. It learns a tree-like graph from historical data to model the decision rules and map an observation to its target value, detecting concept drift and updating the tree model automatically. It has the advantages of low overfitting with no need of pruning, low variance with stable decisions with statistical support, and low resource utilization, using limited hardware resources.

Example use cases from the SAP Labs network include:

- Using temperature sensor data to identify abnormal values based on the DenStream algorithm
- Using the accelerator gyroscope on mobile devices to detect different states
- Monitoring steering wheel rotation and driver bio-info to detect if the driver is drowsy

3.1.4. Automated Data Science with the Automated Predictive Library in SAP HANA

To meet the requirements of data analysts, we need to provide a more automated approach to analysis, one that does not require the analyst to have to know which algorithm to choose and its parameter settings. For this we provide the automated predictive library (APL). The APL provides an automated approach for each of the main classes of application, namely, classification, regression, cluster analysis, association analysis, and time-series analysis. The APL supports the data analyst by automating the tasks of managing missing data, identifying outliers, identifying correlations between the independent variables, looking for leak variables, optimally binning data relative to the target variables. The APL simply presents to the end user the key influencers of a target variable and can efficiently perform on extremely wide data sets – 100K plus.

All the automated components build several models internally and choose the most robust models based on their predictive power and predictive confidence. Predictive confidence corresponds to the proportion of information contained in the target variable that the explanatory variables are able to explain. Predictive power is the capacity of the model to achieve the same performance when it is applied to a new data set exhibiting the same characteristics as the training data set. This is the generalization ability or robustness of the model.

The APL is also available in SAP HANA Cloud Platform, enabling any application built on the platform to embed the automated analytics capabilities through a standard Web-based development model using RESTful Web services. SAP HANA Cloud Platform predictive services supports automated time series analysis, key influencers, outlier detection, what-if analysis, and scoring.

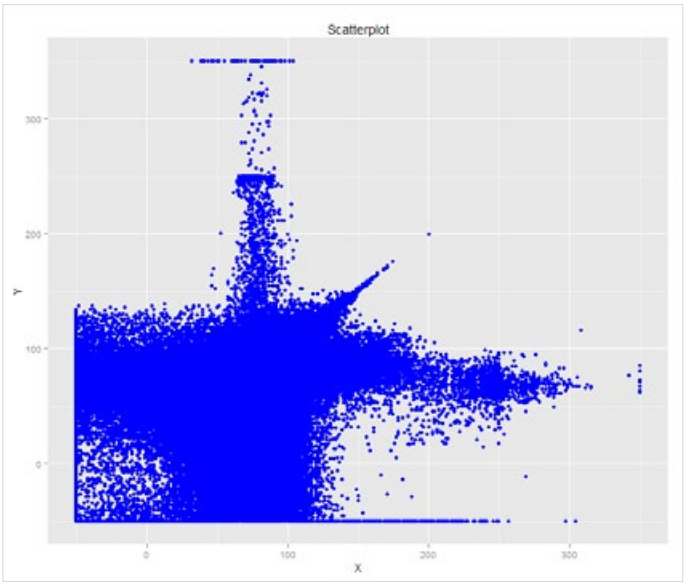
3.1.5. Comprehensive and Scalable, Automatic and Expert

SAP software uniquely meets the requirements of comprehensiveness and performance for the data scientist plus automated data science for the data analyst through the R integration, the PAL, and the APL. These are all integrated into SAP HANA and are all executable from development environments, such as the SAP HANA studio and the application function modeling add-on for SAP HANA, but also very simply from the graphical user interface of SAP BusinessObjects™ Predictive Analytics software.

3.2. DATA VISUALIZATION

Large data-volume data visualization is a challenge. A scatter plot of a million data points may appear as a solid rectangle. We need visual interactivity and smarter representations of numerous data

Figure 9: From Scatter Plot to Hexbin Plot



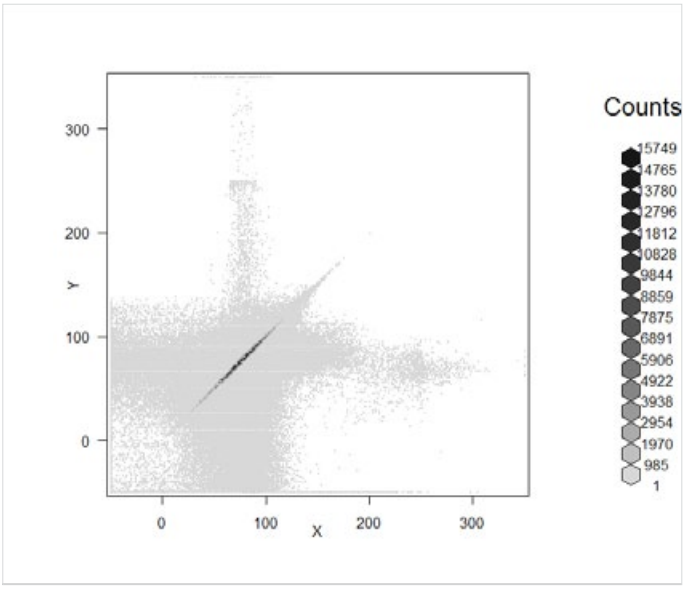
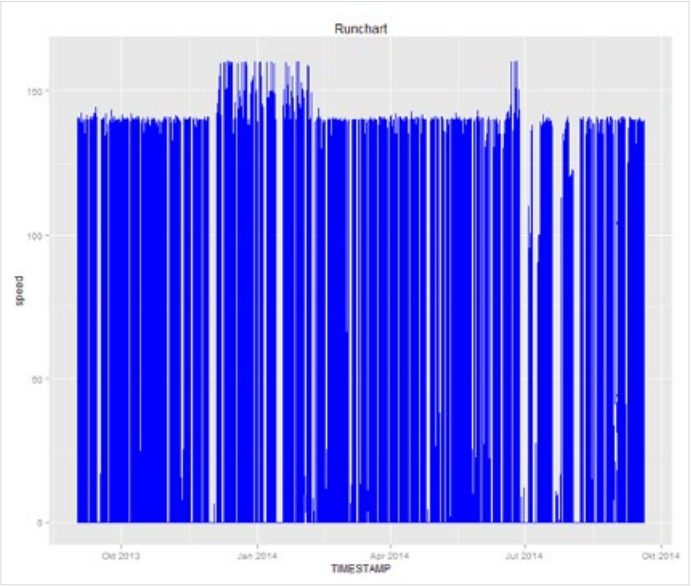
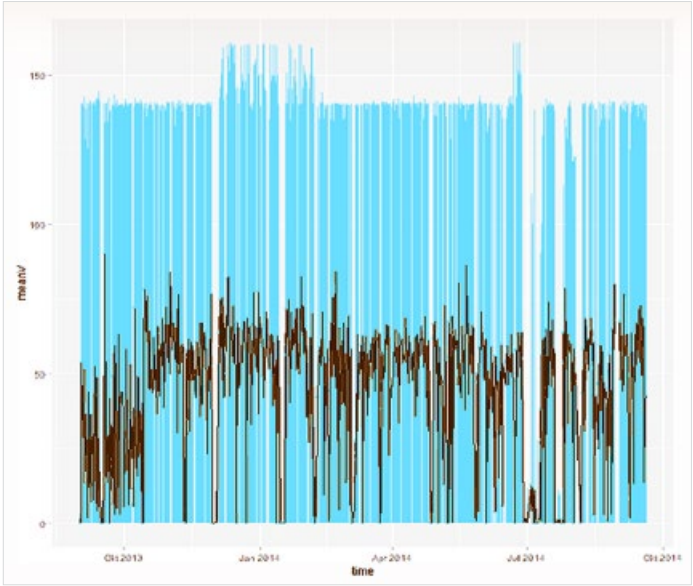


Figure 10 shows a run chart of sensor data that fluctuates between 0 and 150, but patterns of fluctuation are not discernible. By summarizing the values for specific time intervals in the form of a modified box plot, we can show the minimum, maximum, and mean values as a vertical line with markers for these values. This is done for the second chart in Figure 10, with the black line resembling a trend line making the sensor performance more visible.

Figure 10: From Run Chart to Modified Box Plot



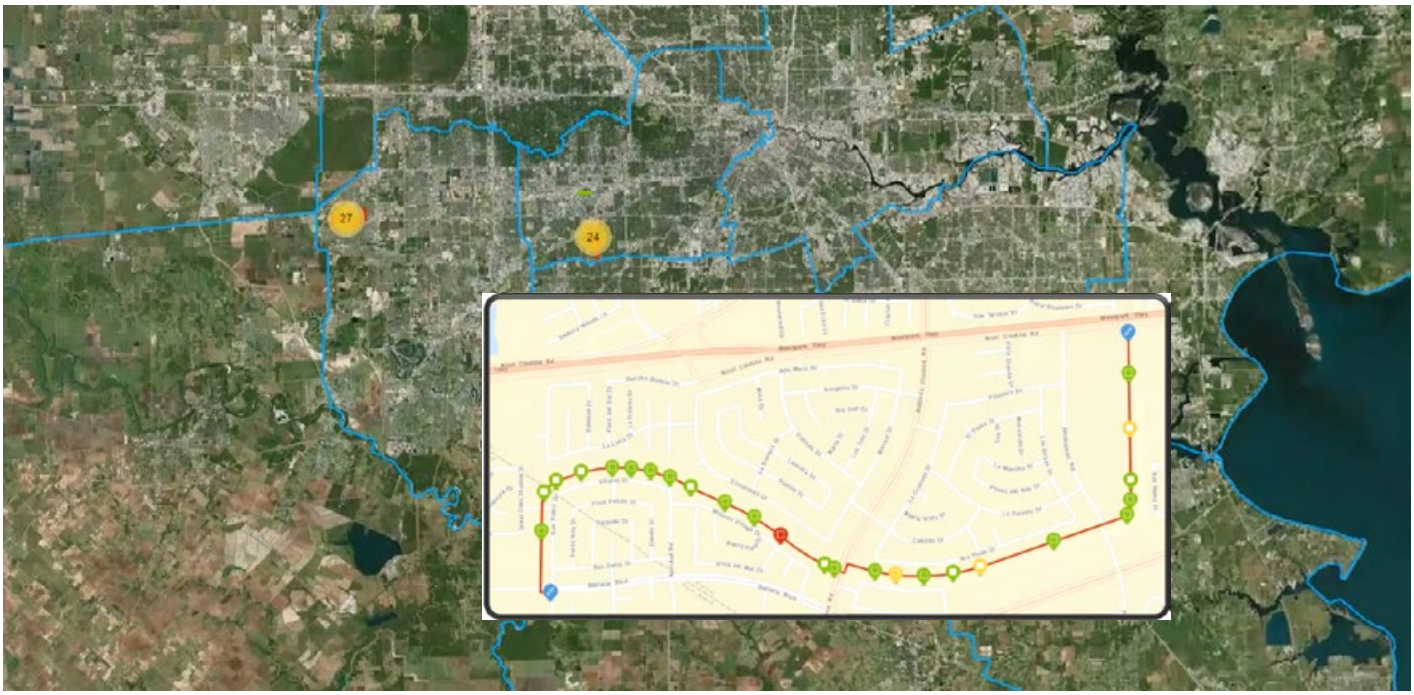


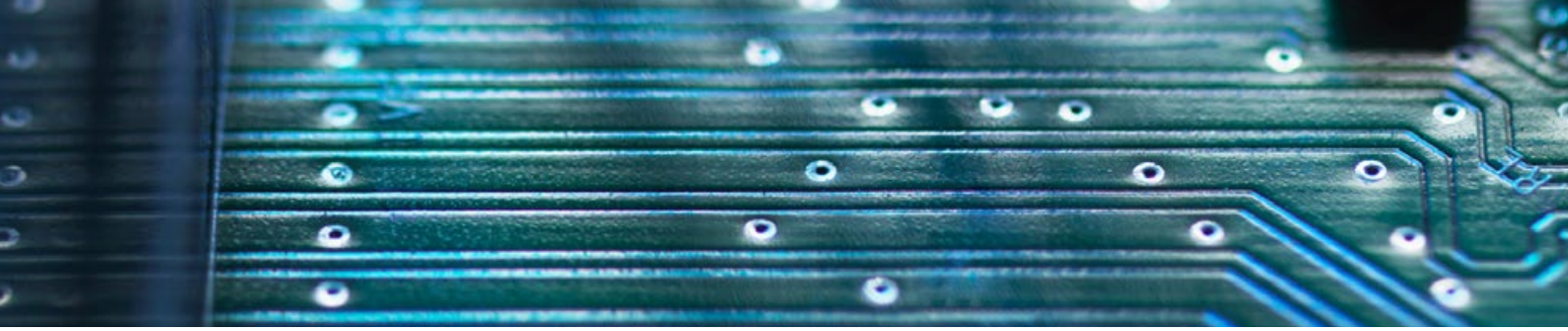
3.3. GEOSPATIAL DATA ANALYSIS

Our natural understanding of our world is through spatial analysis – mapping where things are and seeing how they relate. SAP HANA includes a multilayered spatial engine supporting spatial columns, spatial access methods, and spatial reference systems, to deliver high performance and results in everything from modeling and storage, to analysis and presentation of spatial data. With these enhanced geographical information system features, SAP HANA provides a common database for both business and spatial data. The spatial edition of SAP HANA includes spatial clustering using the algorithms – grid, k means, and DBSCAN (density-based spatial clustering of applications with noise). Spatial clustering can be performed on a set of geospatial points in SAP HANA.

Applications are numerous. A specific example is shown in Figure 11 in which a utility company uses the spatial edition of SAP HANA within the SAP Predictive Maintenance and Service solution to navigate through performance data to identify equipment in need of attention. The solution supports detailed information for displayed assets through tool tip, support for map overlays that can be toggled individually, color coding of displayed assets, and geofencing and selection of geofenced assets. It is also map-provider agnostic.

Figure 11: Spatial Analysis within SAP® Predictive Maintenance and Service





3.4. SERIES DATA PROCESSING

When monitoring machine efficiency, energy consumption, or network flow, the ability to monitor data over time enables you to investigate and act on patterns in the series data. SAP HANA supports series data processing to enable efficient processing of large volumes of series data in conjunction with business data to assess business impact. This is critical functionality for IoT and predictive maintenance applications in which series data volumes are huge.

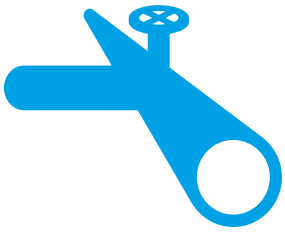
3.5. UNSTRUCTURED DATA ANALYSIS

Data science is mainly associated with structured data analysis – in other words, the analysis of data with a structure to it, usually in the form of variables or columns, by records or rows. However, there is a huge amount of data in unstructured formats, such as documents, e-mails, and blogs, which is generally textual, and hence the term "text analysis" is used when trying to analyze this unstructured content. It is said that up to 80% to 90% of enterprise-relevant information originates in unstructured data residing inside or outside an organization, such as in blogs, forum postings, social media, Wikis, e-mails, contact-center notes, surveys, service entries, warranty claims, and so on. The list is almost endless. The challenge, as with data mining, is to extract useful information.

SAP HANA supports text-search, text-analysis, and text-mining functionality for unstructured text sources. It supports full-text and fuzzy search using a full-text index to preprocess text linguistically, using techniques such as normalization, tokenization, word stemming, and part-of-speech tagging.

When text is processed, SAP HANA applies linguistic markup to give structure to core entities discovered in the text, such as people, organizations, dates, machines, sensors, and so on. It classifies relationships among entities for sentiment analysis. Semantic determinations about the overall content of documents relative to other documents can be extracted, and text-mining functions can be used to build a text repository, identify key terms, or categorize documents. Text mining supports categorization using the kNN classifier and supports various statistical analysis in the related and relevant document and term functions, namely, correlation matrixes, principal component analysis, and hierarchical clustering.

Practical applications in the IoT and predictive maintenance domain include text analysis and text mining to extract the topic of repair reports or warranty claims, analyzing telematics data and relating it to equipment service and warranty data, and using text analysis to understand maintenance activities.



Predictive maintenance is part of a connected asset's lifecycle that broadly comprises three phases – warranty, maintenance, and investment.

3.6. SIMULATION – DETERMINISTIC AND PROBABILISTIC, AND OPTIMIZATION

Predictive maintenance is part of overall connected asset lifecycle management, which addresses warranty, maintenance, and investment-related business challenges throughout the asset lifecycle, as was shown in Figure 3. In the latter stages of an asset's lifetime, new investments need to be considered and, as such, can be evaluated using simulation and optimization techniques.

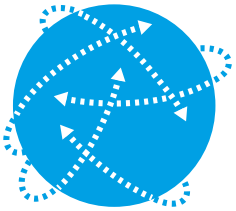
Simulation can take the form of deterministic modeling, whereby specific data values are used to model processes or operations and sensitivity analysis or what-if analysis is used to explore the inherent uncertainty in the data. Simulation in the form of probabilistic modeling explores the uncertainty through assigning probability distributions to the input data for a model and calculating the probability distributions for the output variables. For example, in a capital investment appraisal, you can estimate finding the probability of achieving specific net present values or discounted cash flow yields of the cash flow.

Optimization may be used to determine the overall optimal capital investment program subject to constraints such as the total amount invested and the required individual project investment levels, for example for maintenance. Both simulation and optimization are supported in SAP HANA through application function libraries.

3.7. DEEP LEARNING ON SENSOR DATA

Professor Geoffrey Hinton, known as the "Father of Deep Learning," believed that there is no limit as to what deep learning can learn. The idea is that it gets better as it receives more data and more computational time. Applications include image recognition, speech recognition, and robotics and motor control. Deep learning has been described as a set of algorithms that "mimics the brain" and is equated to neural networks that "learn in layers."

There have been great successes with deep learning, but it is also controversial. Some data scientists consider these successes to be over-hyped when claims are made to be able to solve every problem ever experienced. The relevance of deep learning to IoT comes from the huge volumes of data generated by sensors and thus back to what Hinton believed – the more data you give and the more computational time you provide, the better it is. SAP has several research teams specifically working on deep learning and its application.



Data science has a huge range of applications.

3.8. EDGE COMPUTING

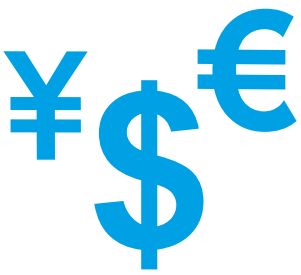
To quote Wikipedia, "Edge Computing is pushing the frontier of computing applications, data, and services away from centralized nodes to the logical extremes of a network. It enables analytics and knowledge generation to occur at the source of the data."[3]

It's a paradigm shift – from the connected device to the intelligent device. To quote Bernd Leukert, member of the Executive Board of SAP SE, with global responsibility for development and delivery of all products across the SAP product portfolio: "Take the algorithms to the data, not the data to the algorithms."

Computing on the edge is very important when a very quick response from the system is required – for instance, in the automotive area, the interaction between a navigation system has to be very quick when the data science component is trying to optimize fuel consumption by taking the driving style into account. It is also required when the data volume generated on-site is so large that the throughput required to process it by a central application, together with the incoming streams from other sources, cannot be provided. This may be the case in scenarios in which high-resolution images or videos need to be analyzed.

Edge computing and connectivity are highly significant for IoT and data science, and this is reflected in SAP by recent announcements such as the partnership with OSIsoft in January 2016: "The SAP HANA IoT Connector solution by OSIsoft [now SAP HANA IoT Integrator solution by OSIsoft] joins the power and advanced analytics of the SAP HANA platform with the OSIsoft PI System – an enterprise infrastructure for connecting sensor-based data, operations, and people to enable real-time intelligence"; and the partnership with Telit Communications: "a global enabler of the Internet of Things (IoT), which has entered into an agreement with SAP to license and resell the Telit deviceWISE IoT platform. The deviceWISE platform connects your 'things' to your 'apps' – seamlessly integrating any devices, production assets, and remote sensors with your Web-based and mobile apps and enterprise systems."

The benefits of the proactive approach to maintenance in the manufacturing industry include a reduction in maintenance costs of factory equipment, a reduction in equipment downtime, and a reduction of capital investment.



4. Data Science in SAP Predictive Maintenance and Service

SAP Predictive Maintenance and Service targets domain experts and business users who are in charge of monitoring the health of assets and planning maintenance activities. The user has the ability to view the assets in a map, view the hierarchy of assets, view important key performance indicators, get alerts triggered by preconfigured rules, and view the sensor data in a two-dimensional or three-dimensional time-series chart. In order to assess the health of assets, SAP Predictive Maintenance and Service provides health scores that are computed using machine learning. Health scores indicate the health status of assets by either showing the remaining useful life, probability of failure, or anomaly scores. SAP Predictive Maintenance and Service operationalizes analytics and machine learning for consumption by domain experts and business users.

4.1. THE ARCHITECTURE – AN OVERVIEW

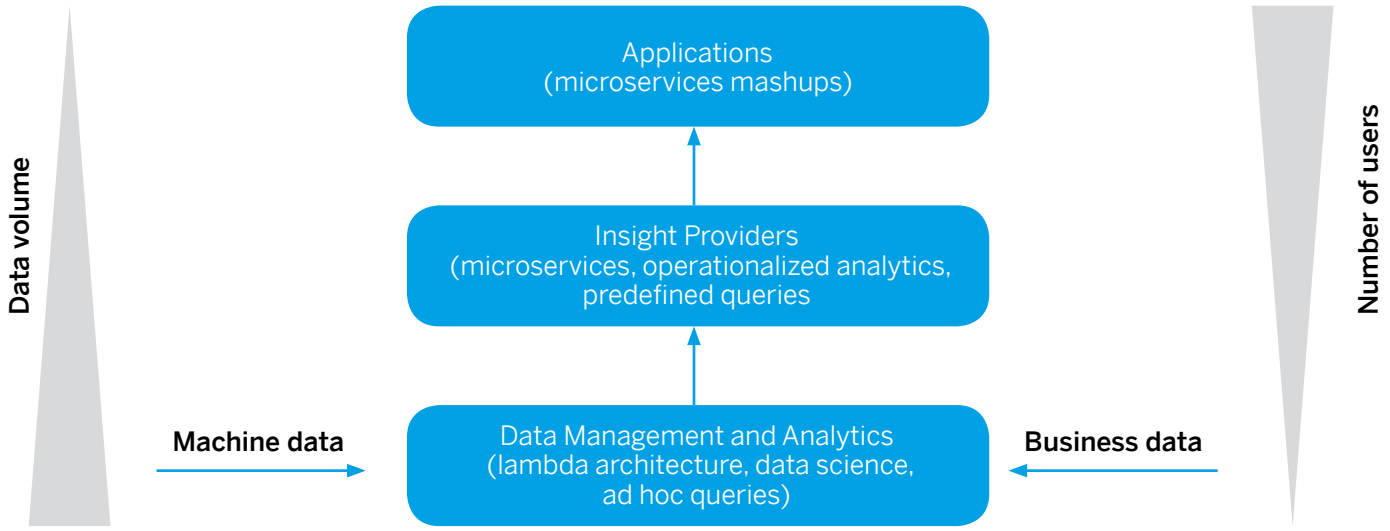
The SAP Predictive Maintenance and Service system architecture is based on two fundamental concepts:

- The data management and data processing, including data fusion, are based on concepts of the Lambda architecture,[4] which is a generic, scalable, and fault-tolerant data processing architecture for Big Data.
- The operationalization of analytics for consumption by the domain experts is based on the concept of insight providers. The insight providers are microservices that provide pieces of the analytical functionality, which are plugged into the applications. Thus the applications are designed to be composed of modular microservices and can be easily enhanced or extended with additional analytical capabilities by adding insight providers to the system.

The main applications of data science, both in IoT and in general, are concerned with "relationships" in which we are aiming to build a model to define the relationships between inputs and outputs.

A key aspect of the system is that the number of users increases from the data-management layers to the application layer while the data volume decreases from the data-management layer to the application layer, as shown in Figure 12.

Figure 12: Predictive Maintenance and Service Architecture

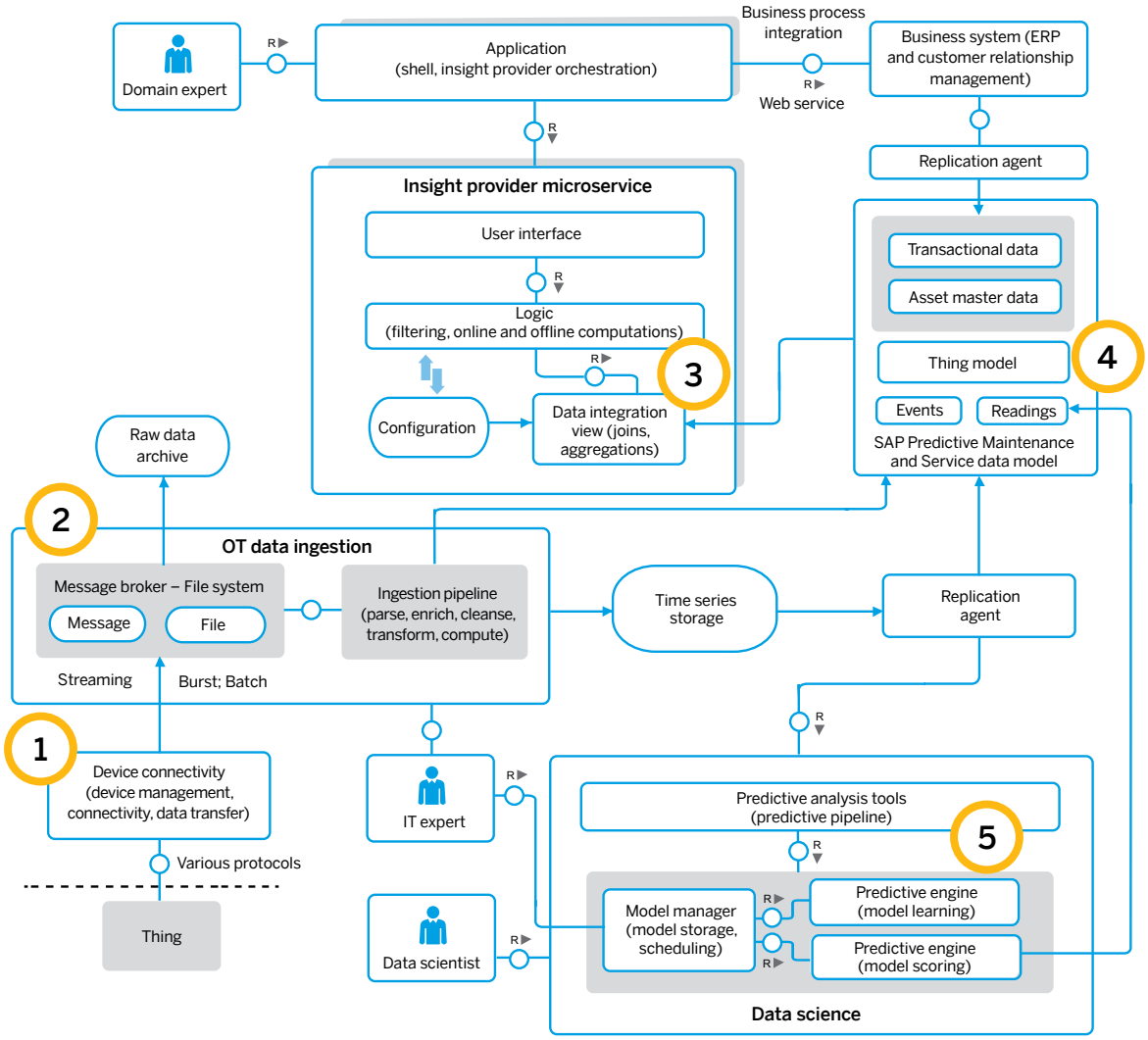


An insight provider offers one specific analysis whose result, the insight, makes sense in the application domain to a domain expert with an engineering background – for example:

- Abnormal exhaust temperature of the engine on a specific airplane
- A high frequency of low-oil-level alerts in a fleet of cars
- A geofenced area of pipelines that show a high degradation in remaining useful life

An insight provider implements operationalized analytics created by data analysts or data scientists and validated by domain experts and business users. The logic of the insight provider may contain derived signal calculations, key value calculations, statistical models, or predictive models. The insights generated by an insight provider may be displayed in its own user interface or made available to other insight providers. The insight providers are implemented as microservices and extend the application functionality in a modular manner. The conceptual overview of the system architecture is shown in Figure 13:

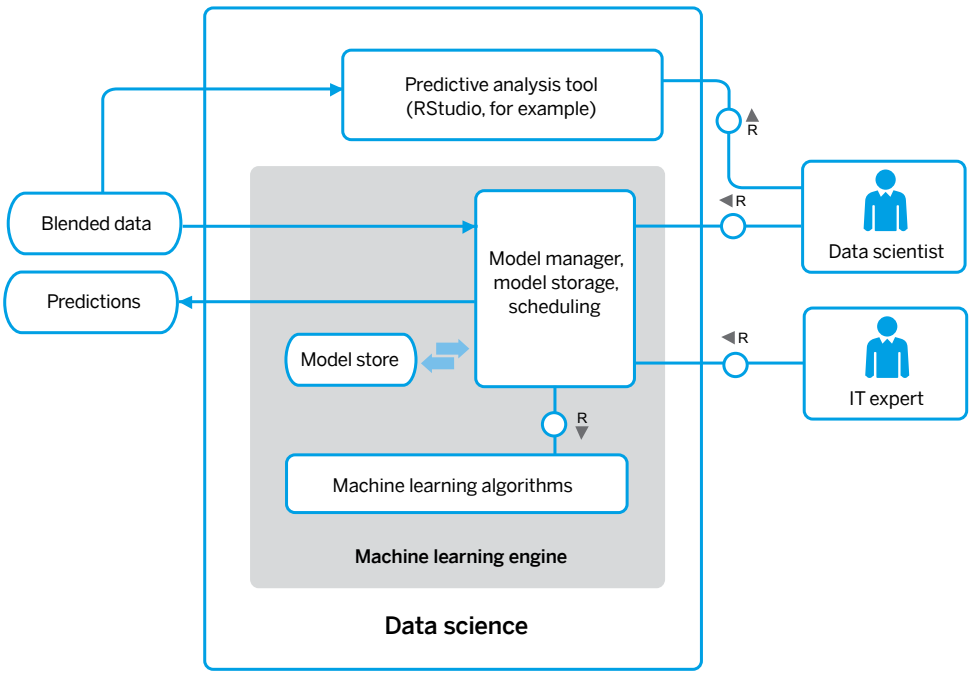
Figure 13: Conceptual Overview of the SAP Predictive Maintenance and Service Solution



1. The device connectivity layer is mainly responsible for transferring data collected through various protocols from the devices to the central storage location. The device connectivity layer also provides the underlying network connectivity, device management capabilities (such as pushing configurations to the devices), and monitoring of devices.
2. Telematics and sensor data (operational technology [OT] data) is ingested through the OT data ingestion layer that handles computations on the incoming data stream. Typically, computations range from simple ones, such as detecting sensor value threshold violations, to complex ones, such as applying predictive models to streaming data (for example, determining the probability of a component resulting in a warranty claim as the component goes through an assembly line). See section 3.1.3 for more information on streaming analytics.

3. The management of business data (information technology [IT] data) and OT data comprises a data ingestion process that consists of parsing the raw data (OT data management) and its conversion into a format that can be processed. The IT data is replicated from a business system such as the SAP ERP or SAP Customer Relationship Management application. Transformations such as pivot transformations, enrichment (including joining data that might be computationally expensive to do later), syntactic cleansing that doesn't require understanding of the meaning of data, and semantic cleansing are done in so-called integration views.
4. The "thing" model is a representation of assets and allows the modeling of them as a hierarchical structure of components and thing types. The latter represents technical as well as functional parts of an asset and, in particular, contains the metadata on sensors (for example, names, physical quantity measured, and unit of measure). The thing model is the primary means to consume technical data for analysis in a meaningful and semantically rich way.
5. Data scientists pull the required data, typically a blend of sensor data, asset metadata, and business data, into their predictive tool, such as RStudio, to perform explorative analysis and model learning. Once the model type and corresponding settings have been found by the data scientist, a model is created in the model manager, which provides the means for ad hoc or scheduled training, scoring, and retraining (see Figure 14).

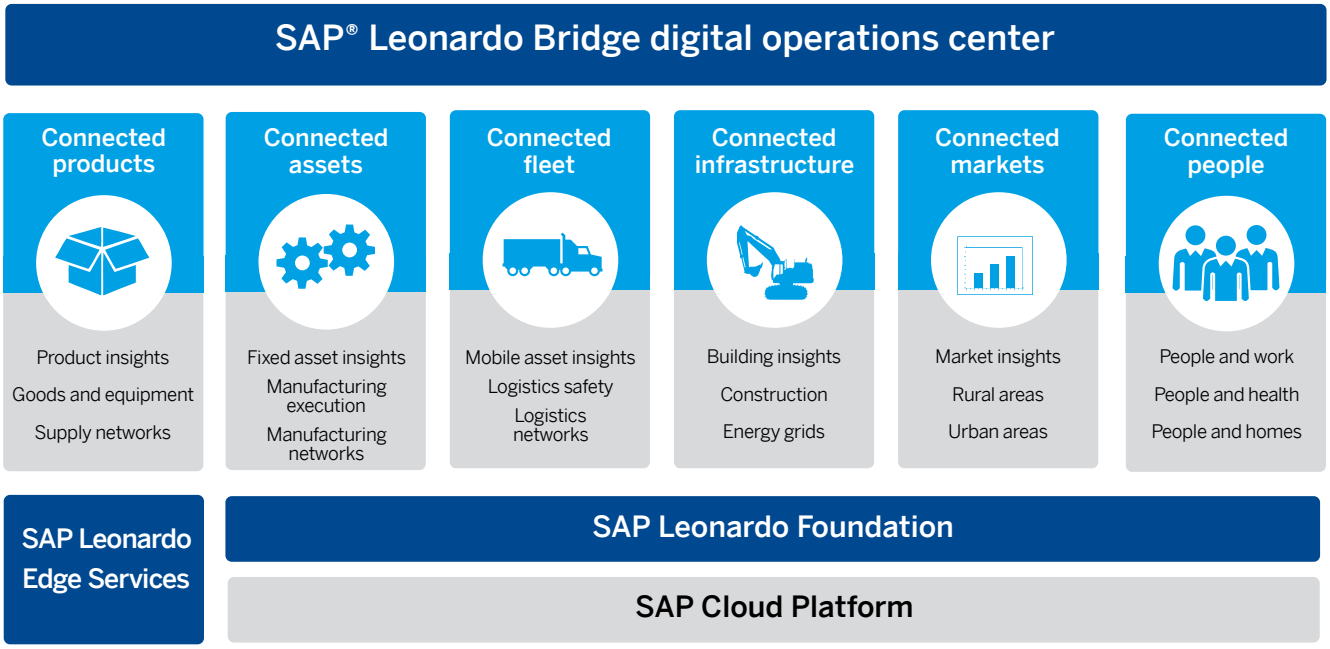
Figure 14: The Data Science Architecture



4.1.1. Support for IoT Machine Learning

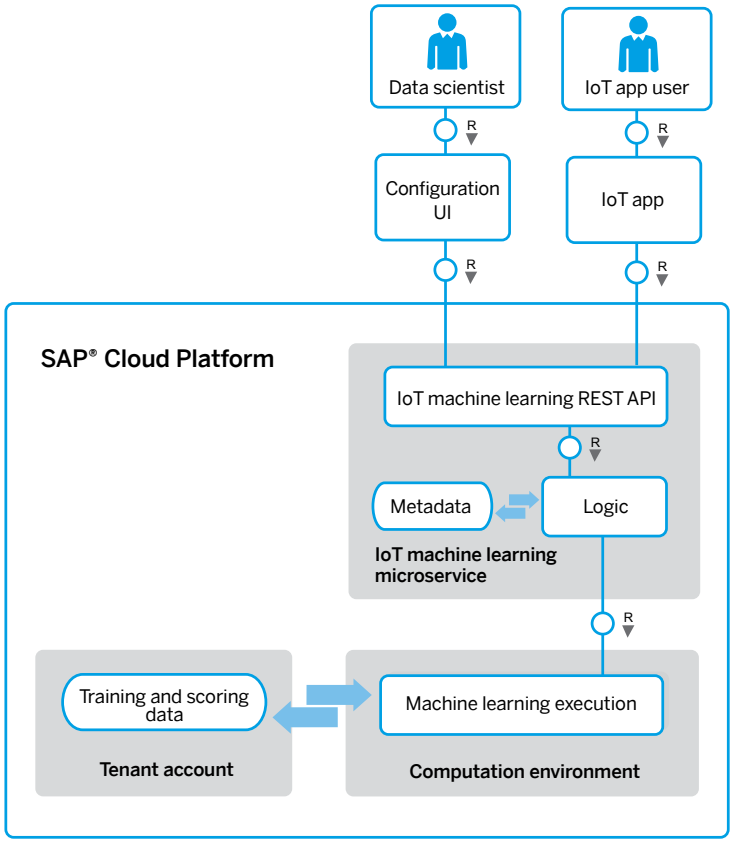
Support for IoT machine learning is part of SAP Leonardo Foundation and builds on the capabilities of SAP Cloud Platform, as shown in Figure 15. It provides machine learning functionality, in particular model management, training, scoring, and anomaly detection, as a cloud-based service for customers and partners using SAP Leonardo solutions.

Figure 15: The SAP Leonardo Portfolio



Offered as a cloud-based service, machine learning benefits from the scalability of processing and storage resources. With execution in a tenant-aware environment, machine learning tasks are mutually isolated to guarantee execution robustness and data privacy. The support for IoT machine learning from SAP allows data to be consumed from tenant accounts for both training and scoring. Likewise, the result of the scoring process is sent back to the tenant account, from where it is consumable again in integration scenarios. The service functions are invoked via a representational state transfer (REST) interface by IoT applications. Additionally, a UI supports data scientists with the configuration and model management. Figure 16 shows the architecture of the IoT machine learning service.

Figure 16: Support Architecture for IoT Machine Learning



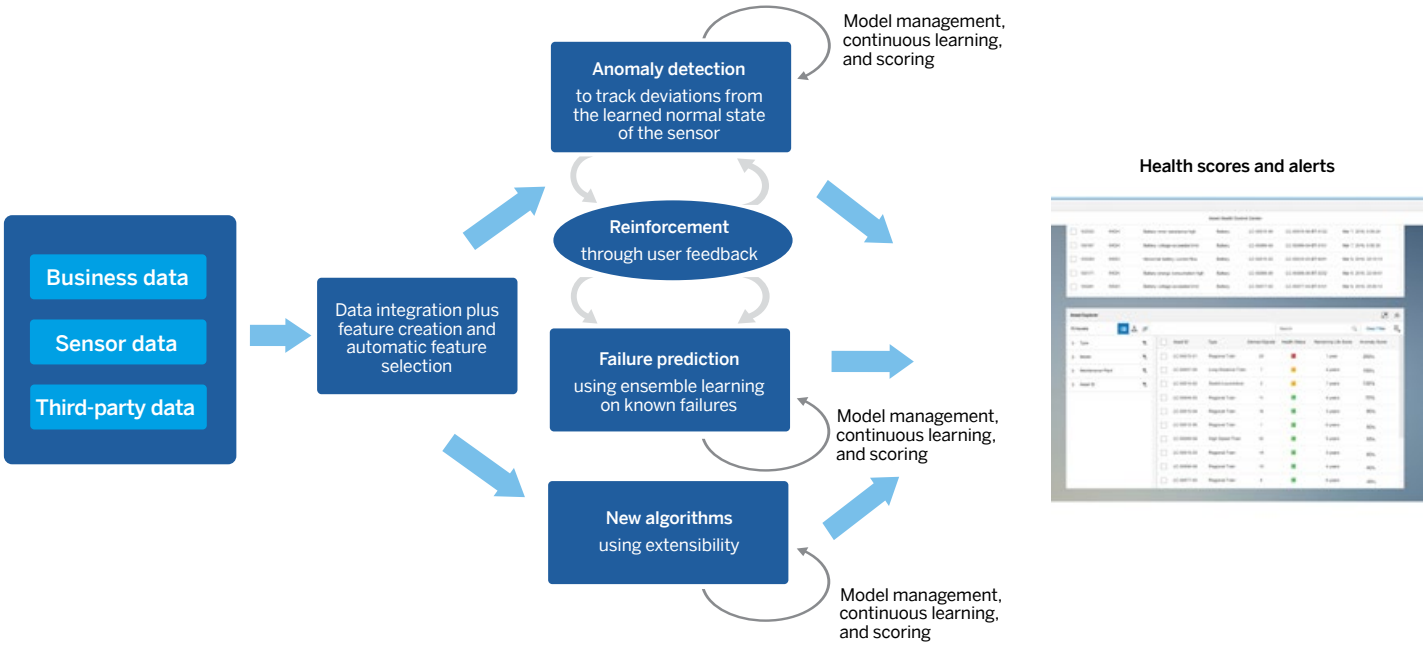
4.2. DATA SCIENCE COMPONENTS IN SAP PREDICTIVE MAINTENANCE AND SERVICE

SAP Predictive Maintenance and Service delivers machine-learning functionality in terms of certain algorithms and data-preparation methods shaped for specific use cases, such as "Weibull Analysis for Remaining Useful Life Prediction," "Principal Component Analysis for Anomaly Detection in Sensor Data," and "Earth Mover's Distance Metric for Bad Actor Detection."

In addition, the solution also provides functionality to trigger the model learning (including parameter setting and data binding), to save the model, to deploy the model and apply it to new data that generates new scores, and to manage the lifecycle of the model (for example, retrain, update, and delete). Insight providers and applications can consume the generated scores. Importantly, SAP Predictive Maintenance and Service is extensible in the sense that data scientists for customers or partners may add additional machine-learning functionality for specific use cases.

This functionality all comes together in the machine learning engine, with the main features shown in Figure 17. The machine learning engine comprises a class of algorithms for anomaly detection (see section 5), such as principal component analysis, distance-based failure analysis, and multivariate autoregression. There is also a class of algorithms for failure prediction, such as tree ensemble classifiers and useful-life analysis using Weibull distribution (see section 5). In addition, the engine is open to new algorithms using extensibility. With the support of model management as well as continuous learning and scoring, the results manifest themselves in asset health scores and alerts.

Figure 17: The Machine Learning Engine



4.2.1. Data Fusion and Preparation

Several steps to transform the raw data have to be performed to create a unified view on the relevant data before a model can be created. These steps are very likely to be highly use-case specific and therefore model specific – for example, the data has to be prepared in a different manner to serve as input for "Remaining Useful Life" scoring compared to a classification model.

The data has to be fused – that is, different data sources have to be integrated. In the predictive maintenance scenario, it is mainly business data (IT data) and telematics and sensor data (OT data) plus sometimes other external data sources, such as weather data. The sensor data collected by a machine has to be mapped to the service notifications or warranty claims created for the respective component that the sensor is attached to.

This is especially important for supervised learning tasks in which the target variable has to be attached to the predictors. In unsupervised learning the sensor data does not require to be fused to business data for the purpose of creating a target variable; however, data fusion might still be needed for other reasons, such as for the evaluation of the results.

Depending on the availability of explicit or implicit links between different data sources, you will require different methods of linking data – for example, in a direct manner by applying joins using primary and foreign keys, such as connect a warranty claim of a specific component to sensor data that is linked to that component, or in a fuzzy manner when explicit links are not available. Fuzzy data fusion is, for example, performed when either the time that an event occurred is not explicitly known (for example, time of failure), failure is only recorded after an unknown number of days, or an explicit connection between items and data is not available (for example, sensors not explicitly linked to components). Text matching or fuzzy time-based joining can be applied. The data fusion is performed using SAP HANA functionality, such as information views. Fuzzy joins are calculated using SAP HANA text analysis functionality or R – see sections 3.5 and 3.1.1.

Data preparation consists of two major steps: data cleansing; feature selection and construction. Data quality has to be verified and the data cleansed before it can be used for analysis. SAP Predictive Maintenance and Service supports both semantic and syntactic data verification and cleansing. In semantic data verification and cleansing, depending on the algorithms to be applied or the business problem at hand, different cleansing methods have to be applied. Examples are removing or filling in missing values using interpolation techniques for some classification algorithms, finding and removing or replacing outliers, and discretizing continuous values when creating a target variable for classification. In syntactic data verification and cleansing, the format of the columns has to be verified and adjusted if required. Examples of such adjustments are transformations of columns from text format to time stamp and conversion of numerical data to other numerical formats.

Critical to getting a model of good quality is the need to perform the right transformations and select and create the right features as input to the algorithm that will be learning the model. In SAP Predictive Maintenance and Service, the focus is on time series–related transformations and feature creation. Features mainly used in

Algorithms are an important part of data science, but, more importantly, they are part of a process for analysis.

supervised learning can be created by using aggregation functions, such as minimum, maximum, average, variance, standard deviation, range of an attribute over a specific time period, or more complex ones like the integral or the increase of an attribute over a certain time period. If a supervised algorithm is to be applied, then the target variable has to be created.

In SAP Predictive Maintenance and Service, data preparation capabilities can be supplemented by the PAL, such as substitution of missing values, different algorithms for outlier detection, scaling, and binning. Additional data preparation methods can be used from R or programmed in SQLScript.

4.2.2. Modeling and Deployment

During the exploratory stage of analysis, the data scientist has to define the test design and choose the algorithm to be applied to the data for model learning. Depending on the chosen algorithm, parameters may have to be estimated. Once a suitable algorithm is found with appropriate parameters, the data scientist triggers the initial model learning in the productive environment to create the model in the productive model catalogue. The model catalogue contains model metadata that includes required information to deploy a learned model in a particular computation engine. A model repository is an artifact in which model metadata can be stored. The catalogue allows potential model consumers to search and select a model based on their metadata, such as model name, version, status, owner, or used features.

Before a model is learned, it consists of a set of instructions for data preprocessing and scoring; then, after data binding and parameter setting, it also contains details about the model itself, such as model statistics and influencing parameters. After a model is learned and deployed, scoring can take place. Scoring is the act of applying a fitted model to a new data set. The performance of the model with new data has to be monitored, and periodical model retraining can be scheduled to avoid a model deterioration from expected performance. New incoming data that is ready to be scored can be inspected for deviations before the scoring stage.

In order to be agnostic to the computing engine as well as to be able to support cloud applications, models are made available as services. A scoring service applies the scoring function or scoring artifact to a new data set.

There is a mechanism to make the service aware of new data that is ready to be consumed for scoring. Whether the data comes as a stream, as a burst, or is persisted and accessed asynchronously, it has to be cleansed and prepared for scoring. Service requests and responses are managed by a broker that is aware of the infrastructure resources, application load, and priorities.

Algorithm accuracy is important but so is understanding and clarity of the model.

5. Data Science in SAP Predictive Maintenance and Service – Examples

Over a period of many years, SAP's Data Science group has conducted many data science projects, examples of which are shown in Figure 18.

Figure 18: Data Science in Predictive Maintenance

Defect pattern identification
- Statistical analysis, text clustering, association analysis, and decision trees
- Visualization of Big Data with parallel coordinates and multidimensional scaling

Systems trending and alert management
- Detect outliers and anomalies in the data with supervised and unsupervised machine learning
- Text analysis and text mining to classify scheduled versus unscheduled maintenance events

Machine health prediction
- Historic machine data used to predict breakdowns through decision trees
- Energy consumption pattern profiles calculated with k-means clustering
- Domain expert knowledge modeled in SAP HANA with decision tables

Vehicle health prediction
- Use association rule mining and regression tree learning to correlate production rework and customer satisfaction data
- Apply R data mining and data visualization capabilities of SAP HANA to surveys and production data sets

Emerging issues
- Analyzing telematics data and relating it to equipment's service and warranty data using text mining and association analysis

Predictive quality assurance
- Visual detection of cracks (image processing techniques)
- Heat image comparison of "areas of interest" of sample images of material with issues to current material (Euclidean vector distance calculation)

Maintenance prioritization
- Forecast production and probability of failure calculation at asset level
- Prioritize maintenance activities based on actual and forecast key performance indicators

Health prediction for aircraft components
- 5 aircrafts over 5 years, 400 sensors each aircraft – 44.1 billion sensor readings
- Correlated with maintenance history (notifications), weather data, and geolocations
- Text analysis to understand maintenance activities
- Use of statistical process control and symbolic aggregate approximation for anomaly detection

Bad actor analytics
- Weibull lifetime analyses
- Classification techniques to identify rotating equipment likely to fail based on past patterns

Root cause analysis for quality issues
- Find causal relationships between claims and production settings from machine readings
- Improve on Statistical Process Control usage

Maximize machine efficiency in production
- Augmenting (human) expert rules with (machine) rule mining (regression trees)
- Approximating machine state to circumvent "rare event problem" (anomaly detection)
- Decluttering sensor data for root cause analysis (trend analysis)

Asset health prediction
- Optimize testing and crew efficiency based on limited resources
- Optimize capital investment for underground residential distribution (URD) cables
- Predict machine health from historic data and outages

We will review some of these analytical approaches in detail.

5.1. ANOMALY DETECTION IN MULTIVARIATE SENSOR DATA USING PRINCIPAL COMPONENT ANALYSIS

There are two broad approaches to anomaly detection. One is to look for anomalies within a given data set, looking for the unusual. The other is to look at expected values compared with actual values, looking for the unexpected. The former is sometimes referred to as unsupervised data mining, the latter as supervised data mining; alternatively, undirected and directed data mining. Predictive maintenance is concerned with asset degradation or failure, which is hopefully very rare, and therefore by definition provides very few examples in a data set of failure upon which to build a robust model and in which the data primarily comprises nonfailure. Consequently, anomaly detection of such data generally takes the form of looking for the unusual.

An example of a data science algorithm for this approach is the simple interquartile range test, in which we can set the parameters of the algorithm to identify anomalies or outliers, calculate an outlier score (such as the percent distance of a data point from the median), and then rank the outliers. We could use other simple tests, such as the Grubbs' test for outliers or other probability distributions. We may need to transform the data before applying these tests – for example, take first differences to derive a stationary series.

We could assemble the results of various tests to get an "agreed-upon opinion." These are all sound approaches; however, they apply to univariate data, whereas data sets in predictive maintenance are more usually multivariate.

There can be thousands of variables, and as such it is almost impossible to detect outliers. We can use cluster analysis for anomaly detection by looking for data unattached to clusters or data within a cluster but a long way from its center. However, that introduces its own issues –for example, some cluster analysis algorithms can make the outlier itself a cluster, some cluster algorithms do not scale well, and some cluster algorithms require a predetermined number of clusters. A successful approach to the problem of outlier detection has been to use principal component analysis to identify the principal components that account for the majority of the variation in the original data.

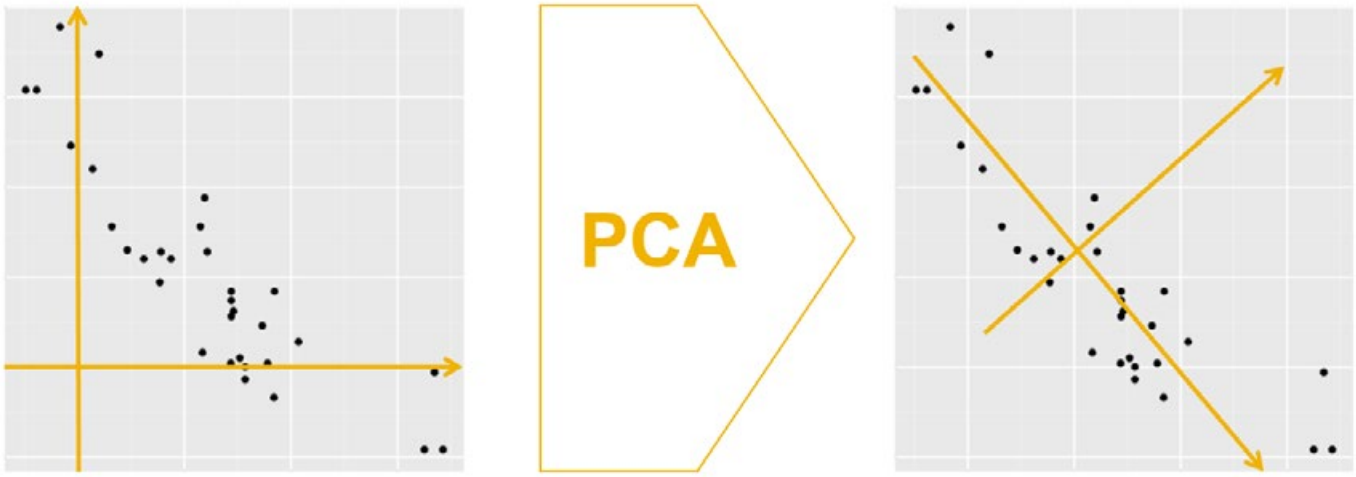
A formal definition is available from Wikipedia:

"Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. The number of principal components is less than or equal to the number of original variables.

"This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest variance possible under the constraint that it is orthogonal to the preceding components. The resulting vectors are an uncorrelated orthogonal basis set."[5]

PCA rotates the coordinate system to explain a major part of the variation of the data by the first few new coordinates, as represented in Figure 19.

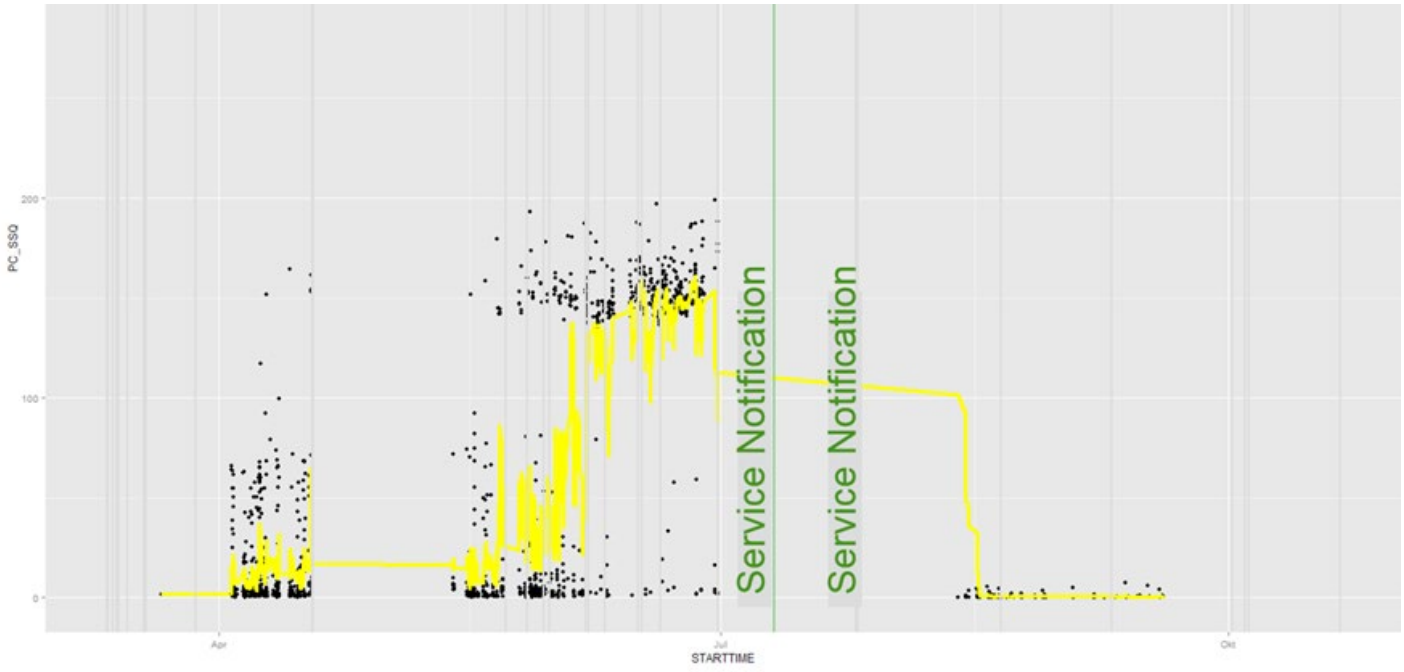
Figure 19: Principal Component Analysis



The anomaly score is the distance of an observation from the mean of the transformed data, where the distance from the mean on every axis of the transformed coordinate system is normalized by the variance of the coordinates of this axis. Sometimes, there are sporadic anomalies – that is, anomalies that occur only for very few successive observations, which according to domain experts do not indicate a failure, which implies that we need to smooth the outlier score. The outlier score may be smoothed using, for example, exponential smoothing or the running median.

A practical example comes from a project on motor sensors. Normally, temperature changes at all sensors in a similar way. Nonnormal behavior can be identified using PCA – Figure 20 shows an example in which the x-axis is "Time"; the y-axis is the PCA score based on the "Mahalanobis Distance" (also known as "Hoteling T2"); the black dots are the values of the PCA scores; and the yellow line is the exponentially weighted average of the black dots. Under normal conditions, the yellow line shows small and regular variation. In contrast, volatility of the yellow line indicates nonnormal behavior.

Figure 20: Principal Component Analysis Applied to Motor Sensor Data

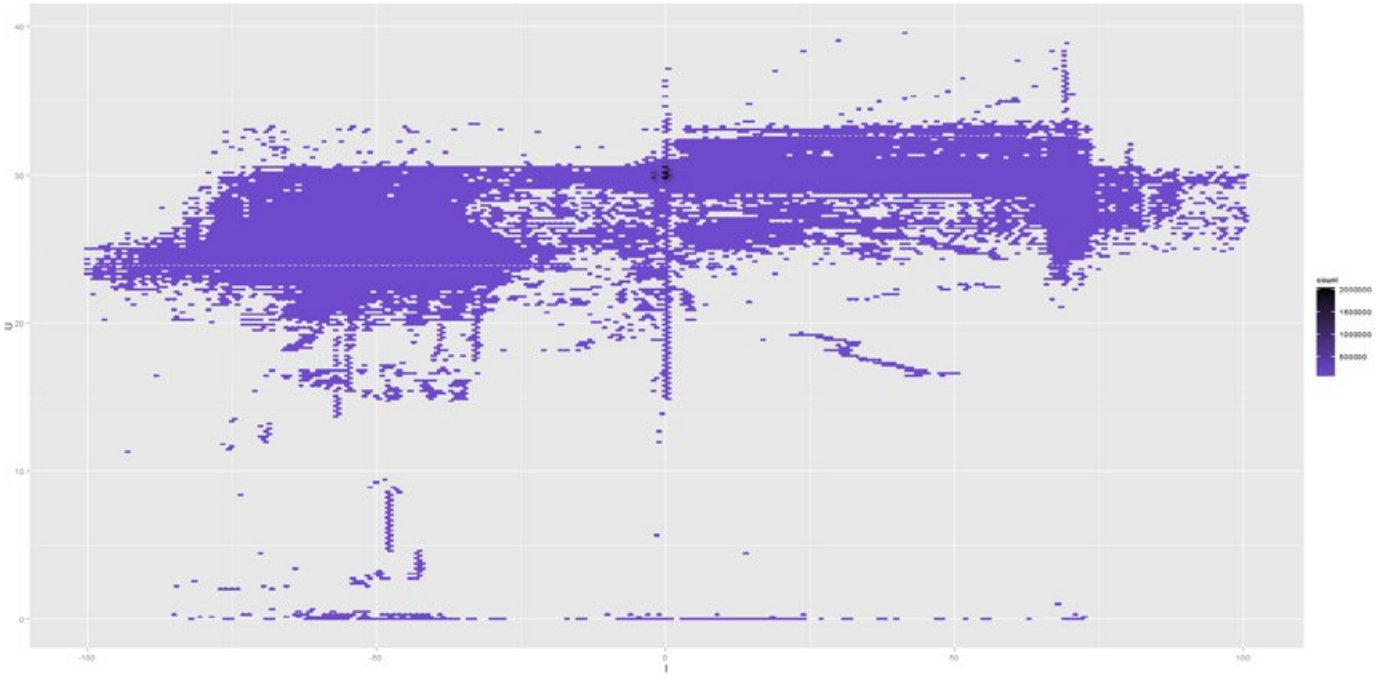


It is important to stress that algorithms can help identify anomalies; however, that is just the start of the analysis. The main task is then looking at the anomalies and determining if they are variations from normal that require investigation, which requires the involvement of domain experts. It is the analogous issue to having a great mathematical model, but do we have a strong causal relationship? The algorithms can identify anomalies, but the question is whether they mean something.

5.2. ANOMALY DETECTION OF SENSOR DATA USING DISTANCE-BASED FAILURE ANALYSIS

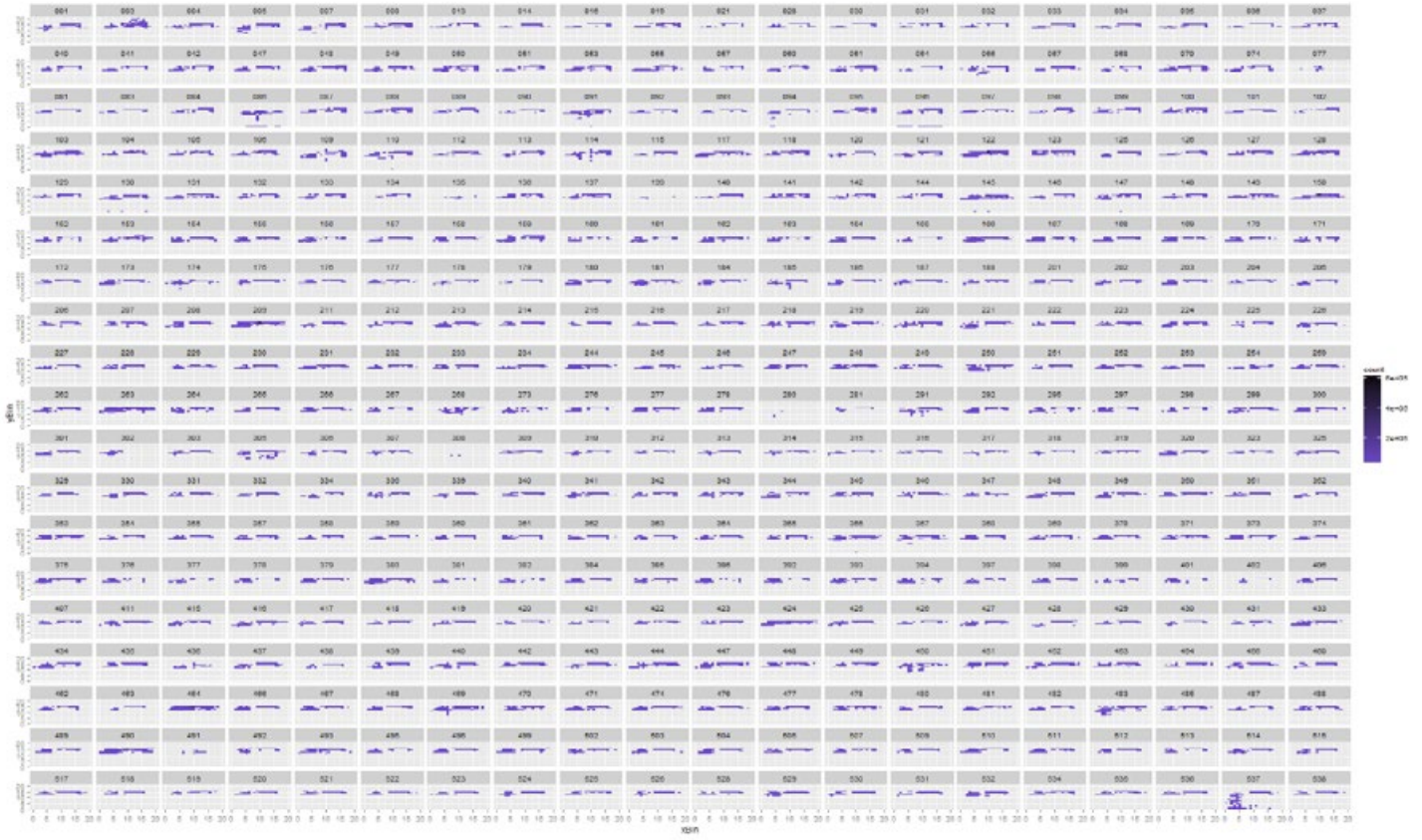
Figure 21 shows the distribution of voltages and current from 380 batteries, in total 15 million data points. The challenge is – how do we find anything interesting in this mass of data?

Figure 21: 15 Million Battery Voltages and Current



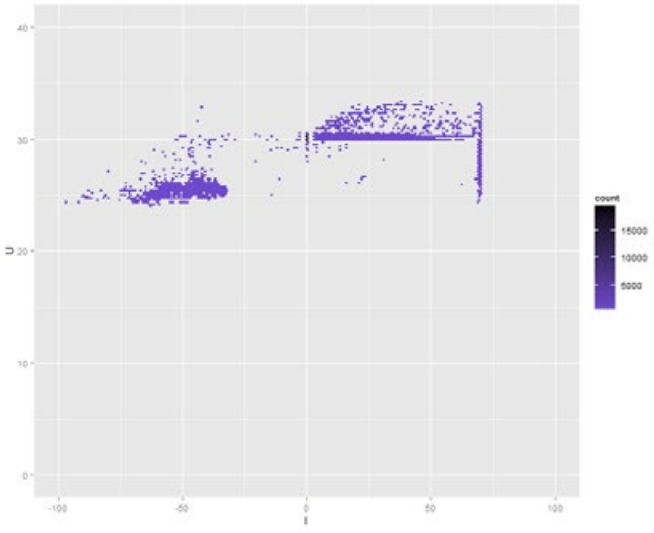
If we drill down into the data, it is still hard to discern patterns and identify abnormal behavior – Figure 22 shows the sensor data for the 380 batteries presented in two-dimensional histograms.

Figure 22: Sensor Data for the 380 Batteries

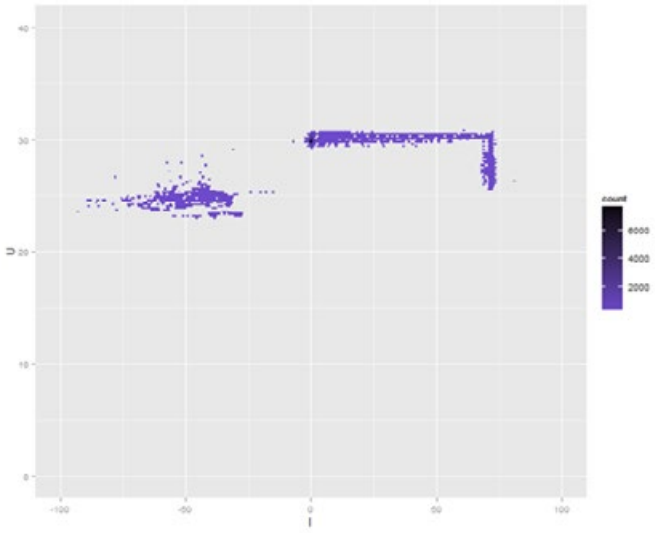


Drilling down further, we can view the aggregate battery voltage "U" and the battery current "I" over time and see the distribution of sensor values. The "histogram" aggregates a time series over the time dimension and visualizes the distribution (see Figure 23). If we can define a "fingerprint" of particular performance, then we can train an algorithm to find variations in each battery's performance compared to the fingerprint – for example, normal behavior.

Figure 23: Specific Battery Performance for Comparison



Battery A

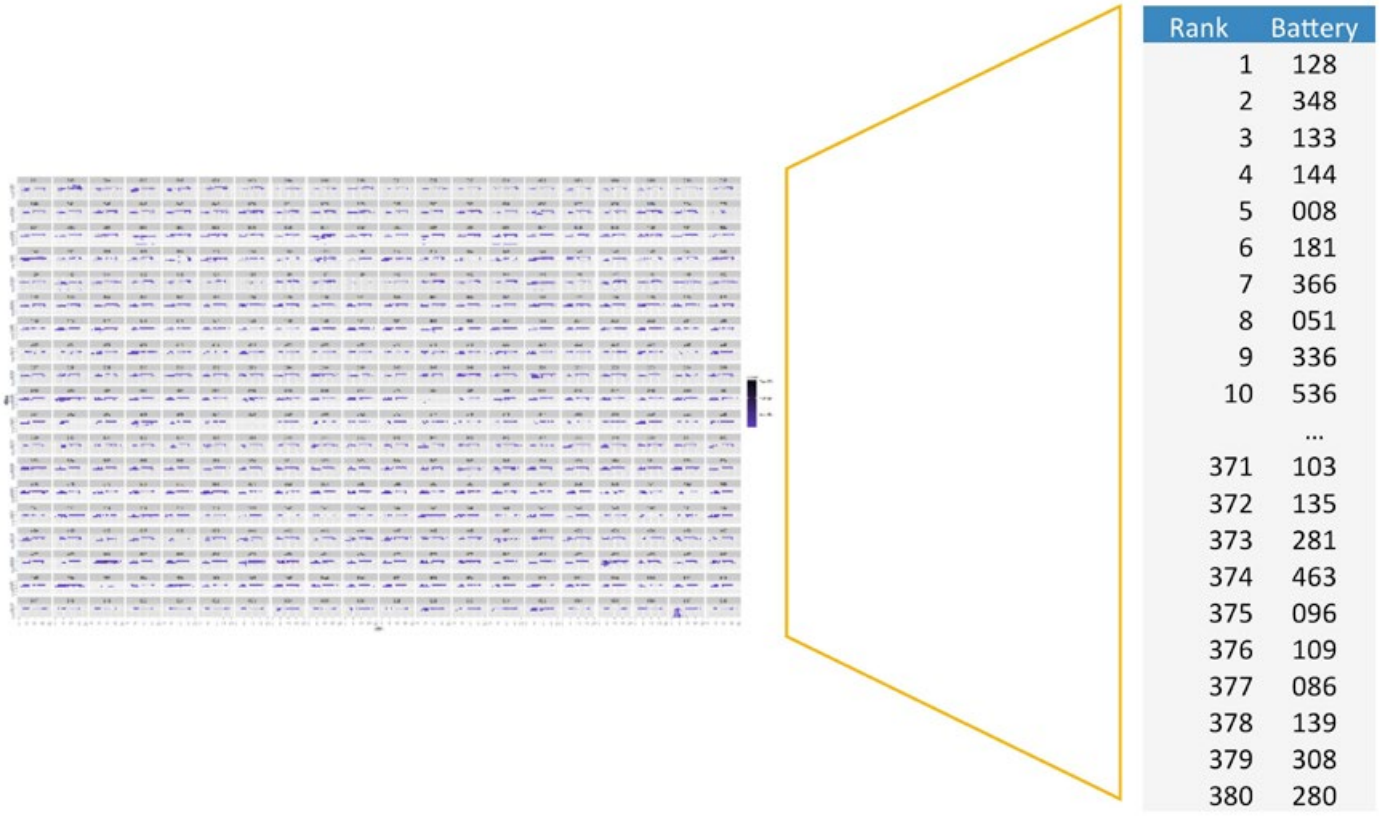


Battery B

The approach is to rank or cluster a set of alike devices by computing probability metrics between samples of one or multidimensional sensor distributions. We can apply this method to identify nonnormal behavior and support data-driven condition-based maintenance strategies. We can infer health or age conditions of a device in comparison to its siblings or to raw models from regular periodical sensor readings or derived features. The sensor snapshots can either be aggregated by statistical summaries for location (mean, median), scale (standard deviation, range), and correlation or by computing representations of the empirical distribution. For up to three dimensions, we can visualize the distributions with scatter plots, histograms, or density plots.

There are many tests for the similarity of probability distributions – for example, the Kolmogorov-Smirnov statistic, the Hellinger Distance, the Wasserstein metric (also known as the Earth Mover's Distance). Using the latter measure and discretizing the battery performance into a matrix, the "distance" between two distributions is computed. Thus we can train an algorithm to automatically inspect the data and rank the "unusual" (see Figure 24).

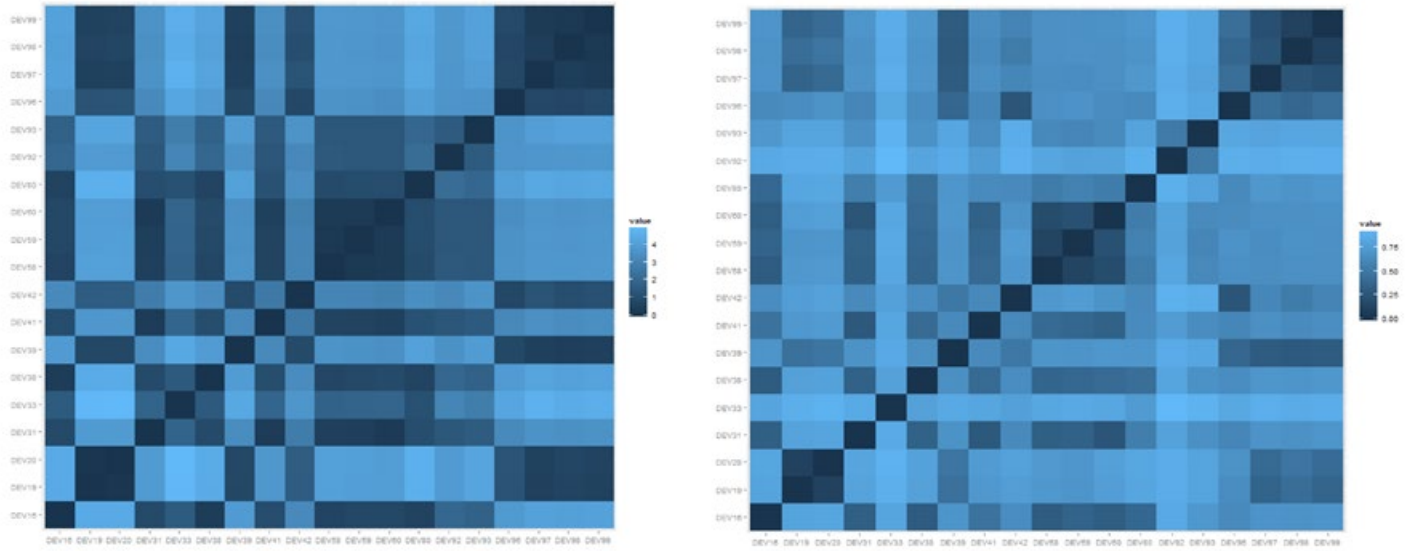
Figure 24: Battery Performance Comparisons – Ranked to Show Anomalies



Rank	Battery
1	128
2	348
3	133
4	144
5	008
6	181
7	366
8	051
9	336
10	536
	...
371	103
372	135
373	281
374	463
375	096
376	109
377	086
378	139
379	308
380	280

We can construct heat maps to show the "similarity" for selected devices (see Figure 25). In this example, 19 devices are listed on the y axis, and the same ones are also listed on the x axis. The "similarity" between the devices is color coded such that the more similar they are, the darker the color. This can help identify "areas of problems." The heat map on the left uses the Earth Mover's Distance as a measure of similarity, while the one on the right uses the Hellinger Distance for comparison.

Figure 25: Heat Maps of Battery Similarity

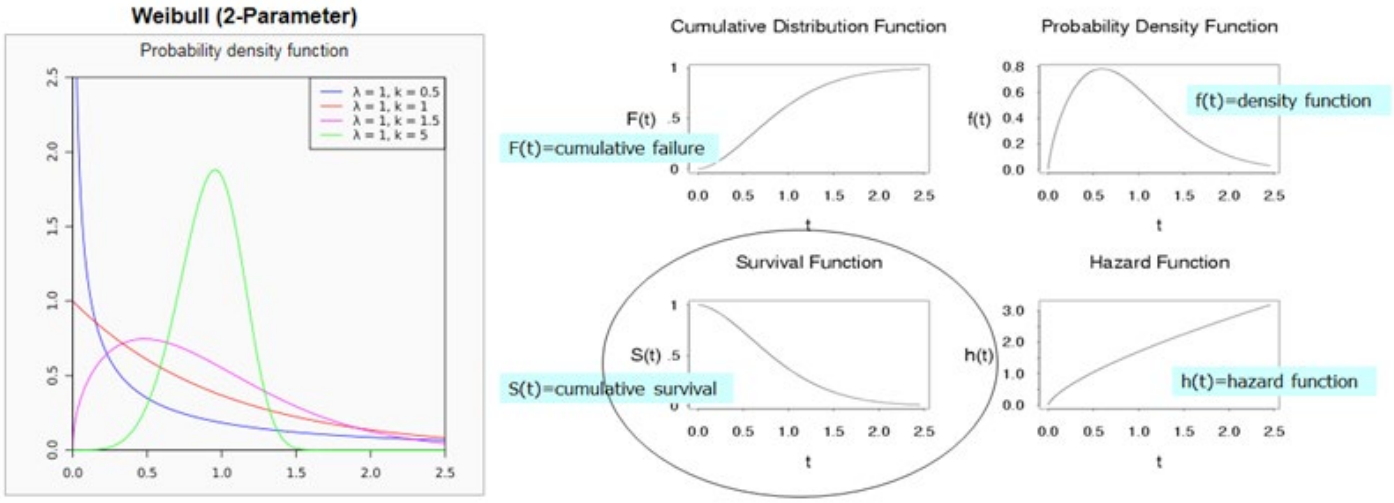


5.3. USEFUL-LIFE ANALYSIS USING THE WEIBULL DISTRIBUTION

Maintenance planners plan their maintenance activities in part on the expected remaining useful life of a machine or component. A well-established model for this kind of analysis is the Weibull distribution. Based on run-to-failure data, the distribution of lifetimes of a modeled machine or component can be estimated. The resulting probability distribution function can be used to calculate the remaining useful life of a machine based on its current age. They will want to synchronize the planned maintenance activities with the probable failures. Information required for this activity is the probability of a failure before the next planned maintenance activity and the period of time that a machine can run before the probability of failure gets too high. The calculated probabilities of failures can be combined with the potential consequence of such a failure, and a risk score for any machine that is used in an operational environment can be calculated.

The Weibull distribution is very popular in predictive maintenance for modeling survival data. It comprises a very simple hazard function (also known as the failure rate, hazard rate, or force of mortality) and is the ratio of the probability density function to the survival function. The survival function (also known as a survivor function or reliability function) is a property of any random variable that maps a set of events, usually associated with mortality or failure of some system, on to time. It captures the probability that the system will survive beyond a specified time. It's the complement of the cumulative density function of the probability density function. (See Figure 26 from Wikipedia.)

Figure 26: The Weibull Distribution

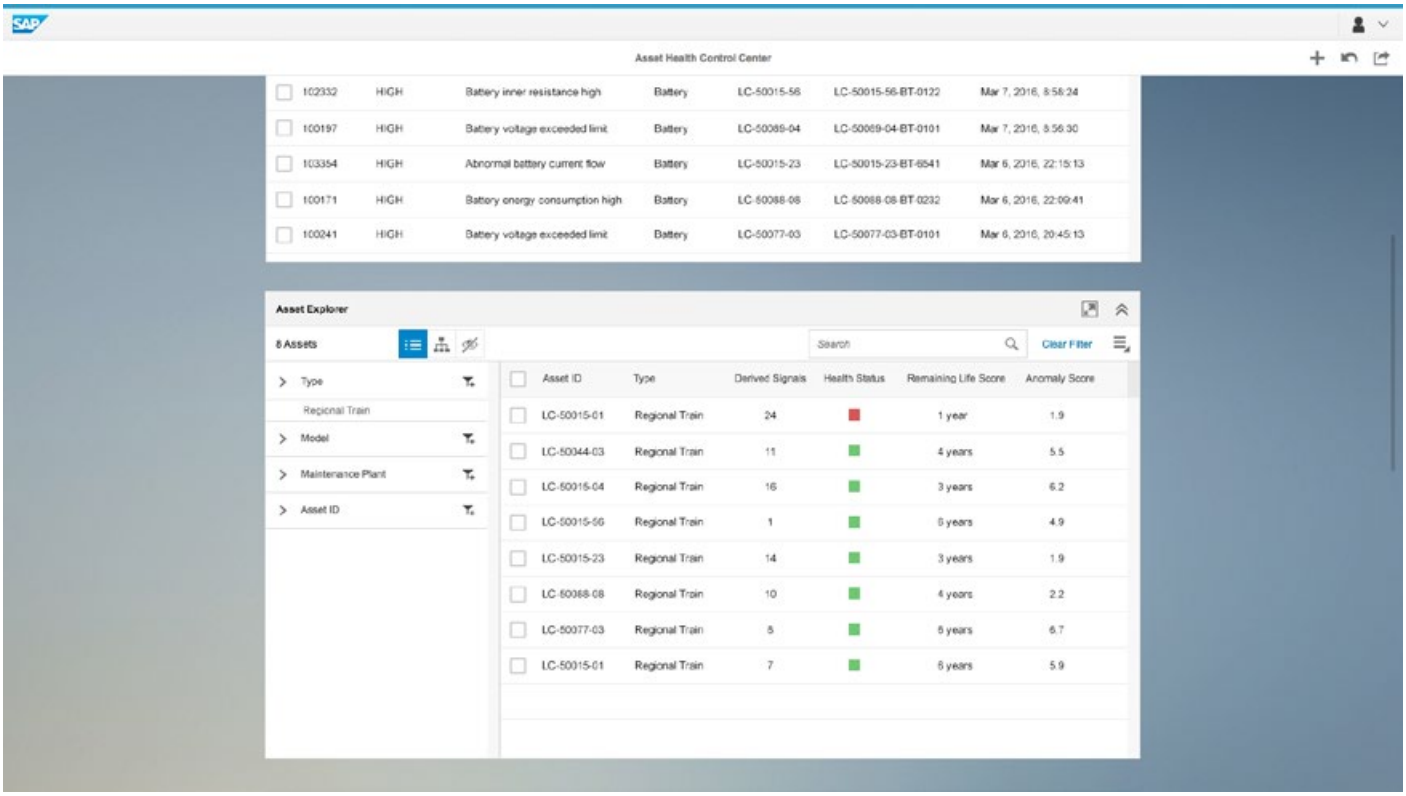


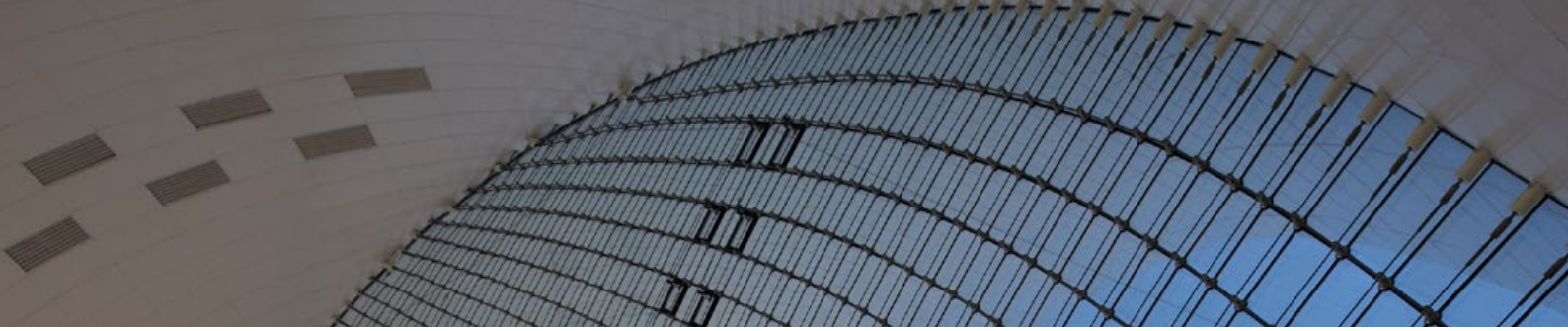
It is also popular, as it can model a wide range of failure behaviors or distributions and supports the "weakest link" property: the minimum of a set of Weibull-distributed random variables with the same shape parameter also follows a Weibull distribution. Furthermore, even with small sample sizes, the Weibull distribution can be used to predict failure times of machines or components.

Asset Health Score models can be derived from data on the machine usage lifetime data. Based on a fitted Weibull distribution and information about the current age of a specific machine or component, it is possible to make statements about its remaining life expectancy and the probability of failure within a certain time interval.

The Weibull distribution can be fitted based on historical run-to-failure data – that is, information on how long machines were operating in the past before the failure of interest occurred. An example of output from SAP Predictive Maintenance and Service is shown in Figure 27.

Figure 27: Asset Health Scores in SAP® Predictive Maintenance and Service



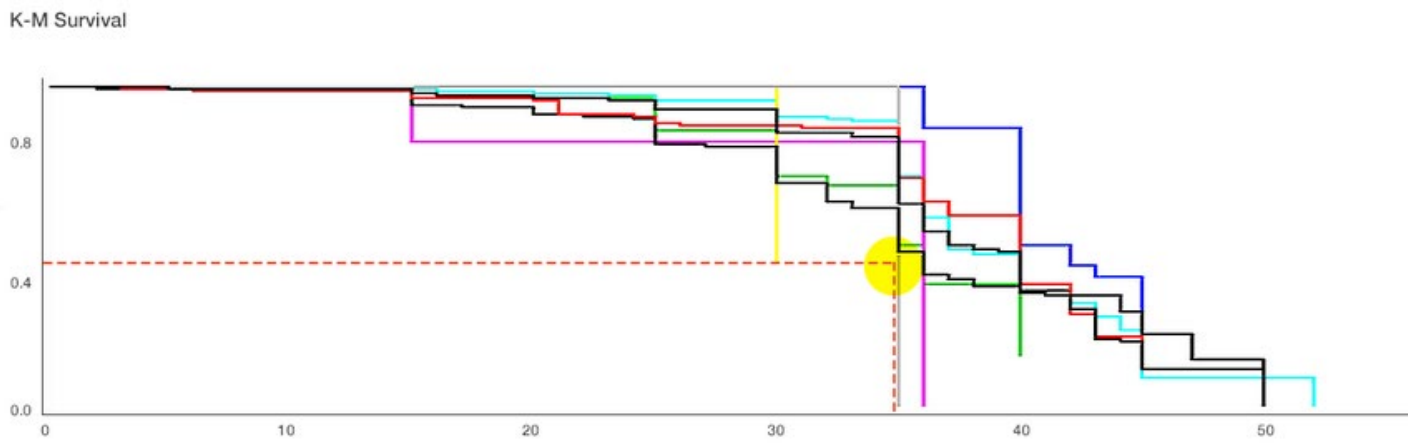


5.4. REMAINING LIFE ANALYSIS USING KAPLAN-MEIER

The Kaplan-Meier estimator, named after its co-inventors Edward L. Kaplan and Paul Meier, is a nonparametric statistic that may be used to estimate the survival function from lifetime data. It is very popular in medical research, where it is used to measure the fraction of patients living for a certain amount of time after treatment. In the domain of predictive maintenance, the Kaplan-Meier estimator may be used to estimate the time-to-failure of machine parts. The advantages of the estimator are that the calculation is very straightforward, the data need not be complete, and statistical measures may be used to compare survival curves with appropriate confidence.

An example from one of the Data Science group's projects is shown in Figure 28.

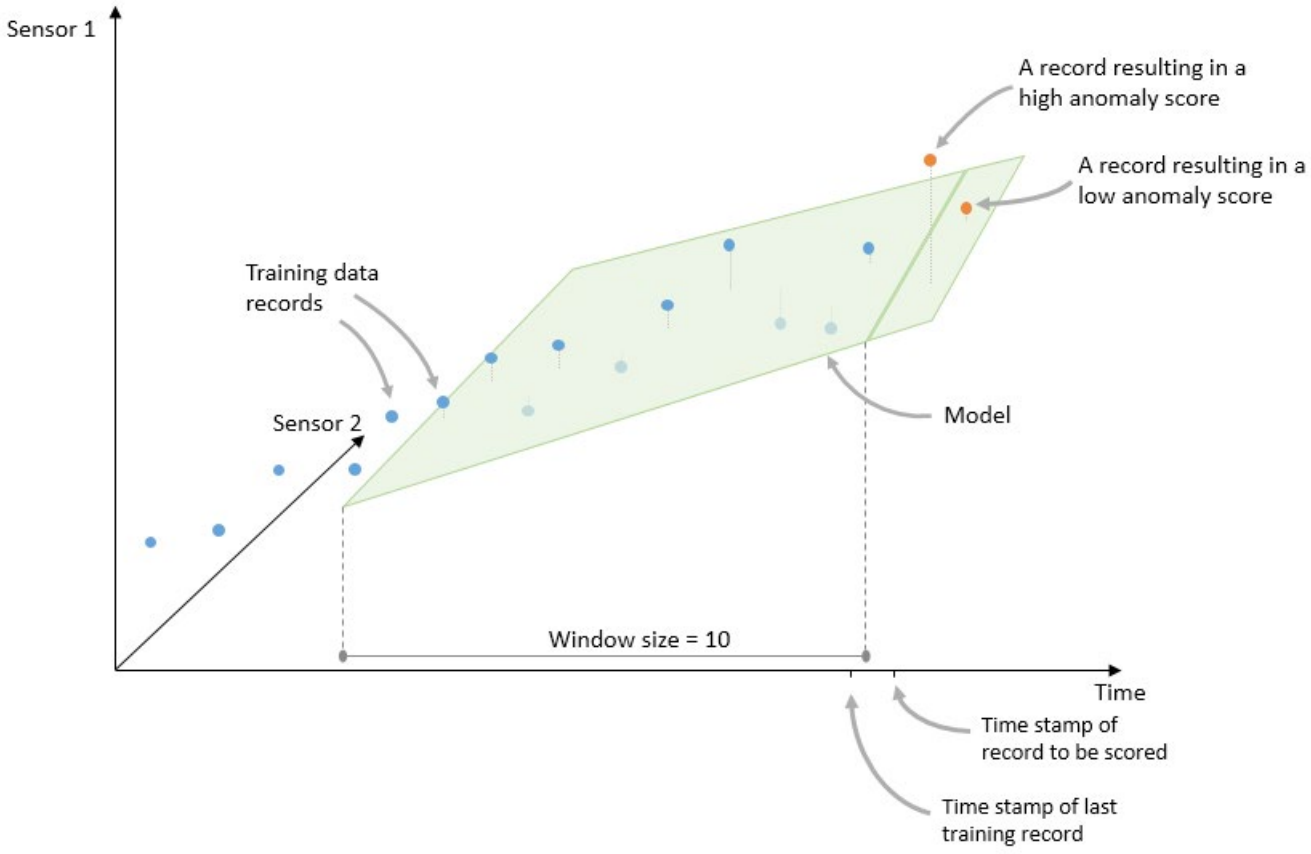
Figure 28: Kaplan-Meier Survival Curves



5.5. ANOMALY DETECTION USING MULTI-VARIATE AUTOREGRESSION

A multivariate autoregressive model can be used to detect anomalies in a univariate, but more usefully, in a multivariate series of sensor data records varying over time. Based on the historical training data, which in this case is a time series of data records, the algorithm trains a model. If trained on regular data – that is data without anomalies present – then the model is capable of learning the regular behavior of a system. Based on a window of recently observed data records, the model can then predict the data for one time period into the future. Once the actual values for this point in time are available, the model prediction can be compared to the actual observation, and an anomaly score assigned based on the distance between the prediction and the actual observation. If large deviations appear, this can indicate abnormal behavior of the underlying system. Figure 29 illustrates how the predictive model is derived from a two-dimensional time series of sensor data.

Figure 29: Anomaly Detection Using Multivariate Autoregression



The implemented algorithm computes one multivariate predictive model per input variable, and aggregates the deviations for the components from each model. A harmonization of time stamps (accounting, for example, for delays in data acquisition) may be performed for certain systems to make sure data belonging to one record is consistent, in the sense that it belongs to the same system state.

Scoring is applied to a series of the window size (n) plus one, which are consecutive records, referring to the order of their time stamps. The first of these records is used as input for the models established during training to produce predictions for each target of the record number window size plus one. Each prediction is compared to the actual values of the window size plus one record. An anomaly score is derived based on the distance between predictions and observations as well as other influencing factors, such as model uncertainty.

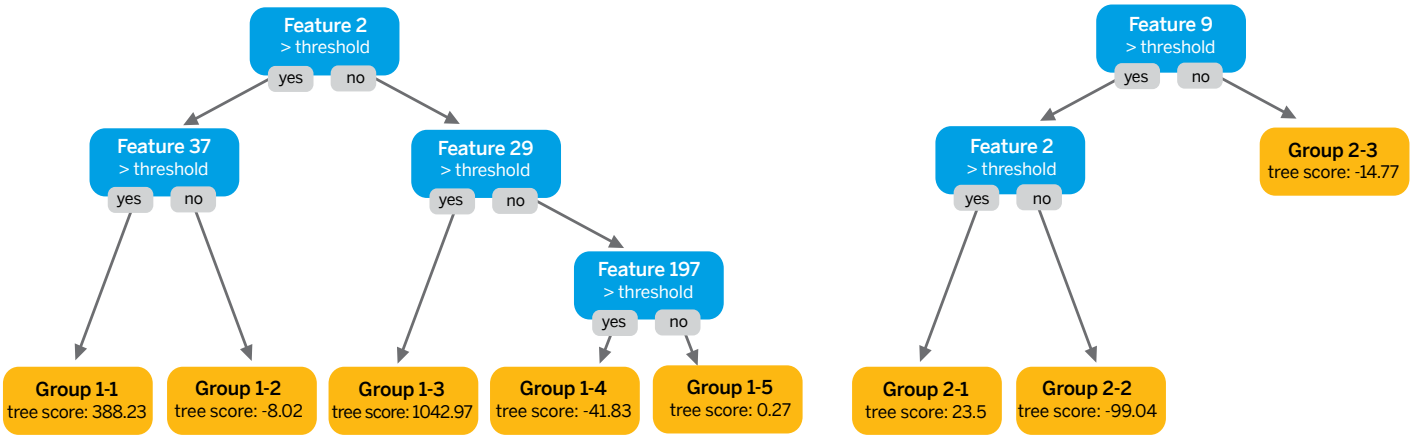
5.6. FAILURE PREDICTION USING A TREE ENSEMBLE CLASSIFIER

Based on records of sensor data, a tree ensemble classifier (TEC) model can learn to predict future system failures from past failures, particularly when the occurrence of failure is very rare, which is by definition quite difficult. The ensemble method used is gradient boosting applied to regression trees, which builds an ensemble of trees one by one. Then the predictions of the individual trees are summed. Each subsequent tree attempts to model the difference between the target variable and the current ensemble prediction by reconstructing the residual.

The algorithm trains a boosted regression tree model (a series of trees) that encodes characteristics of data records with regard to failure. Based on the values of features of a given data record, the model is trained in such a way that each tree can decide which set of record groups the given record belongs to. An appropriate weight is then assigned to each record, indicating evidence for or against the record belonging to a failing system. The model aggregates the evidence weights of all trees and outputs a probability of failure. Thus, the model reflects the likelihood that the given data record is an indication of a failing system.

Figure 30 illustrates a tree ensemble model created by the algorithm. Group x-y denotes the yth leaf in tree number x. In the first tree, the first number is always 1 and the second number is simply consecutive for each branch in the tree. For the second tree, leaves are labeled Group 2-1 through to Group 2-n. The tree score is the contribution to the final score based on this tree, where the final score is given as the sum of all "tree scores."

Figure 30: Example of a Tree Ensemble Classifier



This algorithm is a supervised learning method, which means that it requires training data records featuring a column that indicates for each record whether a record belongs to a regular or a failing system. Model training for the TEC means using the provided historical training data to learn a series of regression trees, including decision thresholds, evidence weights, and an appropriate mapping of weights to probabilities of failure that represents the training data well. Together, these make up the model as referred to in the context of this algorithm. The aim is to find a model that best represents the data set used for training. Standard hyperparameter configuration aiming at robustness is provided. For example, advanced options permit tuning for things like the number of trees, their maximum depth, the learning rate or bias, row and column subsampling, and weight regularization.

To score a record, the TEC model determines the group that the record belongs to for each regression tree based on the feature values of said record. The assigned evidence weights of each tree are then aggregated for this record. As illustrated in Figure 30, these weights could be -41.83 (taken from group 1-4 of tree 1), and -99.04 (taken from group 2-2 of tree 2). Which weights of a tree are aggregated depends on the decision result of a tree. Based on the resulting final weight, the algorithm determines whether or not the record indicates a system failure alongside the certainty of this prediction. Both pieces of information are written to the SAP HANA database as a health score for further use in SAP Predictive Maintenance and Service.

6. Customer Case Studies

Here are three case studies –
Kaeser Kompressoren, Trenitalia, and GEA.

6.1. KAESER KOMPRESSOREN

Kaeser Kompressoren implemented a real-time business solution powered by SAP HANA to analyze vast amounts of granular, real-time data.

Hear from CIO Falko Lameter about how SAP HANA enables Kaeser to offer world-class customer service.

Watch it here (then under *Related Links,* click *Video: Kaiser Kompressoren testimonial*).

6.2. TRENITALIA

Trenitalia uses predictive analytics inside SAP HANA to anticipate failures before they occur, eliminating unnecessary maintenance.

Read about it here.

Also, hear from CIO Danilo Gismondi – "Trenitalia: Creating a System of Maintenance Management Powered by SAP HANA."

Watch it here.

6.3. GEA

GEA uses SAP Predictive Maintenance and Service to help remote service technicians monitor the status of machines at customer sites and identify unusual trends or machine behavior.

Read about it here.

7. References

1. http://help.sap.com/hana/SAP_HANA_R_Integration_Guide_en.pdf
2. http://help.sap.com/hana/SAP_HANA_Predictive_Analysis_Library_PAL_en.pdf
3. https://en.wikipedia.org/wiki/Edge_computing
4. http://lambda-architecture.net
5. https://en.wikipedia.org/wiki/Principal_component_analysis

The SAP Enterprise Architecture Explorer site focuses on specific IT areas, such as user experience, landscape architecture, and the IoT, and provides valuable insights for enterprise architects and others interested in driving IT decisions and improving their enterprise.

For information on the SAP Predictive Maintenance and Service solution, visit the following Web sites:

- https://eaexplorer.hana.ondemand.com/_item.html?id=11527
- https://eaexplorer.hana.ondemand.com/_item.html?id=11568

The major activity in the data science process is spent on identifying, accessing, and preparing data for analysis.

The Best-Run Businesses Run SAP®