Computers and Chemical Engineering 181 (2024) 108513

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/cace

# Predicting the performance of an industrial furnace using Gaussian process and linear regression: A comparison

Andrea Galeazzi [a], Francesco de Fusco [a], Kristiano Prifti [a], Francesco Gallo [b], Lorenz Biegler [c], Flavio Manenti [a,*]

[a] Dipartimento di Chimica, Materiali e Ingegneria Chimica "Giulio Natta", Politecnico di Milano, Piazza Leonardo Da Vinci, 32, Milan, 20133, Italy
[b] Itelyum Regeneration S.p.A., Via Tavernelle, 19, Pieve Fissiraga, 26854, Italy
[c] Department of Chemical Engineering, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA, 15213, United States of America

ARTICLE INFO

*Keywords:*
Machine-learning
Gaussian process regression
Linear regression
Furnace
Predictive maintenance

ABSTRACT

Maintenance is a crucial aspect of the process industry affecting economic and efficiency losses. Among different approaches, predictive maintenance allows for anticipating failure, thus reducing downtime. This work explores a data-driven approach to predictive maintenance by comparing the performance of two different statistical models in extrapolating the future performance of an industrial furnace. The models of interest are a polynomial regression model and a Gaussian process regression model, compared using rolling cross-validation. Moreover, three different machine learning techniques were compared during the training phase: cross-validation, ensemble method and train/test split. The models were trained on real-time series data collected from the distributed control system of a refinery plant. The best performance was obtained with the Gaussian process regression model trained with a train/test split approach. The resulting model can satisfactorily extrapolate the performance of a process furnace over a relatively short-term period.

## 1. Introduction

The impact of maintenance is a well-known problem in the process industry. In 2016, the economic loss associated with preventable maintenance was estimated at around $119.1 billion (Thomas and Weiss, 2020), in the US alone. Therefore, approaching maintenance with a shift from a reactive or even a preventive to a predictive paradigm should be considered when aiming at cost and time optimization (Dekker and Scarf, 1998; Kobbacy and Murthy, 2008; Poor et al., 2019; Venkatasubramanian et al., 2003c). Reactive maintenance is applied primarily as a failure-based approach, meaning that maintenance is conducted only after a failure occurs. Preventive maintenance, on the other hand, is scheduled and planned in advance based on some criteria that may not reflect the current status of machinery. Predictive maintenance, instead, is a condition-based maintenance strategy that carries out maintenance actions only when needed, avoiding unnecessary preventive actions or failure (Florian et al., 2021). Predictive maintenance methods can be divided into three categories (i) physical model-based, (ii) knowledge-based, and (iii) data-driven based (Venkatasubramanian et al., 2003c,a,b; Zonta et al., 2020).

The first (i) approach cannot deal with the complexity of real-world aging phenomena since it is usually the result of controlled experimentation, e.g. clean small-scale laboratory studies, therefore mechanistic models may rarely be used as is in a real-world production environment (Bogojeski et al., 2021). The weakness of the first approach is addressed by the (ii) knowledge-based method that combines a mechanistic model with knowledge derived from data. This may result in the development of a digital twin asset. In this case, predictive maintenance solutions using digital twins in the process industry are not yet broadly available and are rarely used to forecast aging phenomena (L. H. Chiang and Braatz, 2001; Melesse et al., 2020; Bogojeski et al., 2021). Moreover, developing a mechanistic model that can account for all the relevant data-driven information might be infeasible, in some circumstances. In particular when the underlying process is complex, intricate, or not fully understood. In these cases, no mechanistic model could be developed using simple equations.

With the increasing interest in big data and following the latest advancements in Artificial Intelligence (AI), predictive maintenance based on data-driven methods (iii) has become the most attractive approach to fault diagnosis and remaining life assessment so far, especially in the framework of Industry 4.0 (Qin, 2012; Yan et al., 2017; Zhu et al., 2018; Zhang et al., 2019; Carvalho et al., 2019; Yeardley et al., 2022). This approach implies analyzing and elaborating time series data descriptive of a unit life cycle through the employment of

* Corresponding author.
*E-mail address:* flavio.manenti@polimi.it (F. Manenti).

https://doi.org/10.1016/j.compchemeng.2023.108513
Received 26 May 2023; Received in revised form 27 October 2023; Accepted 19 November 2023
Available online 21 November 2023

AI-based techniques. The traditional approach to tackling data-driven predictive maintenance relies mainly on statistical analysis methods, such as principal component analysis, least squares, etc. (Qin, 2009; Yin et al., 2012, 2014). However, several more complex Machine Learning (ML) methods have been applied to find statistical models of the useful life of chemical process equipment (Susto et al., 2015; Cline et al., 2017; Carvalho et al., 2019; Quiñones-Grueiro et al., 2019; Silvestrin et al., 2019; Traini et al., 2019; Becherer et al., 2020; Ren, 2021; Efeoglu and Tuna, 2022; Hichri et al., 2022; Natanael and Sutanto, 2022). For example, Wu et al. (2018) have modeled fouling in heat exchangers through a classical statistical method such as the partial least squares regression; while Radhakrishnan et al. (2007) and Aminian and Shahhosseini (2008) have trained recurrent neural network also to model a heat exchanger fouling process.

In recent years, Gaussian Process Regression (GPR) has received extensive attention in the world of ML, due to its advantages in data modeling given by its Bayesian framework (Yan et al., 2009). Indeed, these non-parametric models are flexible enough to model highly complex data while preventing over-fitting; also they aim at making predictions that quantify the uncertainty due to limitations in the quantity and quality of the data (Frigola, 2015). For example, Ge et al. (2011) developed a combined local Gaussian process regression to formulate a soft sensor for predicting the quality of polypropylene at an industrial level, with successful results. Zeng and Liang (2022) applied a Deep Gaussian Process approach for predicting the remaining useful life of physical assets, with an application on the NASA turbofan engine degradation dataset C-MAPSS, already studied by the authors using a more conventional GPR approach (Zeng et al., 2021). Susto et al. (2012) compared the performance of the Kalman filter and Gaussian kernel density estimation on the silicon wafer epitaxial deposition, finding the latter to be more accurate. Gaussian process regression has the potential to be hybridized with physics laws. Zhang et al. (2022) developed a hybrid approach using physics-guided GPR which showed much higher prediction accuracy than standard GPR, supported vector regression, and recurrent neural networks, on a data set for a Heating Ventilation and Air Conditioning (HVAC) system. At last, Pandit and Infield (2018) successfully implemented Gaussian processes to assess three important operational curves of wind turbines, namely, the power curve, rotor speed curve, and blade pitch angle curve.

In this work, a machine learning algorithm using Gaussian process regression for the extrapolation of time series patterns is proposed. The prediction performance of the GPR method is compared to that of a traditional parametric polynomial linear regression. Also, three different approaches for the training phase are compared.

The remainder of this article is structured as follows. In Section 2 the problem statement is described (Section 2.1), along with the background theory and the assumptions made in the modeling. After a description of the industrial process (Section 3.1), Section 3 focuses on the application of the framework proposed, with an analysis of the performance of the industrial process variables (Sections 3.3.1, 3.3.2, and 3.3.3) and long-term extrapolation reliability (Section 3.4). From this approach, conclusions are drawn in Section 4.

## 2. Methods

### 2.1. Problem statement

The aim of this work is the development of an algorithm for a purely data-driven approach towards performance forecasting of the furnace *PH-401B* of the Itelyum used oil re-refining plant, with the intention of developing more accurate planning of maintenance operations.

Currently, the maintenance strategy is to wait until the pressure drop of the process furnace reaches approximately $10\,\mathrm{barg}$, then halt operation and perform cleaning procedures. However, this decision is not subject to time-optimization strategies but it is a decision of the operations management team. By monitoring and predicting the trend of the methane feed, tube pressure, and tube skin temperature it is possible to give better insights and aid in answering the question of when to stop operations for cleaning and maintenance.

One of the questions that this work is trying to answer is about one of the most important aspects of data-driven time series forecasting, i.e., automatic model selection. The solution applied for selecting the best performing Gaussian process model is to perform a greedy search optimization over the possible combinations of Gaussian kernels. The data-driven approach has been chosen as it is agnostic concerning the process under investigation or the specific variable of such process.

### 2.2. Process, data, and integration

The process unit taken under study is the furnace PH-401B in the thermal de-asphalting section of the Itelyum Regeneration S.p.A. plant located in Pieve Fissiraga (LO). The plant processes around $200\,\mathrm{kt}$ per year of used mineral oils to produce regenerated lubricant bases. The process, based on the Revivoil technology (Dalla Giovanna, 2012; Kupareva et al., 2013; Baldo and Gallo, 2020; Gallo et al., 2021), is composed of three main steps: preflash, thermal de-asphalting, and hydro finishing. Furnace PH-401B is responsible for heating the oil to $360\,^{\circ}\mathrm{C}$ for the subsequent vacuum distillation in the thermal de-asphalting section. A more in-depth description of the unit operation is given in Section 3.1.

The time series data of the selected tags were taken from the Yokogawa Exaquantum Plant Information Management System (PIMS) responsible for managing process data. Furthermore, additional data accounting for special interventions at the unit was integrated manually for representing the blowing procedure, shown in Fig. 5. The data automatically extracted from the Exaquantum PIMS is stored in a CSV (Comma Separated Values) file and fed on a daily basis to the forecasting algorithm which is rebooted every time the furnace is shut down for complete maintenance. Therefore, the algorithm is executed only inside each useful life cycle period.

The method presented has been implemented in *Python* (Van Rossum and Drake, 2009) using the *Statsmodel* library for polynomial regression (Seabold and Perktold, 2010), the *Scikit Learn* library for Gaussian process regression (Pedregosa et al., 2011), *Matplotlib* for plotting, *NumPy* and *Pandas* libraries for general algebra operations (Harris et al., 2020; McKinney, 2010).

### 2.3. Gaussian process regression

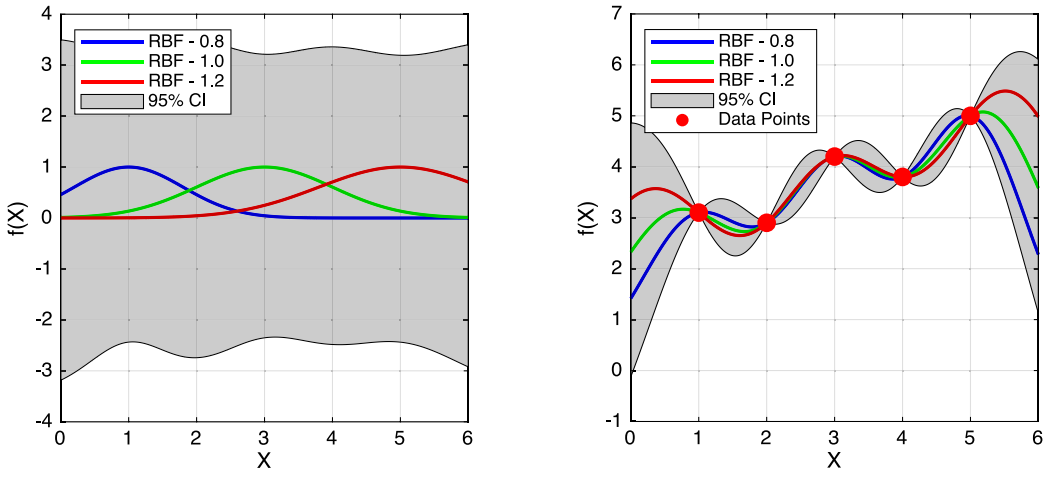
A Gaussian Process Regression (GPR) is a probabilistic, non-parametric, Bayesian approach for regression analysis. In a GPR model, the data is modeled as a function of random variables, which are assumed to follow a multivariate Gaussian distribution. The main idea behind GPR is to model the data as a function of some unknown function $f(x)$, and to specify a prior distribution over the function $f(x)$ that is Gaussian. The Gaussian process is defined by its mean function and covariance function, which together capture the prior knowledge about the data.

In GPR, the observed data are used to infer the posterior distribution over the unknown function $f(x)$. The posterior distribution is then used to make predictions about new, unseen data points. One of the main advantages of GPR is that it provides a flexible and intuitive way to model complex relationships in the data, without making strong assumptions about the underlying functional form, i.e. without specifying model parameters (Rasmussen and Williams, 2008).

A Gaussian Process (GP) is defined as

$$f(\mathbf{t}) \sim \mathcal{GP}(m(\mathbf{t}), k(\mathbf{t}, \mathbf{t}')) \tag{1}$$

where $m(\mathbf{t})$ and $k(\mathbf{t}, \mathbf{t}')$ are respectively the mean and covariance functions. Given two data points, a prior probability distribution is assigned to every function that could interpolate the points. Higher probabilities are assigned to functions that are considered to be more likely to do




(a) Gaussian Process Prior (b) Gaussian Process Posterior

**Fig. 1.** Comparative visualization of Gaussian Process (GP) utilizing Radial Basis Function (RBF) kernels with distinct characteristic lengths: 0.8, 1.0, and 1.2. The left-hand panel displays the GP prior distributions, capturing the initial assumptions or beliefs about the underlying function in the absence of observational data. Conversely, the right-hand panel illustrates the GP posterior distributions, which assimilate the observed data to refine and update the beliefs, thereby providing insights into potential function behaviors and uncertainties.

so given their properties. The combination of the data with the prior distribution leads to the posterior distribution which is the result of the regression.

A graphical representation of Gaussian process regression in the function-space view is reported in Fig. 1 (Rasmussen and Williams, 2008).

The prior distribution over functions is fully specified by a mean function and a covariance function. The covariance function, also known as the kernel function, plays a critical role in determining the smoothness and correlation structure of the functions that are generated from the prior distribution.

Specifically, the covariance function determines the degree of similarity between any two function values as a function of their input variables. This similarity or correlation is typically defined in terms of a distance measure between the inputs, and is often referred to as the kernel function.

The choice of the kernel function is therefore closely related to the prior assumptions that are made about the underlying data. For example, a smooth kernel may be used if the data is assumed to be smooth and continuous, while a periodic kernel may be used if the data exhibits periodic patterns.

The kernels adopted in this work are the Linear (LIN) kernel, the Radial Basis Function (RBF) kernel, and the Rational Quadratic (RQ) kernel, defined, respectively, as follows

$$k_{LIN}(t,t') = \sigma^2 + t \cdot t' \tag{2}$$

$$k_{RBF}(t,t') = \sigma^2 \exp\left(-\frac{(t-t')^2}{2\lambda^2}\right) \tag{3}$$

$$k_{RQ}(t,t') = \sigma^2 \left(1 + \frac{(t-t')^2}{2\alpha\lambda^2}\right)^{-\alpha} \tag{4}$$

where $k$ is the covariance function, $t$ and $t'$ are two points of the input space (time, in this case), $\sigma^2$ is the variance, $\lambda$ is the length scale parameter, and $\alpha$ is defined as the power of the RQ kernel.

The LIN kernel is more suitable to model linear behavior, while the RBF and RQ kernels are able to detect fluctuations and changes in length scales in the data. The base kernels are usually combined by addition and multiplication to form a complex kernel whose characteristics are retained from the base kernels from which it is composed. For example, the multiplication of two LIN kernels returns a kernel able to model a quadratic trend. A complex kernel will be more able to model intricate time series structures.

The properties of a kernel are defined by its hyperparameters such as the length scale $\lambda$ or variance $\sigma^2$. The optimal hyperparameters are estimated via the minimization of the Log Marginal Likelihood (LML), defined as

$$log\left(p(\mathbf{y}|\mathbf{t},\boldsymbol{\theta})\right) = -\frac{1}{2}\mathbf{y}^\top K_y^{-1}\mathbf{y} - \frac{1}{2}log|K_y| - \frac{n}{2}log(2\pi) \tag{5}$$

where $\mathbf{y}$ is the target variable, $K_y$ is the covariance matrix of the target and $p(\mathbf{y}|\mathbf{t},\boldsymbol{\theta})$ is the marginal likelihood, which is defined as the integral of the likelihood times the prior. For a deeper understanding of Gaussian Processes, the reader can refer to Rasmussen and Williams (2008).

Gaussian process regression is not without limitations. The main disadvantage of using a GPR is the computational time required for the calculation since it asymptotically scales as $\mathcal{O}(n_y^3)$, as $n_y \to +\infty$ where $n_y \in \mathbb{R}^+$ and $y \in \mathbb{N}$. The interpretation given to $n_y$ is that of the number of samples used for the computation. The high numerical complexity is due to the inversion of the covariance matrix $K_y$ in Eq. (5).

Furthermore, the fundamental formulation of GPR rests on two primary assumptions: normally distributed noise and homoscedasticity. The former suggests that all observation errors in GPR follow a normal distribution. Although this assumption might hold true for numerous systems, care should be taken when applying GPR to systems where the error could deviate from a normal distribution. Instances that might exhibit such behavior include transient operations, faulty measurements, or non-Gaussian processes, particularly those influenced by reactions. The latter assumption, homoscedasticity, asserts that the variance of the observations is consistent throughout the entire operational domain. Conversely, heteroscedasticity implies a non-constant variance, presenting challenges for GPR application. Possible sources of heteroscedasticity can be imprecise measurement systems, time-variant variance, or external disturbances.

### 2.4. Polynomial regression

Polynomial linear regression is a well-known, widely used technique for time series modeling. Despite being a simple approach, polynomial regression has many different engineering applications. The advantages of this method are its explainability along with the lower computational power required for calculation. Conversely, being a parametric model, it lacks flexibility thus making it less suitable for modeling complex time series data. A general polynomial with one input variable $x$ is defined as follows

$$y = \sum_{i=0}^{n_p} \theta_i x^i + \epsilon \tag{6}$$

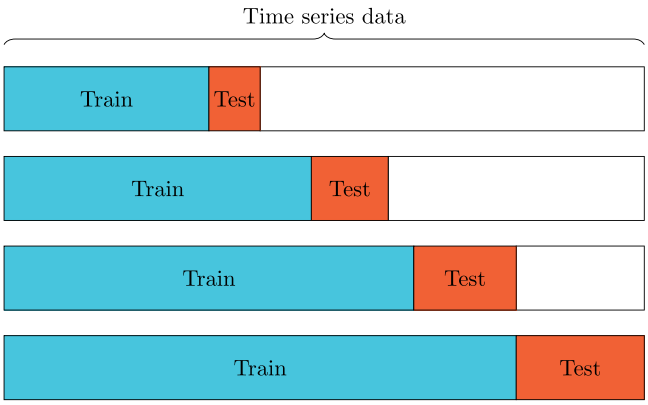



**Fig. 2.** Time series rolling cross-validation with 4 folds.

where $\theta$ are the features of the model, $n_p$ is the selected degree of the polynomial, and $\epsilon$ is the residual normally distributed error. By minimizing the squared residual between the prediction of Eq. (6) and the true values it is possible to find the parameters $\theta$ of the polynomial model.

The computational advantage of linear regression is known. Linear regression is efficient and scales as $\mathcal{O}(n_y \cdot n_p)$.

### 2.5. Models evaluation

The metric used to evaluate the performance of the models is the Mean Average Error (MAE), defined as

$$MAE = \frac{1}{n}\sum_{i=1}^{n_s} |y_i - \hat{y}_i| \tag{7}$$

where $n_s$ is the total number of samples, $y_i$ is the scalar value of the observation and $\hat{y}_i$ is the predicted value. The metric ranges between 0 and $+\infty$ and is a negatively oriented score, meaning that lower values are indicative of a better fit. MAE does not penalize large errors, thus making it more desirable when working with noisy datasets (Willmott and Matsuura, 2005; Willmott et al., 2009; Chai and Draxler, 2014). In the case of a k-fold cross-validation test, the global MAE is given by averaging the MAE calculated on every fold as

$$MAE_{CV} = \frac{1}{k}\sum_{i=1}^{k} MAE_i \tag{8}$$

where $k$ is the number of folds.

To assess the quality of fit of each model the classical $R^2$ metric is also presented, defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{9}$$

### 2.6. Model selection

For selecting the best-fit model, i.e. degree of the polynomial and GPR kernel, different training approaches are compared for both methods, (i) time series cross-validation (CV), (ii) ensemble forecast (E) and (iii) train/test split (nCV). Time series cross-validation, also referred to as rolling cross-validation, implies that data are split in $k$-folds following a strictly temporal order and not with random sampling since the dependency between data in time series is strong. In Fig. 2 this rolling cross-validation is shown for $k = 4$. An ensemble forecast is an ML technique in which weaker learning methods are combined to create a more robust ensemble model (Valentini and Masulli, 2002). In this case, the train set is sampled in two different groups given by data collected on even and odd days. Each model is trained for each group and the final ensemble model is the result of the average of the two models. At last, the train/test split approach consists in using a fraction of the available data (usually 80 %) for training and the remaining (20 %) for testing. The latter approach can be still considered a $k$-fold cross-validation approach where $k$ is simply equal to 1.

The models' performance on time-series data is evaluated on a rolling basis. For the GPR the best-performing kernel must be selected from the combinations of the tree described in Section 2.6.1. On the other hand, the degree of the polynomial is selected after a straightforward exhaustive search by computing all the polynomial linear models up to a specified order. The best-fit polynomial is then selected following the aforementioned procedures, based on the MAE values. The stopping criteria is polynomial degree 5, selected in order to avoid unreliable models for long-term predictions. The lower the order the more stable the prediction is.

#### 2.6.1. Gaussian process kernel selection

The training phase of the GPR consists of two steps: generation of kernels combination and estimation of hyperparameters via LML (Eq. (5)).

The kernels are generated using a purely combinatorial procedure on a search tree, shown in Fig. 3. At the first iteration, every selected basic kernel is combined using all the basic operators chosen. Redundant combinations are pruned. After this initial branching, a node is selected and from that node another vertical iteration is performed. At this point, the kernel of such node is combined with all the basic kernels and operators. This process is repeated up until the desired tree depth.

In particular, for the case study reported, only the *linear*, *radial basis function*, and *rational quadratic* kernels (See Eqs. (2), (3), and (4), respectively) are selected and the operators used are the *addition* and *multiplication*. A similar approach is described by Duvenaud et al. (2013).

A graphical representation of this algorithm is shown in Fig. 3. At each iteration, the best kernel is selected by picking the one with the lowest MAE and this is used as the starting branch for generating new combinations with all the basic kernels and the basic operators. All the other branches are discarded. In this way, a tree of combinations is generated and it is climbed using a greedy search strategy, i.e., only the local optimal solution at each layer is selected to keep on constructing the tree. This method does not guarantee global optimality and neither guarantees that kernels generated deeper into the tree are more accurate. In fact, the deeper the tree is investigated, the higher probability of overfitting occurs and the extrapolation capabilities are negatively affected. In case an adjacent LIN kernel addition is generated throughout the tree, it is pruned since the addition of two linear kernels is still a linear kernel, thus it does not have any effect.

It is worth noting that a global search strategy on the tree of combinations would probably give better results, on the test data set, in terms of accuracy but the computational expense required increases exponentially with the depth of the tree. Thus, as a compromise, the greedy search strategy has been selected instead. The tree depth defines how many vertical iterations are done in the kernel generation strategy. For this study, it has been set to 5 maximum iterations for numerical reasons. At iteration 5, the chosen kernel possesses at least 6 different kernels. For example, as opposed to the polynomial regression applied in this work, having 6 LIN kernels multiplied by each other translates to a polynomial of order 6. In comparison, the maximum degree of the polynomial that could be generated in this work is degree 5, as mentioned in Section 2.6.

As mentioned above, during branching it is possible that the best solution, in terms of lower MAE, remains on lower levels of the tree (i.e., with fewer kernels). In this case, the tree is still fully investigated up to the pre-defined search depth, following the algorithm described above, but the best kernel selected is the one coming from lower levels of the tree, as the best solution is not guaranteed to be on the last combinatorial layer.

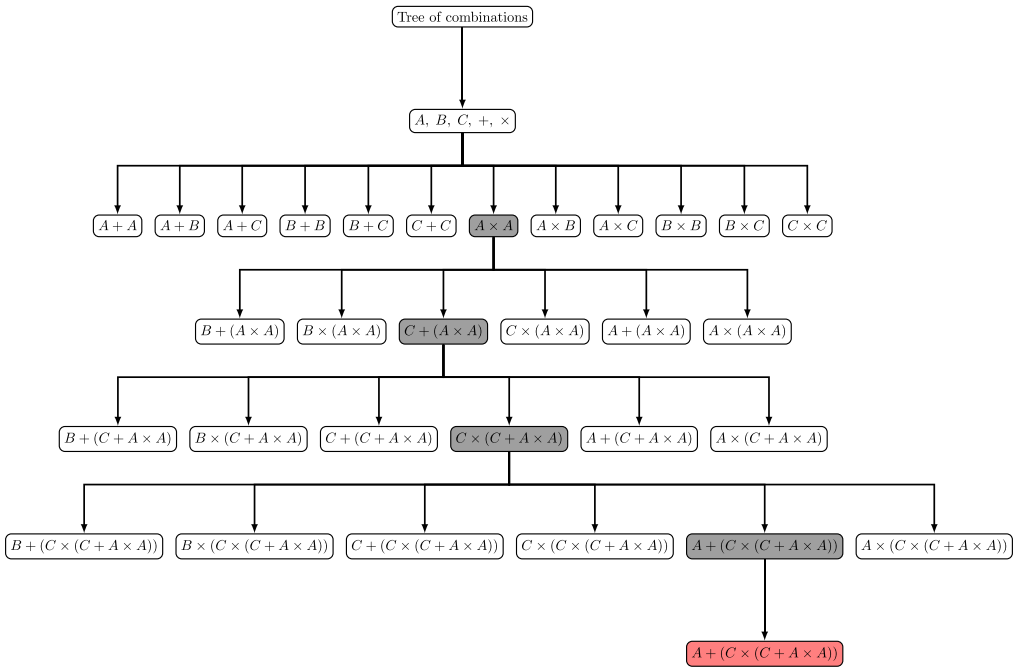



**Fig. 3.** Tree of combinations of general basic kernels A, B, and C (e.g. LIN, RBF, and RQ) using the *addition* and *multiplication* basic operators. The tree is explored following a greedy search strategy. In dark gray, the node selected at each iteration is highlighted while in red the final resulting kernel is shown.

The choice of tree depth stands as a crucial hyperparameter in the outlined method. Identifying the optimal value, though, relies largely on the complexity of the process being analyzed and leans on user insights and familiarity with the process. Problems of a more intricate nature are likely to benefit from an increased tree depth.

While determining the right hyperparameter for a new process, a retrospective application of this method could prove invaluable. Users are advised to commence by setting a moderately high tree depth and subsequently applying the algorithm to prior iterations of the process. Subsequent to this, a periodic prediction – for instance, on a daily basis – can be drawn for a designated time span. This horizon should take into account the timing of maintenance activities. Using cross-validation, these predictions can then be compared with actual process data. Subsequently, a collection of the most effective kernels is generated for every specified time-step. Users might then opt for a kernel tree depth that exceeds the deepest tree identified during this process by one order, adopting a conservative stance. It is essential to note, however, that employing this strategy necessitates having access to data from earlier process runs, which becomes instrumental when tweaking the presented algorithm. A general guideline would suggest operating with a tree depth that ideally ranges from 3 to 6.

## 3. Results & discussion

### 3.1. Industrial process

As mentioned in Section 2.2 the work focuses on the furnace PH-401B in the thermal de-asphalting section of the Itelyum Regeneration S.p.A refinery plant of Pieve Fissiraga in Lodi (Italy). In particular, the Key Performance Indicators (KPIs) evaluated for monitoring the performance of the furnace are the methane feed, the pressure drop of the charge tube side, and the tube skin temperatures. A qualitative scheme of the furnace is reported in Fig. 4.

The furnace can be divided into three sections upper, middle, and lower. Methane is fed to the lower section of the furnace making it the hottest. Methane feed is monitored through tag *FI-4091*. Conversely, the oil charge is fed to the upper part of the furnace. The total pressure drops of the charge are monitored by computing the difference between

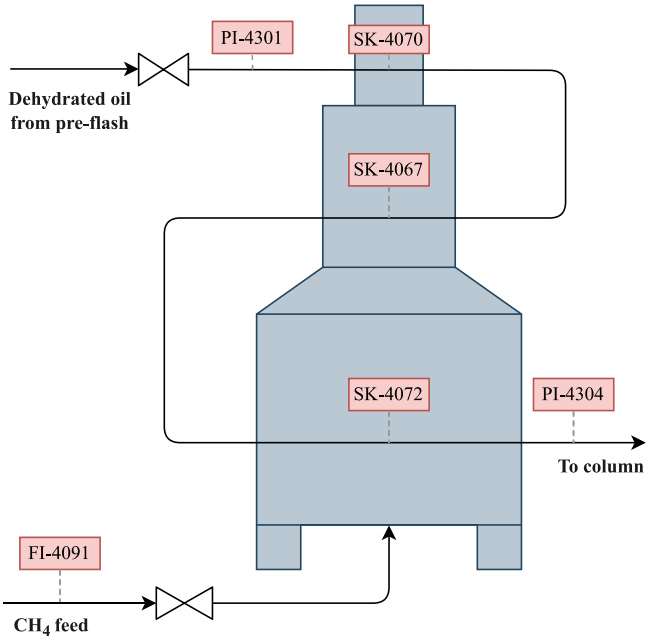



**Fig. 4.** Qualitative schematic representation of furnace PH-401B.

the tags at the inlet and outlet of the furnace, *PI-4301* and *PI-4304* respectively. Since *PI-4304* constantly remains very close to 0 barg, we will refer to the total pressure drop as *PI-4301*. For the tube skin temperatures, the measurement of one tube per section was considered so that every measurement is indicative of all the tubes belonging to that section. The tags for the tube skin temperatures are *SK-4072*, *SK-4067*, and *SK-4070* for the lower, middle, and upper parts of the furnace, respectively.

Fig. 4, from top to bottom, presents the time series plot of the methane feed *FI-4091*, pressure drop tube side *PI-4301* and tube skin temperatures *SK-4072, SK-4070, SK4067* of the tubes in the lower

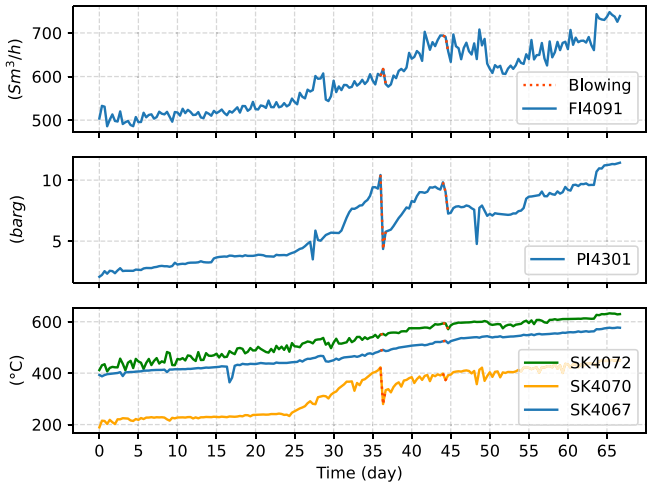



**Fig. 5.** From above, time series plot of $CH_4$ feed, pressure drop, and tube skin temperature for the lower, middle, and upper section of the furnace. The blowing procedure is highlighted in red.

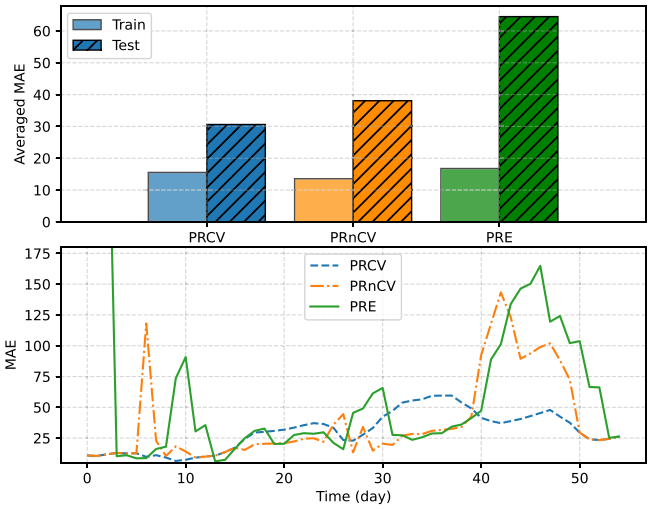



**Fig. 6.** Comparison of the performances of PRCV, PRnCV, and PRE.

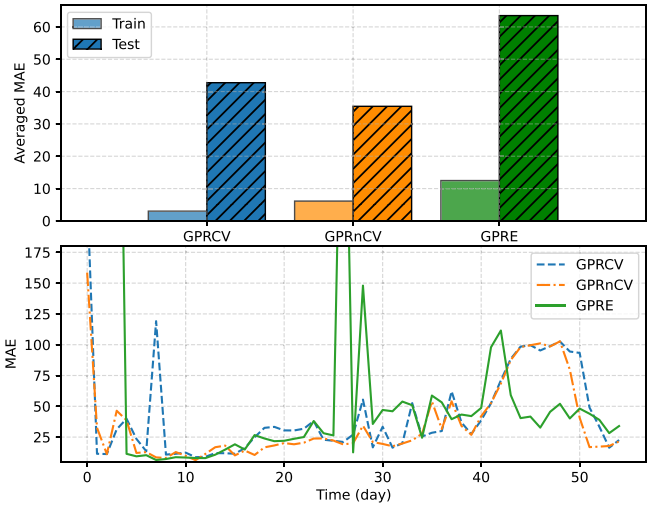



**Fig. 7.** Comparison of the performances of GPRCV, GPRnCV, and GPRE.

middle and upper section, in order. The inner part of the tubes suffers from coking given by the surface deposition of elemental carbon. This leads to an increasing pressure drop caused by the reduction of the section of the tubes. When *PI-4301* exceeds a pre-defined threshold, the coke is burned off with a dedicated procedure called blowing, during which high-pressure steam is injected inside the tubes for a certain amount of time. Since coke is not totally removed by this procedure, the pressure drop issue usually reappears after a short period, as shown in Fig. 5. The threshold value at which blowing is performed is around 10 barg. Clearly, the fouling process greatly affects methane feed and tube skin temperatures. The increase in methane consumption between the end of the run and the start of the run, while maintaining the same desired oil temperatures, is about 40%. During the blowing procedure, the methane feed is momentarily reduced, thus affecting the tube skin temperatures. When the coke buildup is considered excessive the furnace goes to a complete shut-down for maintenance. In the time frame under study, the furnace was stopped after 67 days of operations and two blowing procedures were performed on days 36 and 44.

### 3.2. Models selection comparison

The learning approaches described in Section 2.6 for selecting the best model are compared for both Polynomial Regression (PR) and Gaussian Process Regression (GPR) on the data set of the measured methane feed, tag *FI-4091*, shown in Fig. 5. In particular, the methods applied for both PR and GPR are the CV (time series cross-validation), nCV (train/test split) and E (ensemble). For example, the GPRE method is a Gaussian Process Regression with the Ensemble forecast approach. Tag *FI-4091* was chosen as the only target for this task since it is the one which shows a smoother behavior and fewer discontinuities.

In Figs. 6 and 7, the results of this comparison using polynomial regression and Gaussian process regression are reported, respectively.

The metric applied to analyze the methods is the MAE, defined in Eq. (7). In this case, the MAE is evaluated daily by comparing the value of the raw data and the model prediction after 10 days. In the upper plot, the average MAE value of all the daily 10-day-ahead predictions is shown. In the lower plot, this daily MAE is reported.

By inspecting the first plot of Fig. 6 it seems that the PRnCV model yields a better performance on average. Moreover, the second plot reveals that PRnCV is more stable and less subject to sudden changes in the time series, given that it shows fewer spikes with respect to the train/test split and Ensemble methods. In fact, the rolling cross-validation split encourages less complex models since the resulting degree of the polynomial is always of the first order. Even though this results in a lower MAE, the model is not able to follow higher-order trends, if present, e.g. quadratic trends. On the other hand, PRnCV and PRE choose different orders for the polynomial model at each iteration and the large MAE spikes are caused by models with degree greater than one, which rise more steeply than a linear first-order model during extrapolation.

In Fig. 7, the results of the GPR are reported. In this case, the GPRnCV has a lower average MAE, compared to the other two methods. Also, its daily MAE exhibits fewer fluctuations. The results of the ensemble method are underwhelming since they are more prone to create large localized errors, for example on days 25 to 30. This is the case since no cross-validation is performed on the data and even though two models are added linearly by averaging (as explained in Section 2.6) the sudden changes to the trend significantly affect the optimized kernels chosen by the tree search and result in over-fitting the data. The CV and nCV methods in this case have more comparable results but as it is shown in the upper plot of Fig. 7 the PRnCV method is slightly better.

For the aforementioned reasons, even though the numbers might show better performance for the CV method on the PR case, a more

holistic analysis tells that the most plausible suitable candidate is the nCV, or train/test split, method. For example, the approaches with CV and ensemble methods seem to give sub-optimal results more frequently. The former usually chooses lower order models (less complex kernel or first order polynomial), thus keeping the error low but being less able to extrapolate on future data. The main problem with the CV and ensemble approaches is probably caused by a lack of data in the training set. Indeed, the models start to converge for the last iterations which are the ones in which more data are used for the training of the model.

### 3.3. Results on the case study

In Section 3.2, the one-fold cross-validation, i.e., the simple train/test 80/20 split, conveniently referred to as "no cross-validation" (nCV), has been selected as the approach to model the time series data of interest. Hereafter, Gaussian process regressions (GPRnCV) and polynomial regressions (PRnCV) are explored and compared in detail on the time series of *FI-4091*, the pressure drop *PI-4301*, and the tube skin temperature of the middle section *SK-4067*.

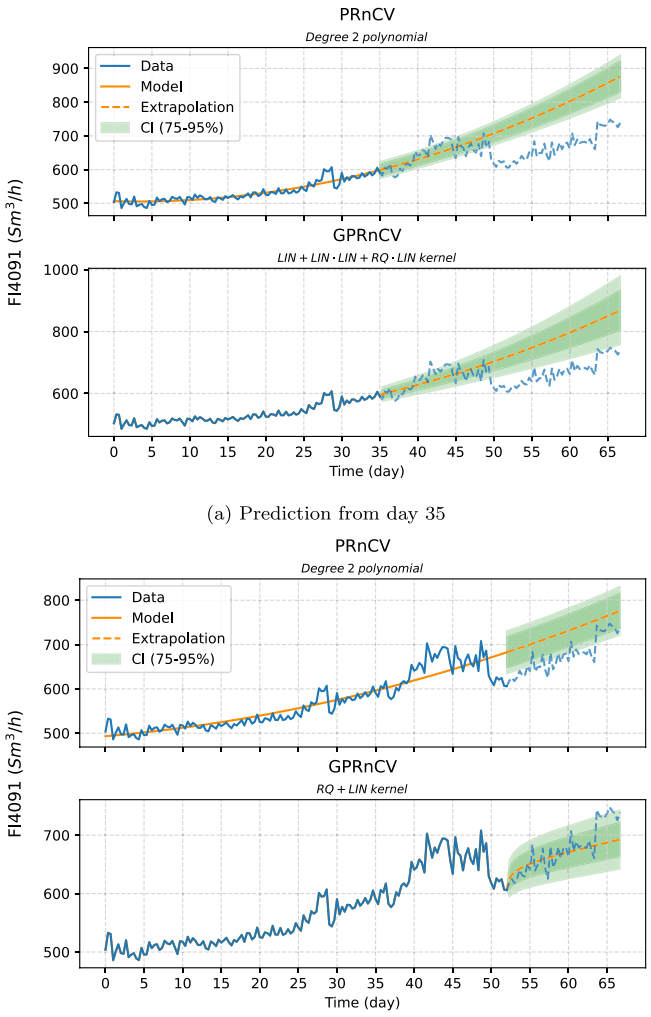
#### 3.3.1. Methane feed: FI-4091

In Fig. 8, the predictions obtained by the models GPRnCV and PRnCV on *FI-4091* are compared. The dashed line in orange represents the extrapolation returned by the model, while the dashed blue line shows the unseen data, i.e., data not used during training or testing the model. Fig. 8(a) shows the prediction obtained 35 days after the startup, while Fig. 8(b) has been obtained 52 days after the startup. Until day 42, the time series follows an almost quadratic trend that is indeed identified by both the PRnCV and the GPRnCV models, as is shown in Fig. 8(a). Within this time frame, the optimization procedure described in Section 2.6.1 selected combinations that are mainly composed of multiplication of LIN kernels, as reported in Fig. 8(a). As mentioned in Section 2, the multiplication of $n$ LIN kernels together results in a structure equivalent to that of a polynomial of degree $n$. Thus, the optimal GPR selected can effectively model the quadratic behavior of the trend, as the PR does, by multiplying together two linear kernels. Moreover, the addition of the RQ·LIN kernel models the small fluctuations in the data that, however, are not extrapolated in the long-term prediction, given the resulting small length-scale hyperparameter $\lambda$ assigned to the RQ kernel.

In this case, the extrapolation from day 35 onward seems to agree with future data almost until day 50 (see Fig. 8(a)), which is an extrapolation period of 15 days. Due to the almost quadratic trend with no major disruptions, both PR and GPR models can make a reasonably good prediction inside this window, especially if we account for the confidence intervals, which could be considered narrow for this case.

However, several days after the two blowing procedures, on day 52, it is possible to grasp the potential of the Gaussian process regression (see Fig. 8(b)) since, being non-parametric, it is able to better follow the structure of the time series and more easily adjust to sudden changes in data. On the other hand, the polynomial regression is not able to swiftly adjust itself given the inertia caused by the weight of past data and its parametric nature. By looking closer at the plots of Fig. 8(b) it is possible to notice that the mean projection of the GPR model is almost perfectly aligned, except for sudden shifts which are still falling inside the confidence intervals, to the process data until the end of day 63, which is almost a 12 days prediction horizon. Conversely, as already discussed, the polynomial model was not able to follow the trend shift, thus, is not able to accurately predict the following 12 days. However, it might seem that the polynomial regression does a better job at predicting the end-of-run values but that is not useful if the immediate next days will never be accurately predicted, and it might as well be just a matter of chance.

Table 1 reports the metrics obtained by the two predictions for both the GPRnCV and PRnCV models. It comes as no surprise that the $R^2$ of GPRnCV is always close to 100 % and surely higher than the one of PRnCV given the nature of the GPR itself, described in Section 2.3. However, the ability to fit past data is of no interest to anyone in this case. Thus, the most relevant metric to look after is the MAE on the test set, which should be more indicative of the quality of the extrapolation; during the first training of the models this data is unused, as explained in Section 2.6. For day 35, the values of the test MAE for both models are very close but on day 52, some days after the human intervention, i.e. the blowing procedure, the GPR shows a higher performance. In this case, the GPR seems the better choice, given the fact that it can always fall back to a more traditional polynomial regression when combining linear kernels by multiplication. In fact, until day 35, the kernel optimization strategy can find a kernel that is both non-complex in nature and yields good results.




(a) Prediction from day 35

(b) Prediction from day 52

**Fig. 8.** Comparison of predictions by GPRnCV and PRnCV on methane feed to furnace *FI-4091*.

**Table 1**
Accuracy sample report on methane feed *FI-4091*.

| Days after startup | Model | $R^2$ train | MAE train | MAE test |
|---|---|---|---|---|
| 35 | GPRnCV | 96.53% | 0.022 | 20.476 |
| | PRCV | 84.61% | 8.867 | 19.787 |
| 52 | GPRnCV | 99.93% | 1.223 | 14.398 |
| | PRCV | 86.97% | 15.314 | 59.726 |

**Table 2**
Accuracy sample report on pressure drop *PI-4301*.

| Days after startup | Model | $R^2$ train | MAE train | MAE test |
|---|---|---|---|---|
| 33 | GPRnCV | 98.34% | 0.108 | 0.770 |
| | PRnCV | 87.18% | 0.490 | 2.677 |
| 39 | GPRnCV | 96.53% | 0.128 | 1.445 |
| | PRnCV | 96.52% | 0.144 | 2.473 |

### 3.3.2. Pressure drop: PI-4301

The pressure drop trend shows a smaller length scale, if compared to the feed methane time series. For this reason, the limitations of the parametric model resulting from PRnCV are more visible, as shown in Fig. 9 which compares the predictions obtained 33 and 39 days after the startup of the furnace.

At day 33, as reported in Fig. 9(a), both models can predict well the behavior of both past data and future data, up until the small maintenance on day 36. After day 36, any prediction made before that day is expected to fail since, if not purposely implemented, no model can make such a prediction. However, it is probably worth noting that even in this scenario the confidence interval of the GPR becomes much wider right after that day and this could probably warn the user in advance of a possible unexpected, or rapid, change in the underlying process. In this case, we could guess that the coke build-up could have unexpectedly and suddenly increased to a great extent.

Day 39 was chosen in order to compare the models on their prediction abilities after the sudden change in pressure caused by blowing. As shown in Fig. 9(b), the PRnCV model is trying to minimize the sum of squares by tracking a line that goes across the data. On the other hand, the flexibility of the GPRnCV model can fit the sudden decrease caused by blowing on day 36 and its subsequent increase. Even though the mean prediction of the GPR is not spot-on on the immediate days, as is unexpectedly the one from the polynomial regression, its confidence interval is still reliable and, most importantly, the long-run prediction is closer to reality. In fact, the values predicted by the optimized GPR are extremely close to the actual data, even after 24 days. This might be a matter of chance, but, in any case, the most important aspect is that the confidence interval is still reliable, although it is not narrow enough probably because GPR has not had time to settle to the new sudden change. The composition of the kernel reported in Fig. 9(b) shows a single linear kernel and many other smaller-scale kernels. This suggests that the long-run behavior is that of a linear model.

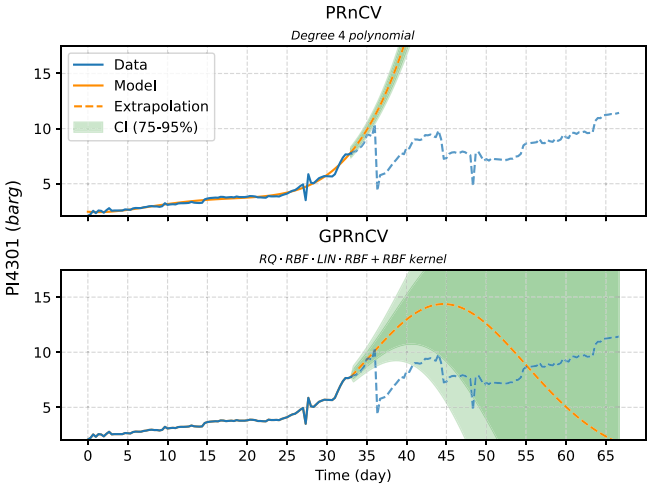
In Table 2 the metrics resulting from the predictions returned by the models for the two cases are reported. The results show that both models can describe the majority of the variability on the training set for both cases. Nevertheless, by comparing the values of MAE on the test set, GPRnCV shows better prediction performance.

### 3.3.3. Tube skin temperature: SK-4067

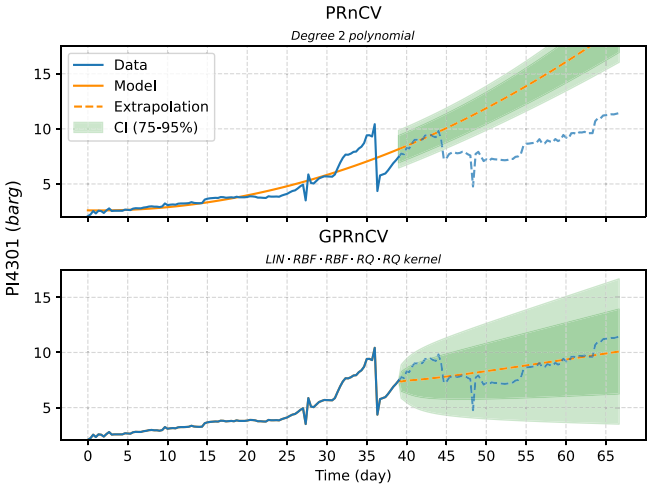
In Figs. 10(a) and 10(b) the predictions obtained 22 and 53 days after the startup respectively, are presented. In the case of Fig. 10(a), both the GPRnCV and PRnCV models can properly predict the trend of the time series; however, only the GPRnCV model is able to fit the unexpected decrease in temperature at day 16, for which no assignable cause is available. This results in an optimized kernel with a wider confidence interval in the long run. Also, in this case, the confidence intervals of the GPRnCV appear to be more reliable for long-term predictions.

Again, Fig. 10(b) shows the ability of the GPRnCV model to be more responsive to the changes in the time-series data. In this case, the changes are smoother. The PR model is not able to overcome the weight of past data and morph the shape to better follow the new trend. In contrast, the GPR, especially given the kernel search optimization, is able to present a more accurate and reliable mean prediction.

In Table 3, the metrics of the predictions obtained for the tube skin temperature *SK-4067* are reported. The value of the $R^2$ for the PRnCV




(a) Prediction from day 33




(b) Prediction from day 39

**Fig. 9.** Comparison of predictions by GPRnCV and PRnCV on pressure drops *PI-4301*.

**Table 3**
Accuracy sample report on tube skin temperature *SK-4067*.

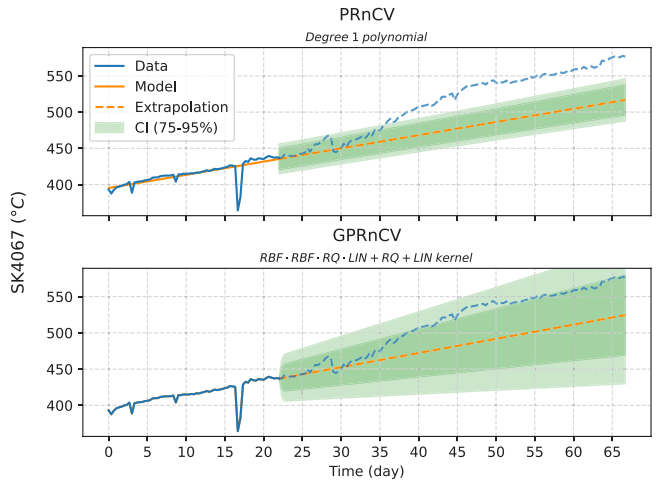
| Days after startup | Model | $R^2$ train | MAE train | MAE test |
|---|---|---|---|---|
| 22 | GPRnCV | 96.35% | 1.692 | 4.975 |
| | PRnCV | 57.58% | 4.252 | 6.289 |
| 53 | GPRnCV | 99.65% | 1.309 | 1.210 |
| | PRnCV | 97.07% | 30.875 | 5.329 |

model is strongly affected by the sudden decrease in temperature on day 16. Again, by comparing the MAE values on the test set, it is possible to conclude that the prediction ability of the GPRnCV model seems higher than the one of the PRnCV model.

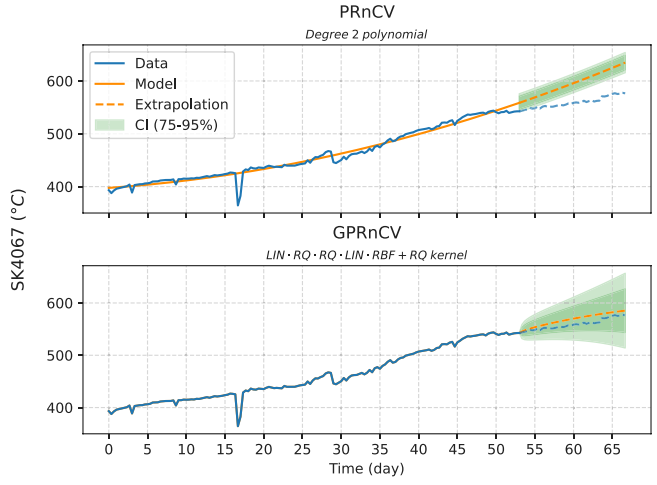
### 3.4. Evaluation of long-term predictions

In order to assess the end-of-run prediction accuracy of both GPRnCV and PRnCV methods, a simulation of daily rolling predictions is performed. Thus, the models are trained to simulate the daily availability of new data. The mean prediction on the day of shutdown is then compared to the actual process data. The metric applied for the comparison is still the MAE, as described in Section 2.5.

This analysis is only indicative of the prediction capacity on the end values and may not give insights on local or smaller horizons.




(a) Prediction from day 22




(b) Prediction from day 53

**Fig. 10.** Comparison of predictions by GPRnCV and PRnCV on tube skin temperature *SK-4067*.

While sudden changes to the final raw values may alter the results, this has been ruled out by inspecting the available data, shown in Fig. 5. For a more reliable result, a moving average approach or average step function over multiple days could be applied both to the prediction and to raw data. However, in these cases it is less evident if the convergence of the models is moving rapidly or not and the final results may not show a significant difference.

In Fig. 11 the results obtained for every simulated daily iteration on the variables of interest *FI-4091*, *PI-4301*, and *SK-4067* are presented. The $x$-axis reports the days after the startup of the furnace. The dashed line labeled *Convergence* indicates the value of the MAE towards which the models tend, while the shaded region indicates the period for which the models are far from convergence. From the plot of *FI-4091*, it is possible to see that both the GPRnCV and PRnCV models stabilize to an almost constant MAE after 52 days following the startup of the furnace. In other words, both models are able to predict the final state of the methane feed 15 days in advance with a mean average error of about $30\,\mathrm{Sm}^3/\mathrm{h}$.

The high instability of the models before day 52 is given by the strong irregularities in the time series thus affecting both models' capacity to make a reasonable prediction of the end state.

For what concerns the pressure drop *PI-4301*, both the GPRnCV and PRnCV models converge to a value of the MAE at around 1 barg. The GPRnCV model is able to return a good prediction 29 days in advance while the PRnCV model is only 22 days in advance.

Finally, the tube skin temperature before shutdown is adequately predicted 17 days before by the GPRnCV model and 10 days before by the PRnCV model with a value of the MAE lower than 10 °C.

In this case study, the criterion for shutdown is based on the pressure differential, as measured by the tag *PI-4301*. In the proposed run, Itelyum decided upon a shutdown threshold of approximately 11 barg on day 67 and 36. To evaluate the predictive maintenance capabilities of the proposed method, we conducted a retrospective daily prediction. The results of this assessment are depicted in Fig. 12.

The outcomes presented correlate closely with those in Fig. 11. However, a key distinction lies in the nature of the predictions. While the former figure illustrates average predictions assessed via MAE, the latter showcases specific values. A closer examination of Fig. 12 reveals that the algorithm initially takes a more conservative stance, predicting a shutdown between days 110 and 120. By day 29, just 10 days before the blowing procedure, the algorithm starts to align more with the actual first shutdown day, i.e. the day the blowing procedure was executed.

Post this intervention, the model gradually approaches the real shutdown day. Although a few outliers (on days 49–51) caused by unanticipated pressure drops at day 47 are visible in Fig. 5, predictions become increasingly dependable as the shutdown day approaches, especially from almost 20 days prior.

It is important to emphasize that, in some situations, the upper uncertainty boundary can tend towards infinity, particularly in the days leading up to the shutdown. This behavior stems from the inherent nature of the problem. The uncertainty can be visualized as a two-dimensional cone, and the upper boundary for the predicted shutdown day is identified by intersecting this cone with a line. If the cone bisector tends to align horizontally, the intersection on its rightmost side could be exponentially distant. If the cone tilts too dramatically, such an intersection might become non-existent.

Furthermore, some data points in Fig. 12 are absent due to poor algorithm convergence to models. In other words, certain kernels, chosen based on cross-validation, do not predict a shutdown as they anticipate decreasing values over time. This phenomenon is evident around days 14–15, 20, and 23–26. This shortfall could be attributed to the algorithm's greedy search approach, which occasionally yields subpar results. An alternative could involve refining the search strategy, employing more exhaustive heuristic methods, or reformulating the actual MINLP problem with appropriate approximations to achieve a solution nearer to the global optimum. Overcoming computational constraints might be feasible through the full deployment of parallel computing.

### 3.5. Limitations

The approach described here has some limitations related to the quality of the time series data. Predictions may become more unreliable in the presence of non-smooth data and unexpected external interventions, such as human actions or equipment failure. For instance, Fig. 9(a) illustrates a sudden gradient increase starting from day 30, resulting in future confidence intervals that are excessively wide and an inaccurate mean prediction.

Focusing on tag *PI-4301*, shown in Figs. 5 and 9, it can be seen how the trend disruption at day 36 can affect future predictions. From a purely theoretical point of view, the trend that generating after day 36 should be dissociated from the past trend, due to the blowing intervention, which suddenly changes the boundary conditions for the process unit operation. If we were to consider this, the regression should be interrupted at day 36 and a new regression must be performed with data subsequent to that date. However, the results of this approach, as shown in Fig. 13, may not be considered completely satisfactory.

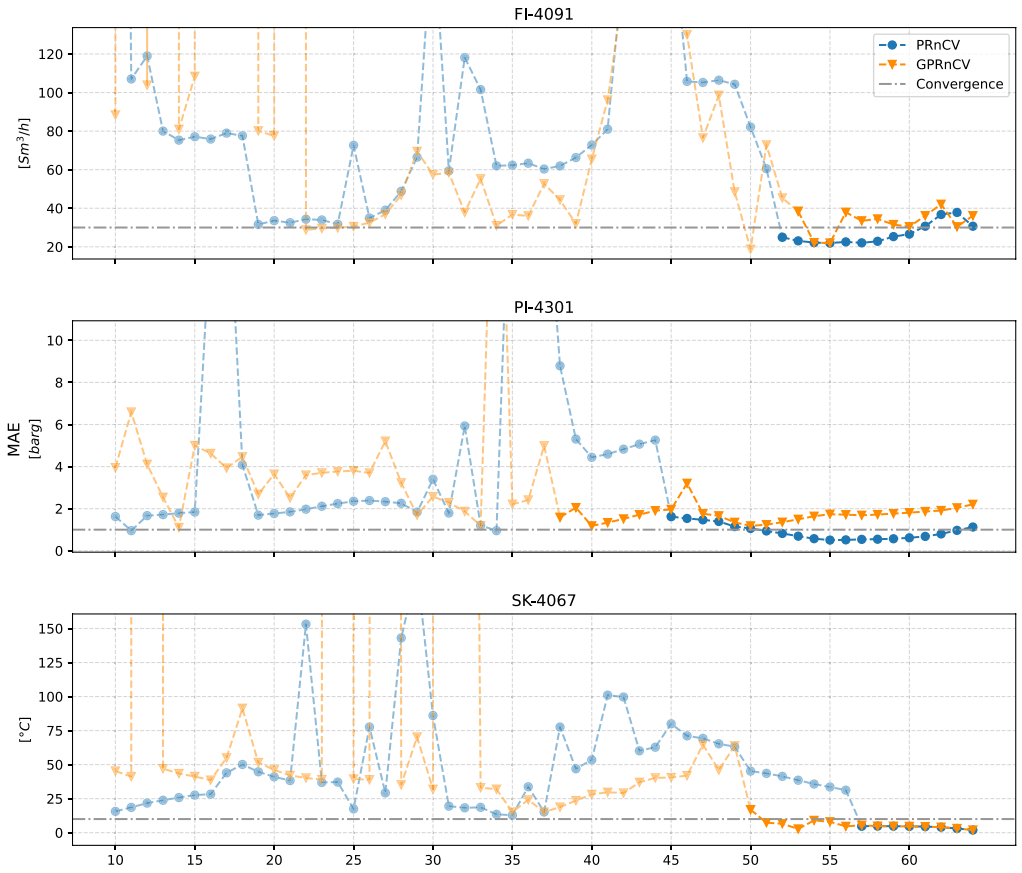



**Fig. 11.** Daily rolling means average prediction error to the end of the run value, starting from day 10. Far from convergence daily predictions are represented in a lighter color.

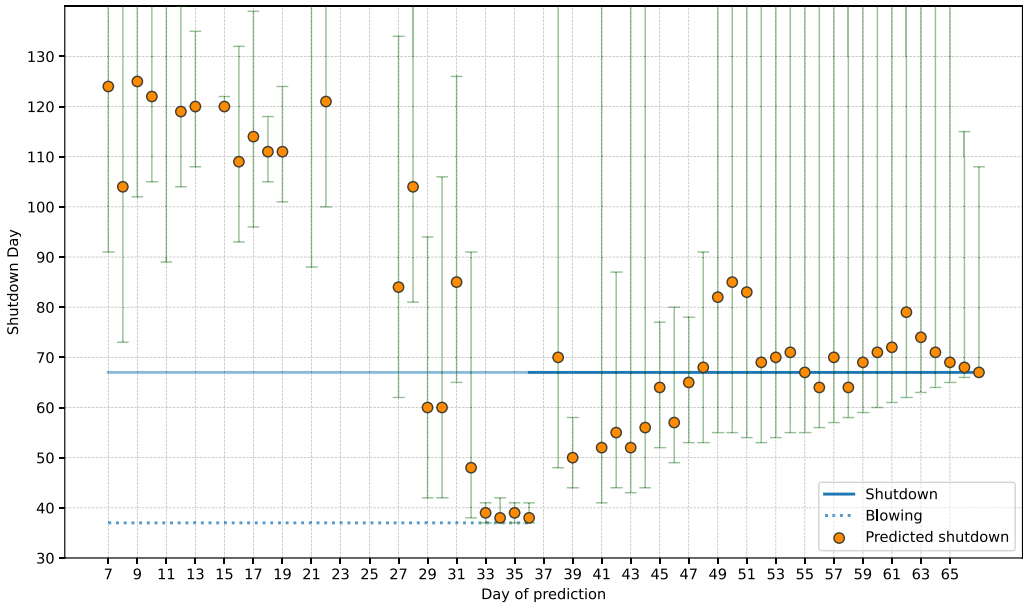
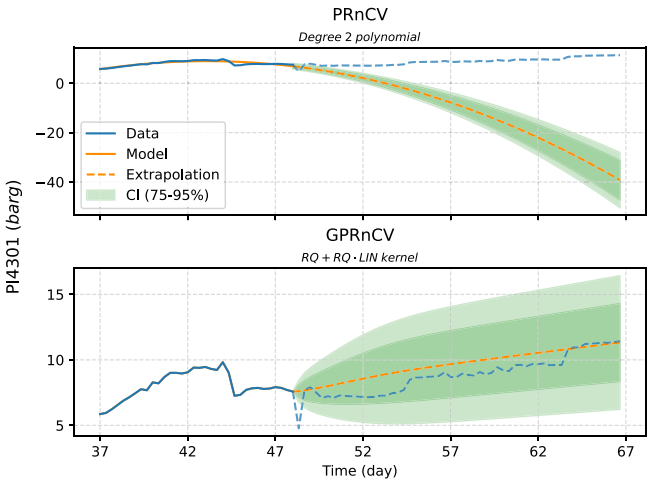


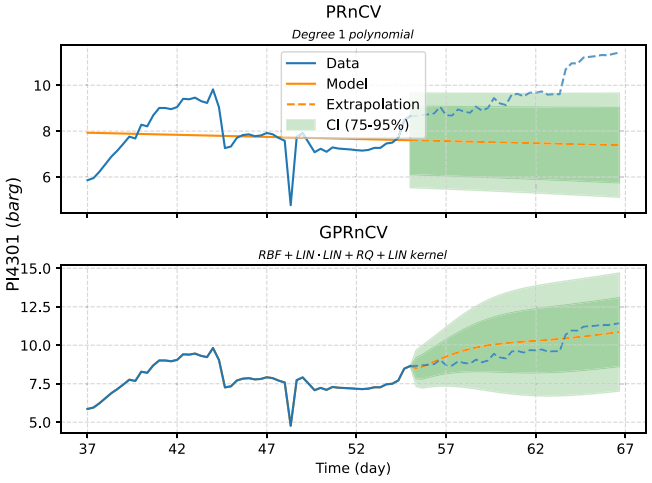

**Fig. 12.** Daily rolling shutdown prediction using GPRnCV on tag *PI-4301* with uncertainty, starting from day 7 until the day of shutdown, day 67.

At first, the predictions attain coherence solely after day 48, and only for GPR. Due to the limited number of data points available, any predictions made before that day cannot be considered entirely trustworthy. In comparison, if applying a single regression as shown in Fig. 9(b), at day 39 it is already possible to obtain a reasonable prediction. This is an 11 day lag and might be considered too long for this process, especially near the end of the run where the predictions must be more accurate.

Secondly, the polynomial regression method struggles to adapt to trends when there are few data points, especially during the period from day 37 to day 67 which exhibits a high degree of non-linearity, see Fig. 13. Once again, the Gaussian process approach demonstrates




(a) Prediction from day 48




(b) Prediction from day 55

**Fig. 13.** Comparison of predictions by GPRnCV and PRnCV on pressure drops PI-4301, after day 36.

its superiority over the polynomial one in adapting to complex trends with limited data. As seen in Fig. 13(b), the Gaussian process model produces near-perfect predictions, while PRnCV is only able to identify an almost horizontal linear model.

### 3.6. Final remarks

The real-world scenario presents several challenges for successful forecasting, i.e., irregular patterns given by unexpected trends, sudden mean shifts, complex seasonality, and outliers. On top of that, there are significant fluctuations and noise. For example, by looking at the time series of the methane feed (Fig. 5 or 8), significant mean shifts are seen after days 30 and 50 for which no assignable cause exists in the raw extracted data. Due to these anomalies, the development of a predictive model for extrapolation of the time series pattern is intricate work, especially for long-term forecasting. However, the data-driven approach seems a reasonable way to develop automatic procedures that are able, for the most general cases, to be agnostic regarding the process or the variable under investigation.

In this study, it has been shown that the Gaussian process regression shows more reliability in long-term predictions and more precision in short-term ones. The optimization of kernels is what really brings impact to the performance of the GPRs and this results in more reliable and comprehensive confidence intervals.

The evaluation of long-term predictions in Section 3.4 indicates that both methods yield accurate predictions up to 10–29 days before shut-down, depending on the process variable. However, as shown in previous sections, Gaussian process regression generally outperforms polynomial regression, particularly for trends that are prone to unexpected shifts. Additionally, the confidence interval generated by Gaussian process regression is more reliable in the majority of cases. Moreover, Gaussian process regression predictions are more reliable on a day to day basis, especially for the short subsequent time period.

Thus, the choice of using the Gaussian process regression over polynomial regression should be straightforward, and the optimization of the kernel composition is highly advised. For this reason, the cross-validation procedure, to select one model among many, is crucial and the choice made after the results of Section 3.2 is that of using one-fold cross-validation instead of the many-folds rolling cross-validation, since it is more nimble and can help in adapting faster to macro trend changes.

In the end, if needed, a GPR can still fall back to a simple polynomial regression as discussed in Section 2.3.

## 4. Conclusions

In conclusion, maintenance of industrial units, even the smaller ones, requires good management of resources and workforce. Knowing well in advance when it would be the right time to stop operations and apply maintenance is highly important but also complex. The work presented shows the development of a purely data-driven machine learning tool to aid in predicting the required maintenance operations of this process furnace (located in the thermal de-asphalting section of the exhausted oil regeneration plant of Itelyum in Pieve Fissiraga, Lodi, Italy). In this study, it was shown that a reasonable prediction horizon for the assessed unit is between 10 to 30 days. For larger units, e.g. large-scale industrial turbines, this prediction horizon might not be long enough to plan ahead the required maintenance procedure. However, on a unit such as the direct-fired heater presented, it is enough to organize and plan the workload appropriately. Still, the order of magnitude of the prediction horizon is comparable to that of the total usual run cycle time.

In particular, on a day to day rolling basis the raw time series data is analyzed, and using a one-fold rolling cross-validation method the degree of the polynomial in the polynomial regression is selected by looking at the mean average error metric. The same approach is applied to the selection of the kernels of the Gaussian process regression by performing a greedy search optimization on a combination of linear, rational quadratic, and radial basis function kernels. It was demonstrated that Gaussian processes bear several major advantages with respect to polynomial regression, and they are more suitable for modeling time series data with high irregularities and no well-defined structure, such as this chemical process.

### Acronyms

| | |
|---|---|
| **CV** | Cross Validation |
| **DT** | Digital Twin |
| **GPR** | Gaussian Process Regression |
| **GPRCV** | GPR with 5-fold rolling CV |
| **GPRE** | GPR with Ensemble method and no CV |
| **GPRnCV** | GPR with 1-fold CV |
| **LIN** | Linear kernel |
| **PR** | Polynomial Regression |

| | |
|---|---|
| **PRCV** | PR with 5-fold rolling CV |
| **PRE** | PR with Ensemble method and no CV |
| **PRnCV** | PR with 1-fold CV |
| **RBF** | Radial Basis Function kernel |
| **RQ** | Rational Quadratic kernel |

## CRediT authorship contribution statement

**Andrea Galeazzi:** Conceptualization, Methodology, Software, Investigation, Writing – original draft, Writing – review & editing. **Francesco de Fusco:** Software, Investigation, Writing – original draft, Writing – review & editing. **Kristiano Prifti:** Methodology, Writing – original draft, Writing – review & editing. **Francesco Gallo:** Conceptualization, Data acquisition, Writing – original draft, Writing – review & editing. **Lorenz Biegler:** Conceptualization, Methodology, Investigation, Writing – original draft, Writing – review & editing. **Flavio Manenti:** Conceptualization, Methodology, Supervision, Writing – original draft, Writing – review & editing.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Data availability

Data will be made available on request.

## References


Aminian, J., Shahhosseini, S., 2008. Evaluation of ANN modeling for prediction of crude oil fouling behavior. Appl. Therm. Eng. 28 (7), 668–674. http://dx.doi.org/10.1016/j.applthermaleng.2007.06.022.

Baldo, V., Gallo, F., 2020. Regeneration of used oils. EP3683295A1.

Becherer, M., Zipperle, M., Karduck, A., 2020. Intelligent choice of machine learning methods for predictive maintenance of intelligent machines. Comput. Syst. Sci. Eng. 35 (2), 81–89.

Bogojeski, M., Sauer, S., Horn, F., Müller, K.R., 2021. Forecasting industrial aging processes with machine learning methods. Comput. Chem. Eng. 144, 107123. http://dx.doi.org/10.1016/j.compchemeng.2020.107123.

Carvalho, T.P., Soares, F.A.A.M.N., Vita, R., Francisco, R.d.P., Basto, J.P., Alcalá, S.G.S., 2019. A systematic literature review of machine learning methods applied to predictive maintenance. Comput. Ind. Eng. 137, 106024. http://dx.doi.org/10.1016/j.cie.2019.106024.

Chai, T., Draxler, R.R., 2014. Root mean square error (RMSE) or mean absolute error (MAE)? – arguments against avoiding RMSE in the literature. Geosci. Model Dev. 7 (3), 1247–1250. http://dx.doi.org/10.5194/gmd-7-1247-2014.

Cline, B., Niculescu, R.S., Huffman, D., Deckel, B., 2017. Predictive maintenance applications for machine learning. In: 2017 Annual Reliability and Maintainability Symposium. IEEE, New York.

Dalla Giovanna, F., 2012. Lubricants recycling – a case study: How Italy managed to become an excellence and an example for the other EU's member states. In: Bilitewski, B., Darbra, R.M., Barceló, D. (Eds.), Global Risk-Based Management of Chemical Additives I: Production, Usage and Environmental Occurrence. In: The Handbook of Environmental Chemistry, Springer, Berlin, Heidelberg, pp. 225–251. http://dx.doi.org/10.1007/698_2011_100.

Dekker, R., Scarf, P., 1998. On the impact of optimisation models in maintenance decision making: The state of the art. Reliab. Eng. Syst. Saf. 60 (2), 111–119. http://dx.doi.org/10.1016/S0951-8320(98)83004-4.

Duvenaud, D., Lloyd, J.R., Grosse, R., Tenenbaum, J.B., Ghahramani, Z., 2013. Structure discovery in nonparametric regression through compositional kernel search. http://dx.doi.org/10.48550/arXiv.1302.4922, arXiv arXiv:arXiv:1302.4922.

Efeoglu, E., Tuna, G., 2022. Machine learning for predictive maintenance: Support vector machines and different kernel functions. J. Mach. Manuf. Reliab. 51 (5), 447–456. http://dx.doi.org/10.3103/S1052618822050041.

Florian, E., Sgarbossa, F., Zennaro, I., 2021. Machine learning-based predictive maintenance: A cost-oriented model for implementation. Int. J. Prod. Econ. 236, 108114. http://dx.doi.org/10.1016/j.ijpe.2021.108114.

Frigola, R., 2015. Bayesian Time Series Learning with Gaussian Processes (Ph.D. thesis). University of Cambridge, http://dx.doi.org/10.17863/CAM.46480.

Gallo, F., Rossi, C.O., Porto, M., Baldo, V., 2021. Valorizzazione dei sottoprodotti di un processo di rigenerazione di olii usati. IT202000001357A1.

Ge, Z., Chen, T., Song, Z., 2011. Quality prediction for polypropylene production process based on CLGPR model. Control Eng. Pract. 19 (5), 423–432. http://dx.doi.org/10.1016/j.conengprac.2011.01.002.

Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., Berg, S., Smith, N.J., Kern, R., Picus, M., Hoyer, S., van Kerkwijk, M.H., Brett, M., Haldane, A., del Río, J.F., Wiebe, M., Peterson, P., Gérard-Marchant, P., Sheppard, K., Reddy, T., Weckesser, W., Abbasi, H., Gohlke, C., Oliphant, T.E., 2020. Array programming with NumPy. Nature 585 (7825), 357–362. http://dx.doi.org/10.1038/s41586-020-2649-2.

Hichri, B., Driate, A., Borghesi, A., Giovannini, F., 2022. Predictive Maintenance Based on Machine Learning Model. In: Maglogiannis, I., Iliadis, L., Macintyre, J., Cortez, P. (Eds.), Artificial Intelligence Applications and Innovations, Aiai 2022, Part II, Vol. 647. Springer International Publishing Ag, Cham, pp. 250–261. http://dx.doi.org/10.1007/978-3-031-08337-2_21.

Kobbacy, K.A.H., Murthy, D.N.P., 2008. An overview. In: Kobbacy, K.A.H., Murthy, D.N.P. (Eds.), Complex System Maintenance Handbook. In: Springer Series in Reliability Engineering, Springer, London, pp. 3–18. http://dx.doi.org/10.1007/978-1-84800-011-7_1.

Kupareva, A., Mäki-Arvela, P., Murzin, D.Y., 2013. Technology for rerefining used lube oils applied in Europe: A review. J. Chem. Technol. Biotechnol. 88 (10), 1780–1793. http://dx.doi.org/10.1002/jctb.4137.

L. H. Chiang, E.L.R., Braatz, R.D., 2001. Fault detection and diagnosis in industrial systems. Meas. Sci. Technol. 12 (10), 1745. http://dx.doi.org/10.1088/0957-0233/12/10/706.

McKinney, W., 2010. Data structures for statistical computing in Python. In: Python in Science Conference. Austin, Texas, pp. 56–61. http://dx.doi.org/10.25080/Majora-92bf1922-00a.

Melesse, T.Y., Pasquale, V.D., Riemma, S., 2020. Digital twin models in industrial operations: A systematic literature review. Procedia Manuf. 42, 267–272. http://dx.doi.org/10.1016/j.promfg.2020.02.084.

Natanael, D., Sutanto, H., 2022. Machine learning application using cost-effective components for predictive maintenance in industry: A tube filling machine case study. J. Manuf. Mater. Process. 6 (5), 108. http://dx.doi.org/10.3390/jmmp6050108.

Pandit, R., Infield, D., 2018. Gaussian process operational curves for wind turbine condition monitoring. Energies 11 (7), 1631. http://dx.doi.org/10.3390/en11071631.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É., 2011. Scikit-learn: machine learning in Python. J. Mach. Learn. Res. 12 (85), 2825–2830.

Poor, P., Ženíšek, D., Basl, J., 2019. Historical overview of maintenance management strategies: Development from breakdown maintenance to predictive maintenance in accordance with four industrial revolutions. In: Proceedings of the International Conference on Industrial Engineering and Operations Management.

Qin, S.J., 2009. Data-driven fault detection and diagnosis for complex industrial processes. IFAC Proc. Vol. 42 (8), 1115–1125. http://dx.doi.org/10.3182/20090630-4-ES-2003.00184.

Qin, S.J., 2012. Survey on data-driven industrial process monitoring and diagnosis. Annu. Rev. Control 36 (2), 220–234. http://dx.doi.org/10.1016/j.arcontrol.2012.09.004.

Quiñones-Grueiro, M., Prieto-Moreno, A., Verde, C., Llanes-Santiago, O., 2019. Data-driven monitoring of multimode continuous processes: A review. Chemometr. Intell. Lab. Syst. 189, 56–71. http://dx.doi.org/10.1016/j.chemolab.2019.03.012.

Radhakrishnan, V.R., Ramasamy, M., Zabiri, H., Do Thanh, V., Tahir, N.M., Mukhtar, H., Hamdi, M.R., Ramli, N., 2007. Heat exchanger fouling model and preventive maintenance scheduling tool. Appl. Therm. Eng. 27 (17), 2791–2802. http://dx.doi.org/10.1016/j.applthermaleng.2007.02.009.

Rasmussen, C.E., Williams, C.K.I., 2008. Gaussian Processes for Machine Learning, 3. print ed. In: Adaptive Computation and Machine Learning, MIT Press, Cambridge, Mass.

Ren, Y., 2021. Optimizing predictive maintenance with machine learning for reliability improvement. Asce-Asme J. Risk Uncertainty Eng. Syst. B 7 (3), 030801. http://dx.doi.org/10.1115/1.4049525.

Seabold, S., Perktold, J., 2010. Statsmodels: Econometric and statistical modeling with Python. In: Python in Science Conference. Austin, Texas, pp. 92–96. http://dx.doi.org/10.25080/Majora-92bf1922-011.

Silvestrin, L.P., Hoogendoorn, M., Koole, G., 2019. A comparative study of state-of-the-art machine learning algorithms for predictive maintenance. In: 2019 IEEE Symposium Series on Computational Intelligence. IEEE SSCI 2019, IEEE, New York, pp. 760–767.

Susto, G.A., Beghi, A., De Luca, C., 2012. A predictive maintenance system for epitaxy processes based on filtering and prediction techniques. IEEE Trans. Semicond. Manuf. 25 (4), 638–649. http://dx.doi.org/10.1109/TSM.2012.2209131.

Susto, G.A., Schirru, A., Pampuri, S., McLoone, S., Beghi, A., 2015. Machine learning for predictive maintenance: A multiple classifier approach. IEEE Trans. Ind. Inform. 11 (3), 812–820. http://dx.doi.org/10.1109/TII.2014.2349359.

Thomas, D.S., Weiss, B.A., 2020. Economics of Manufacturing Machinery Maintenance:: A Survey and Analysis of U.S. Costs and Benefits. Technical Report NIST AMS 100–34, National Institute of Standards and Technology, Gaithersburg, MD, http://dx.doi.org/10.6028/NIST.AMS.100-34.

Traini, E., Bruno, G., D'Antonio, G., Lombardi, F., 2019. Machine learning framework for predictive maintenance in milling. IFAC-PapersOnLine 52 (13), 177–182. http://dx.doi.org/10.1016/j.ifacol.2019.11.172.

Valentini, G., Masulli, F., 2002. Ensembles of learning machines. In: Marinaro, M., Tagliaferri, R. (Eds.), Neural Nets. In: Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, pp. 3–20. http://dx.doi.org/10.1007/3-540-45808-5_1.

Van Rossum, G., Drake, F.L., 2009. Python 3 Reference Manual. CreateSpace, Scotts Valley, CA.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S., 2003a. A review of process fault detection and diagnosis part II: Qualitative models and search strategies. Comput. Chem. Eng. 27 (3), 313–326. http://dx.doi.org/10.1016/S0098-1354(02)00161-8.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S., Yin, K., 2003b. A review of process fault detection and diagnosis part III: Process history based methods. Comput. Chem. Eng. 27 (3), 327–346. http://dx.doi.org/10.1016/S0098-1354(02)00162-X.

Venkatasubramanian, V., Rengaswamy, R., Yin, K., Kavuri, S.N., 2003c. A review of process fault detection and diagnosis: Part I: Quantitative model-based methods. Comput. Chem. Eng. 27 (3), 293–311. http://dx.doi.org/10.1016/S0098-1354(02)00160-6.

Willmott, C.J., Matsuura, K., 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. Clim. Res. 30 (1), 79–82. http://dx.doi.org/10.3354/cr030079.

Willmott, C.J., Matsuura, K., Robeson, S.M., 2009. Ambiguities inherent in sums-of-squares-based error statistics. Atmos. Environ. 43 (3), 749–752. http://dx.doi.org/10.1016/j.atmosenv.2008.10.005.

Wu, O., Bouaswaig, A.E.F., Schneider, S.M., Leira, F.M., Imsland, L., Roth, M., 2018. Data-driven degradation model for batch processes: A case study on heat exchanger fouling. In: Friedl, A., Klemeš, J.J., Radl, S., Varbanov, P.S., Wallek, T. (Eds.), Computer Aided Chemical Engineering. In: 28 European Symposium on Computer Aided Process Engineering, vol. 43, Elsevier, pp. 139–144. http://dx.doi.org/10.1016/B978-0-444-64235-6.50026-7.

Yan, J., Meng, Y., Lu, L., Li, L., 2017. Industrial big data in an industry 4.0 environment: Challenges, schemes, and applications for predictive maintenance. IEEE Access 5, 23484–23491. http://dx.doi.org/10.1109/ACCESS.2017.2765544.

Yan, W., Qiu, H., Xue, Y., 2009. Gaussian process for long-term time-series forecasting. In: 2009 International Joint Conference on Neural Networks. pp. 3420–3427. http://dx.doi.org/10.1109/IJCNN.2009.5178729.

Yeardley, A.S., Ejeh, J.O., Allen, L., Brown, S.F., Cordiner, J., 2022. Integrating machine learning techniques into optimal maintenance scheduling. Comput. Chem. Eng. 166, 107958. http://dx.doi.org/10.1016/j.compchemeng.2022.107958.

Yin, S., Ding, S.X., Haghani, A., Hao, H., Zhang, P., 2012. A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark Tennessee Eastman process. J. Process Control 22 (9), 1567–1581. http://dx.doi.org/10.1016/j.jprocont.2012.06.009.

Yin, S., Ding, S.X., Xie, X., Luo, H., 2014. A review on basic data-driven approaches for industrial process monitoring. IEEE Trans. Ind. Electron. 61 (11), 6418–6428. http://dx.doi.org/10.1109/TIE.2014.2301773.

Zeng, J., Liang, Z., 2022. A deep Gaussian process approach for predictive maintenance. IEEE Trans. Reliab. 1–18. http://dx.doi.org/10.1109/TR.2022.3199924.

Zeng, J., Liang, Z., Guo, C., Song, M., Xue, Z., 2021. A Gaussian process approach for predictive maintenance. In: 2021 IEEE 21st International Conference on Software Quality, Reliability and Security Companion. QRS-C, pp. 745–750. http://dx.doi.org/10.1109/QRS-C55045.2021.00113.

Zhang, J., Liu, C., Gao, R.X., 2022. Physics-guided Gaussian process for HVAC system performance prognosis. Mech. Syst. Signal Process. 179, 109336. http://dx.doi.org/10.1016/j.ymssp.2022.109336.

Zhang, W., Yang, D., Wang, H., 2019. Data-driven methods for predictive maintenance of industrial equipment: A survey. IEEE Syst. J. 13 (3), 2213–2227. http://dx.doi.org/10.1109/JSYST.2019.2905565.

Zhu, J., Ge, Z., Song, Z., Gao, F., 2018. Review and big data perspectives on robust data mining approaches for industrial process modeling with outliers and missing data. Annu. Rev. Control 46, 107–133. http://dx.doi.org/10.1016/j.arcontrol.2018.09.003.

Zonta, T., da Costa, C.A., da Rosa Righi, R., de Lima, M.J., da Trindade, E.S., Li, G.P., 2020. Predictive maintenance in the Industry 4.0: A systematic literature review. Comput. Ind. Eng. 150, 106889. http://dx.doi.org/10.1016/j.cie.2020.106889.