Neurocomputing 403 (2020) 383–399

Contents lists available at ScienceDirect

Neurocomputing

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

Online anomaly detection with sparse Gaussian processes

Minghao Gu[1], Jingjing Fei[1], Shiliang Sun*

School of Computer Science and Technology, East China Normal University, 3663 North Zhongshan Road, Shanghai 200241, PR China

ARTICLE INFO

Article history:
Received 14 May 2019
Revised 25 November 2019
Accepted 13 April 2020
Available online 13 May 2020

Communicated by Prof. Zidong Wang

Keywords:
Online anomaly detection
Gaussian processes
Sparse Gaussian processes
Q-function
Concept drift

ABSTRACT

Online anomaly detection of time-series data is an important and challenging task in machine learning. Gaussian processes (GPs) are powerful and flexible models for modeling time-series data. However, the high time complexity of GPs limits their applications in online anomaly detection. Attributed to some internal or external changes, concept drift usually occurs in time-series data, where the characteristics of data and meanings of abnormal behaviors alter over time. Online anomaly detection methods should have the ability to adapt to concept drift. Motivated by the above facts, this paper proposes the method of sparse Gaussian processes with Q-function (SGP-Q). The SGP-Q employs sparse Gaussian processes (SGPs) whose time complexity is lower than that of GPs, thus significantly speeding up online anomaly detection. By using Q-function properly, the SGP-Q can adapt to concept drift well. Moreover, the SGP-Q makes use of few abnormal data in the training data by its strategy of updating training data, resulting in more accurate sparse Gaussian process regression models and better anomaly detection results. We evaluate the SGP-Q on various artificial and real-world datasets. Experimental results validate the effectiveness of the SGP-Q.

1. Introduction

The arrival of the Internet of Things (IoT) [1] has inspired companies to install more sensors on their machines. These sensors can produce vast amounts of data which change over time and are called time-series data. The time-series data in our life are increasing rapidly, and it is of considerable significance and challenge to effectively process and monitor the time-series data.

A typical application scenario of time-series data is to monitor time-series data and send an alarm message to the operation engineers when an abnormal behavior occurs in the time-series data. This kind of application is called anomaly detection of time-series data. An abnormal behavior means that the current behavior is largely different from the normal behaviors in the past, and the current behavior is very rare [2,3]. In the real industrial environment, the time-series data are generated at all times, and only the previous and current data can be known, while the data after the current time are unseen. Therefore, online learning is essential for the anomaly detection of time-series data. Online anomaly detection determines whether the current behavior is abnormal or not according to the information of the current and previous data.

Online anomaly detection is significant because abnormal data can often convey essential and critical information in an extensive range of applications [4–14]. For example, when abnormal traffic occurs in a computer network, a hacker may be using the attacked computer to send sensitive data to the target computer [15]. When the MRI image is abnormal, it is likely to be due to the existence of some malignant tumor [16]. If abnormal data come from the sensors of the aircraft, it indicates that some parts of the aircraft may have malfunctions that need to be repaired in time [17].

Gaussian processes (GPs) are powerful and flexible tools for modeling time-series data [18]. However, there are few methods of online anomaly detection based on GPs. There are two main reasons. Firstly, the time complexity of GPs is the cube of the number of training data. Training the GP model is time-consuming. Secondly, GPs are Bayesian nonparametric probabilistic models that are unfamiliar to industrial engineers. In recent years, many research work on sparse Gaussian processes (SGPs) have been proposed to reduce the high time complexity of GPs [19–24]. These research work can be divided into four categories. The first kind of method uses the Nyström method to approximate the covariance matrix [19,25]. The second kind of method employs a subset of data and selects informative points from the subset of data [20,26]. The third kind of method uses pseudo-inputs for the sparse approximation of GPs. The fourth kind of method employs variational inference and introduces inducing variables [23,24]. SGPs greatly reduce the time complexity of GPs, which allows SGPs to be widely

* Corresponding author.
E-mail address: slsun@cs.ecnu.edu.cn (S. Sun).
[1] These authors contributed equally to this work.

https://doi.org/10.1016/j.neucom.2020.04.077
0925-2312/

used on various types of data [27,28]. Inspired by the above facts, this paper employs SGPs with variational inducing variables for online anomaly detection.

The existing online anomaly detection methods based on GPs, including the Gaussian process regression with anomaly detection strategy (GPR-AD), Gaussian process regression with anomaly detection and mitigation strategy (GPR-ADAM) and Gaussian process regression with the improved anomaly detection and mitigation strategy (GPR-IADAM) [29], cannot address concept drift in data properly. Concept drift means that data changes over time and the characteristics of new data are different from those of old data [3,30,31]. For example, sensor data collected by machines often changes on account of restarting machines or updating configurations. When the characteristics of data change, the definition of abnormal behaviors of data will also change. If anomaly detection methods cannot update their definition of abnormal behaviors in time, they cannot accurately detect anomalies in new data. Therefore, it is essential that online anomaly detection methods have the ability to adapt to the concept drift [32–38].

This paper proposes the method of sparse Gaussian processes with Q-function (SGP-Q). The SGP-Q uses the Q-function [3,39] to adapt to the concept drift in the data well. Specifically, the SGP-Q employs Q-function to measure the abnormal degree of the current data point relative to that of previous data. When the concept drift occurs in the data, the new data that initially change will be considered as abnormal data. However, when the 'abnormal' behaviors persist for a while, the SGP-Q considers that the abnormal degree of the current data point relative to that of previous data is low. Then the SGP-Q adds the current data point and its time into training data to update the sparse Gaussian process regression (SGPR) model. The SGPR model can relearn the characteristics of new data to redefine the meanings of abnormal behaviors. Therefore, the SGP-Q can adapt to concept drift well. In the experiment, the proposed SGP-Q is compared with the online anomaly detection methods based on GPs, and experimental results validate the effectiveness of the proposed SGP-Q.

The contributions of this paper are listed as follows. Firstly, the proposed SGP-Q employs Q-function to measure the abnormal degree of the current data point relative to that of previous data, which can adapt to concept drift well. Secondly, SGPs with variational inducing variables are used to model time-series data in the SGP-Q, whose time complexity is much lower than that of GPs, thus speeding up online anomaly detection. Thirdly, the SGP-Q updates training data by the strategy based on likelihood and Q-function, which can make use of few abnormal data in the training data, thus making the trained SGPR model more accurate and anomaly detection results better.

The remainder of this paper is organized as follows. Section 2 describes the related work, including the introduction of GPs and several online anomaly detection methods based on GPs. Section 3 briefly reviews SGPs and introduces the proposed method SGP-Q in detail. Section 4 illustrates experiments, including the introduction of datasets, experimental setting, experimental results, and summary. Section 5 concludes the work of this paper.

2. Related work

In this section, we will introduce GPs and several anomaly detection methods of time-series data based on GPs.

2.1. Gaussian processes

GPs are powerful and flexible Bayesian nonparametric probabilistic models, which are mainly applied to regression and classification tasks. The GP regression model predicts continuous values, and the GP classification model predicts discrete values.

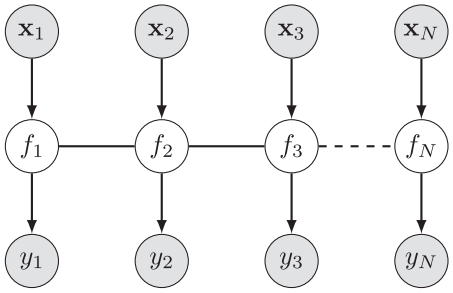


Fig. 1. The graphical model of a GP with N training points.

Assume that the size of training dataset D is N, that is, $D = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, where $\mathbf{x}_i \in \mathbb{R}^Q$ and $y_i \in \mathbb{R}^+$ represent the input and output of the ith data point, respectively. Denote all inputs as X, and denote all outputs as $\mathbf{y}$.

GPs can be seen as Gaussian distributions on real-valued functions. The Gaussian distribution is uniquely determined by its mean and covariance matrix, and the GP is uniquely specified by its mean and covariance function similarly [18]. A noiseless GP $f(\mathbf{x})$ can be expressed as follows,

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')), \tag{1}$$

where $m(\mathbf{x})$ is the mean function of the GP and $k(\mathbf{x}, \mathbf{x}')$ is the covariance function of the GP, which is also known as kernel function. The selection of kernel function plays an important roles in the GP model. Common kernel functions include the radial basis function (RBF) kernel function, periodic kernel function and linear kernel function. The periodic kernel function is capable of modeling periodicity in data. These kernel functions are computed as follows,

$$\begin{aligned} k_{(rbf)}(\mathbf{x}, \mathbf{x}') &= \sigma_{rbf}^2 \exp\left[-\frac{|\mathbf{x}-\mathbf{x}'|^2}{2l^2}\right], \\ k_{(periodic)}(\mathbf{x}, \mathbf{x}') &= \sigma_{periodic}^2 \exp\left[-\frac{1}{2}\sum_{i=1}^{Q}\left(\frac{\sin(\frac{\pi}{T}(x_i - x_i'))}{l_i}\right)^2\right], \\ k_{(linear)}(\mathbf{x}, \mathbf{x}') &= \sum_{i=1}^{Q} \sigma_{linear(i)}^2 x_i x_i', \end{aligned} \tag{2}$$

where σ_{rbf}^2, $\sigma_{periodic}^2$ and σ_{linear}^2 are the variances, l is the length-scale and T is the periodic parameter.

The mean and covariance functions are calculated as follows,

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})], \\ k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))]. \end{aligned} \tag{3}$$

In the noiseless situation, the output y is the GP $f(\mathbf{x})$. However, in general, the observed output y is not the GP $f(\mathbf{x})$, but with some noise, e.g., $y = f(\mathbf{x}) + \epsilon$, where $\epsilon \sim N(0, \sigma^2)$ is the additive noise. Fig. 1 shows the graphical model of a GP.

Given the model assumption of the GP, the Gaussian likelihood is $p(\mathbf{y}|\mathbf{f}) = \mathcal{N}(\mathbf{f}, \sigma^2 I)$. After integrating the latent variable $\mathbf{f}$, we can get the marginal likelihood distribution $p(\mathbf{y}|X) = \mathbf{N}(0, K_{NN} + \sigma^2 I)$ where $K_{NN} = K(X, X)$ is an $N \times N$ covariance matrix. The GP model is learned by maximizing the marginal likelihood. The posterior of the latent variable $\mathbf{f}$ is as follows,

$$p(\mathbf{f}|\mathbf{y}) = \mathcal{N}(\boldsymbol{\mu}, \Sigma), \tag{4}$$

where the mean $\boldsymbol{\mu} = K_{NN}(K_{NN} + \sigma^2 I)^{-1}\mathbf{y}$ and the covariance matrix $\Sigma = K_{NN} - K_{NN}(K_{NN} + \sigma^2 I)^{-1}K_{NN}$.

Given a new input $\mathbf{x}^*$, the prediction distribution is still a Gaussian distribution,

$$p(y^*|X, \mathbf{y}, \mathbf{x}^*) = N(\mu^*, \Sigma_y^*), \tag{5}$$

where $\mu^* = k(\mathbf{x}^*, X)[K_{NN} + \sigma^2 I]^{-1}\mathbf{y}$ and $\Sigma_y^* = k(\mathbf{x}^*, \mathbf{x}^*) - k(\mathbf{x}^*, X)(K_{NN} + \sigma^2 I)^{-1}k(X, \mathbf{x}^*) + \sigma^2$. The time complexity of the GP is $O(N^3)$, where N is the number of training data.

2.2. Gaussian processes for anomaly detection in time-series data

In this section, we mainly introduce several existing anomaly detection methods of time-series data based on GPs, including the Gaussian process regression (GPR) with the anomaly detection strategy (GPR-AD), the GPR with the anomaly detection and mitigation strategy (GPR-ADAM) and the GPR with the improved anomaly detection and mitigation strategy (GPR-IADAM) [29]. These methods belong to anomaly detection methods based on prediction, which detect anomalies by judging whether the data fall into the prediction interval.

Firstly, we focus on the state-of-the-art anomaly detection method of time-series data, that is, the GPR-IADAM. For the task of anomaly detection in time-series data, the input is time. Assuming the current time is m, the training data is composed of q data points that are closest to the current moment, i.e., $D_T = \{t_i, y_i\}_{i=m-q+1}^{m}$. The GPR model can be obtained by maximizing the marginal likelihood. The time t_{m+1} at the next moment is taken as the test input, and the prediction mean μ_{m+1} and variance σ_{m+1}^2 can be obtained according to the Eq. (5). The 95 percent confidence interval for the Gaussian distribution is $[\mu_{m+1} - 1.96\sigma_{m+1}, \mu_{m+1} + 1.96\sigma_{m+1}]$.

If the test data point y_{m+1} is within the 95 percent confidence interval, it is considered as a normal data point, and the data point y_{m+1} and its time t_{m+1} are added into the sliding window D_T. If the test data point y_{m+1} is not in the 95 percent confidence interval, it is marked as an abnormal data point. The value of $\beta(y_{m+1})$ needs to be calculated according to the Eq. (6).

$$\beta(y_{m+1}) = P\left(z < 1.96 - \frac{|\mu_{m+1} - y_{m+1}|}{\sigma_{m+1}}\right), \tag{6}$$

where z obeys the standard Gaussian distribution. The value of $\beta(y_{m+1})$ is used to measure the deviation between the data point y_{m+1} and prediction mean μ_{m+1}. The smaller the value of β is, the greater the deviation between the data point y_{m+1} and prediction mean μ_{m+1} is. The larger the value of β is, the smaller the deviation between the data point y_{m+1} and prediction mean μ_{m+1} is. Comparing the value of β and $\beta_{\max}$, if β is less than or equal to $\beta_{\max}$, the prediction mean μ_{m+1} and its time t_{m+1} will be added to the sliding window D_T. If β is greater than $\beta_{\max}$, the data point y_{m+1} and its time t_{m+1} will be added to the sliding window D_T. Here $\beta_{\max}$ is an artificially specified threshold, and the GPR-IADAM sets the value of $\beta_{\max}$ to 0.05.

After adding the new data point and its time in the sliding window D_T, the earliest data point and its time are removed from the D_T. The data in the new sliding window D_T are then employed to update the GPR model. Algorithm 1 shows the GPR-IADAM method.

Secondly, we will introduce the GPR-AD. The difference between the GPR-AD and GPR-IADAM is the strategy of updating data in the sliding window D_T. The GPR-AD updates the sliding window D_T using the anomaly detection (AD) strategy. The AD strategy adds the data point y_{m+1} and its time t_{m+1} to the sliding window D_T regardless of whether the data point is abnormal or not, and the earliest data point and its time are removed from the D_T. Algorithm 2 shows the GPR-AD method.

Last but not least, we briefly describe the GPR-ADAM. The only difference between the GPR-IADAM and GPR-ADAM is also the strategy of updating data in the sliding window D_T. GPR-ADAM updates the sliding window D_T using the anomaly detection and mitigation (ADAM) strategy. In the ADAM strategy, the data point y_{m+1} and its time t_{m+1} are added to the sliding window D_T when the data point is normal. The prediction mean μ_{m+1} and its time t_{m+1} are added to the sliding window D_T when the data point is abnormal. The earliest data point and its time are removed from the D_T. Algorithm 3 shows the GPR-ADAM method.

Algorithm 1 GPR-IADAM.

Input: current time m, $D_T = \{t_i, y_i\}_{i=m-q+1}^{m}$, the size of sliding window q and threshold $\beta_{\max}$

Initialize: select an appropriate covariance function (the mean function is generally set to 0), and initialize parameters of the covariance function

repeat
  train the GPR model using data in the sliding window D_T
  predict mean μ_{m+1} and variance σ_{m+1}^2 at time t_{m+1} by Equation (5)
  compute 95% confidence interval $[\mu_{m+1} - 1.96\sigma_{m+1}, \mu_{m+1} + 1.96\sigma_{m+1}]$
  if data point y_{m+1} in 95% confidence interval **then**
    y_{m+1} is a normal data point
    add $\{t_{m+1}, y_{m+1}\}$ into D_T
  else
    y_{m+1} is an abnormal data point
    compute the value of $\beta(y_{m+1})$ by Equation (6)
    if $\beta(y_{m+1}) \le \beta_{\max}$ **then**
      add $\{t_{m+1}, \mu_{m+1}\}$ into D_T
    else
      add $\{t_{m+1}, y_{m+1}\}$ into D_T
    end if
  end if
  remove the earliest data point and its time from D_T
  $m = m + 1$
until all test data points have been detected

Algorithm 2 GPR-AD.

Input: current time m, $D_T = \{t_i, y_i\}_{i=m-q+1}^{m}$ and the size of sliding window q

Initialize: select an appropriate covariance function (the mean function is generally set to 0), and initialize parameters of the covariance functions

repeat
  train the GPR model using data in the sliding window D_T
  predict mean μ_{m+1} and variance σ_{m+1} at time t_{m+1} by Equation (5)
  compute 95% confidence interval $[\mu_{m+1} - 1.96\sigma_{m+1}, \mu_{m+1} + 1.96\sigma_{m+1}]$
  if data point y_{m+1} in 95% confidence interval **then**
    y_{m+1} is a normal data point
  else
    y_{m+1} is an abnormal data point
  end if
  add $\{t_{m+1}, y_{m+1}\}$ into D_T
  remove the earliest data point and its time from D_T
  $m = m + 1$
until all test data points have been detected

Algorithm 3 GPR-ADAM.

Input: current time m, $D_T = \{t_i, y_i\}_{i=m-q+1}^{m}$ and the size of sliding window q

Initialize: select an appropriate covariance function (the mean function is generally set to 0), and initialize parameters of the covariance function

repeat
  train the GPR model using data in sliding window D_T
  predict mean μ_{m+1} and variance σ_{m+1} at time t_{m+1} by Equation (5)
  compute 95% confidence interval $[\mu_{m+1} - 1.96\sigma_{m+1}, \mu_{m+1} + 1.96\sigma_{m+1}]$
  if data point y_{m+1} in 95% confidence interval **then**
    y_{m+1} is a normal data point
    add $\{t_{m+1}, y_{m+1}\}$ into D_T
  else
    y_{m+1} is an abnormal data point
    add $\{t_{m+1}, \mu_{m+1}\}$ into D_T
  end if
  remove the earliest data point and its time from D_T
  $m = m + 1$.
until all test data points have been detected

3. Sparse Gaussian processes with Q-function

In this section, we first review the sparse Gaussian processes (SGPs) and then introduce the proposed method of sparse Gaussian processes with Q-function (SGP-Q).

3.1. Sparse Gaussian processes

The GP is a powerful and flexible model, but its high time complexity $O(N^3)$ limits its application scenarios. In order to reduce the high time complexity of the GP, different sparse approximation methods are proposed [19–24]. Here we mainly review the sparse approximation method using variational inducing variables [21,23].

In the approximation methods based on inducing variables, the active set is not a subset of data selected from training data but is the inducing input Z obtained through optimization. The latent variable corresponding to the inducing input is $\mathbf{u} = f(Z)$. SGPs with inducing variables usually have three forms of approximation, that is, deterministic training conditional (DTC) approximation, fully independent training conditional (FITC) approximation and partially independent training conditional (PITC) approximation [21]. The key difference between the three approximation methods is that they assume different conditional distributions $p(\mathbf{f}|\mathbf{u}, Z, X)$.

The DTC approximation refers to the fact that the value of the latent variable $\mathbf{f}$ is deterministic when the inducing variable $\mathbf{u}$ is known,

$$p(\mathbf{f}|\mathbf{u}, Z, X) = \mathcal{N}(\mathbf{f}|K_{NM}K_{MM}^{-1}\mathbf{u}, \mathbf{0}), \tag{7}$$

where M is the number of data points in the inducing input, $K_{NM} = K(X, Z)$, $K_{MM} = K(Z, Z)$ and $K_{MN} = K(Z, X)$.

The FITC approximation means that the latent variable $\mathbf{f}$ is fully independent when the inducing variable $\mathbf{u}$ is known,

$$p(\mathbf{f}|\mathbf{u}, Z, X) = \mathcal{N}(\mathbf{f}|K_{NM}K_{MM}^{-1}\mathbf{u}, diag[K_{NN} - K_{NN}K_{MM}^{-1}K_{MN}]). \tag{8}$$

The PITC approximation signifies that the latent variable $\mathbf{f}$ is partially independent when the inducing variable $\mathbf{u}$ is known,

$$p(\mathbf{f}|\mathbf{u}, Z, X) = \mathcal{N}(\mathbf{f}|K_{NM}K_{MM}^{-1}\mathbf{u}, blockdiag[K_{NN} - K_{NN}K_{MM}^{-1}K_{MN}]). \tag{9}$$

Under the above three assumptions, the marginal likelihood of the SGP is a function of the inducing input and hyperparameters of the kernel functions. The inducing input and hyperparameters can be obtained by maximizing the marginal likelihood, which may lead to over-fitting.

SGPs with variational inducing variables assume the augmented variational posterior $q(\mathbf{f}, \mathbf{u})$ to approximate the augmented true posterior $p(\mathbf{f}, \mathbf{u}|\mathbf{y})$, and minimize the Kullback–Leibler (KL) divergence between the variational posterior and true posterior $KL(q(\mathbf{f}, \mathbf{u})||p(\mathbf{f}, \mathbf{u}|\mathbf{y}))$ [23]. Minimizing the KL divergence $KL(q(\mathbf{f}, \mathbf{u})||p(\mathbf{f}, \mathbf{u}|\mathbf{y}))$ is equivalent to maximizing the variational lower bound of the marginal likelihood. The variational lower bound of the SGP is as follows,

$$\mathcal{L} = \log\left[N(\mathbf{y}|\mathbf{0}, \sigma^2 I + K_{NM}K_{MM}^{-1}K_{MN})\right] - \frac{1}{2\sigma^2}Tr\left[K_{NN} - K_{NM}K_{MM}^{-1}K_{MN}\right]. \tag{10}$$

Here the inducing input Z is seen as the variational parameter obtained by optimizing the variational lower bound and has no impact on the marginal likelihood $p(\mathbf{y})$ and posterior $p(\mathbf{f}|\mathbf{y})$, which is beneficial to avoiding over-fitting to a certain extent.

Given a test input $\mathbf{x}_*$, the prediction distribution of the SGP with variational inducing variables is still a Gaussian distribution as follows [23],

$$p(y^*|X, Z, \mathbf{y}, \mathbf{x}^*) = N(\mu^*, \Sigma_y^*), \tag{11}$$

where $\mu^* = k(\mathbf{x}^*, Z)K_{MM}^{-1}\tilde{\boldsymbol{\mu}}$, $\Sigma_y^* = k(\mathbf{x}^*, \mathbf{x}^*) - k(\mathbf{x}^*, Z)K_{MM}^{-1}k(Z, \mathbf{x}_*) + k(\mathbf{x}^*, Z)Bk(Z, \mathbf{x}_*) + \sigma^2$, $\tilde{\boldsymbol{\mu}} = \sigma^{-2}K_{MM}\Sigma K_{MN}\mathbf{y}$, $A = K_{MM}\Sigma K_{MM}$, $\Sigma = (K_{MM} + \sigma^{-2}K_{MN}K_{NM})^{-1}$, and $B = K_{MM}^{-1}AK_{MM}^{-1}$. The time complexity of the SGP with variational inducing variables is $O(NM^2)$.

3.2. Online anomaly detection with sparse Gaussian processes

The existing online anomaly detection methods have some limitations. The GPR-AD adds a data point to the training data no matter whether the data point is abnormal or not. When lots of abnormal data are added into the training data, the prediction may lose its accuracy, thus leading to wrong anomaly detection results. The GPR-ADAM adds a data point into the training data when the data point is normal and adds the prediction mean of the GPR model to the training data when the data point is abnormal. The GPR-ADAM alleviates the negative impact of abnormal data on the training model. The GPR-IADAM is a compromise between the GPR-AD and GPR-ADAM. Specifically, the GPR-IADAM uses β to measure the deviation between the data point and the prediction mean. If the prediction mean deviates significantly from the data point, the prediction mean is added to the training data. If the prediction mean deviates slightly from the data point, the data point will be added to the training data. The GPR-ADAM and GPR-IADAM decide to add the data point or prediction mean into the training data according to the abnormal degree of the current data point, which cannot solve the problem of concept drift in the data. Concept drift [3,31,40] means that the characteristics of data and the mapping from input to output change over time. For example, when the computer's software is updated or its configuration is changed, data such as CPU utilization and the speed of reading or writing data in the disk will change. In this case, online anomaly detection methods should be able to adapt to the new data and redefine the meanings of abnormal behaviors.

To overcome the limitations of existing anomaly detection methods of time-series data based on GPs, this paper proposes the SGP-Q. On the one hand, the SGP-Q uses SGPs to model the relationship between time and observed data. The time complexity of SGPs is much lower than that of GPs, while their performance is similar. Therefore, SGPs are more suitable for the task of online anomaly detection. On the other hand, the SGP-Q method employs Q-function [39] to solve the problem of concept drift in time-series data.

Specifically, the SGP-Q uses the technique of sliding window, and the size of the sliding window is set to q. Increasing the value of q will slightly improve the experimental performance, but largely increase the running time. Suppose current time is m, then data in the sliding window can be represented as $D_T = \{t_i, y_i\}_{i=m-q+1}^{m}$. Given a test data point $\{t_{m+1}, y_{m+1}\}$, the time t_{m+1} is the input of the SGPR, and the outputs of the SGPR contain the prediction mean μ_{m+1} and variance σ_{m+1}^2. Then the SGP-Q computes the likelihood of the data point y_{m+1} by Eq. (12).

$$p(y_{m+1}) = \frac{1}{\sqrt{2\pi}\sigma_{m+1}} \exp\left[-\frac{(y_{m+1}-\mu_{m+1})^2}{2\sigma_{m+1}^2}\right]. \tag{12}$$

If the likelihood of the test data point $p(y_{m+1})$ is greater than or equal to the threshold ϵ_p, the data point y_{m+1} is normal, and the data point y_{m+1} and its time t_{m+1} are added to the sliding window D_T. If the likelihood of the test data point $p(y_{m+1})$ is less than the threshold ϵ_p, the data point is abnormal. The SGP-Q lists a set of possible thresholds, and selects the threshold which performs best in the validation set according to the F_1 score as ϵ_p. For an abnormal data point, the SGP-Q computes the Q-function based on the absolute error $e(y_{m+1}) = |y_{m+1} - \mu_{m+1}|$ and the Q-function based on the likelihood $p(y_{m+1})$. The computation of Q-function requires three steps [3].

Firstly, the SGP-Q employs two windows to accommodate the latest W values of error and likelihood, respectively. The distributions of the error and likelihood are modeled as Gaussian distributions where the mean and variance are updated according to the latest values of error and likelihood as follows,

$$\begin{aligned}
\hat{\mu}(e(y_{m+1})) &= \frac{\sum_{i=0}^{i=W-1} e(y_{m+1-i})}{W},\\
\sigma^2(e(y_{m+1})) &= \frac{\sum_{i=0}^{i=W-1}[e(y_{m+1-i}) - \hat{\mu}(e(y_{m+1}))]^2}{W-1},\\
\hat{\mu}(p(y_{m+1})) &= \frac{\sum_{i=0}^{i=W-1} p(y_{m+1-i})}{W},\\
\sigma^2(p(y_{m+1})) &= \frac{\sum_{i=0}^{i=W-1}[p(y_{m+1-i}) - \hat{\mu}(p(y_{m+1}))]^2}{W-1}.
\end{aligned} \tag{13}$$

Secondly, the SGP-Q computes the mean of the error and likelihood over the recent short period as follows,

$$\begin{aligned}
\tilde{\mu}(e(y_{m+1})) &= \frac{\sum_{i=0}^{i=W'-1} e(y_{m+1-i})}{W'},\\
\tilde{\mu}(p(y_{m+1})) &= \frac{\sum_{i=0}^{i=W'-1} p(y_{m+1-i})}{W'},
\end{aligned} \tag{14}$$

where W' is the size of the window for the recent short period, and $W' \ll W$. If the value of the parameter W is large, the concept drift occurs after the abnormality lasts for a long time. Conversely, if the value of the parameter W is small, the concept drift occurs when the abnormal duration is short. The parameter W' is used to smooth the value of the current window and reduce the influence of noise data.

Thirdly, the Gaussian Q-function (denoted as Q-function) is of considerable significance in various fields, such as statistics and communication theory [41,42]. The Q-function has no analytic solution, and several research work are proposed for approximate calculation of the Q-function [39,43,44]. An approximation of the Q-function is the exponential approximation which is easy to calculate. The exponential approximation of the Q-function is $Q(x) \approx \frac{1}{12}\exp(-\frac{x^2}{2}) + \frac{1}{4}\exp(-\frac{2x^2}{3})$ [43]. The SGP-Q modifies the original exponential approximation of the Q-function to make it more sensitive to slight input changes. That is to say, for the same input change, the output of the modified Q-function changes more, while the output of the original Q-function changes less. Our modified Q-function is $Q(x) \approx \frac{1}{6}\exp(-\frac{x^2}{4}) + \frac{1}{2}\exp(-\frac{x^2}{3})$. Our modified Q-function is an even function and the value of modified Q-function decreases as the absolute value of the input increases. Then the Q-function based on the absolute error (denoted as QE) and likelihood (denoted as QL) are defined as follows,

$$\begin{aligned}
QE_{m+1} &= Q\left(\frac{\tilde{\mu}(e(y_{m+1})) - \hat{\mu}(e(y_{m+1}))}{\sigma^2(e(y_{m+1}))}\right),\\
QL_{m+1} &= Q\left(\frac{\tilde{\mu}(p(y_{m+1})) - \hat{\mu}(p(y_{m+1}))}{\sigma^2(p(y_{m+1}))}\right).
\end{aligned} \tag{15}$$

Q-function can measure the abnormal degree of the current data point relative to that of previous data. The smaller the value of Q-function, the higher the abnormal degree of the current data point relative to that of previous data. The larger the value of Q-function, the lower the abnormal degree of the current data point relative to that of previous data.

For an abnormal data point, if at least one of the two conditions $QE_{m+1} < \epsilon_e$ and $QL_{m+1} < \epsilon_l$ is satisfied ($\epsilon_e = 3e-1$ and $\epsilon_l = 3e-1$), the SGP-Q adds the prediction mean μ_{m+1} and its time t_{m+1} to the sliding window D_T. If neither of the two conditions is met, the data point y_{m+1} and its time t_{m+1} are added to the sliding window D_T. After adding the latest data point to the sliding window D_T, the earliest data point and its time are removed from the D_T. The new data in the sliding window D_T are then employed to update the SGPR model. Algorithm 4 shows the SGP-Q method.

Algorithm 4 SGP-Q.

Input: current time m, $D_T = \{t_i, y_i\}_{i=m-q+1}^{m}$, the size of the window q, W, W', threshold ϵ_e, ϵ_l, and the size of inducing points M

Initialize: select an appropriate covariance function (the mean function is generally set to 0), and initialize the parameters of the covariance function

repeat
  train the SGPR model using data in sliding window D_T
  predict mean μ_{m+1} and variance σ_{m+1}^2 at time t_{m+1} by Equation (11)
  compute the likelihood of data point $p(y_{m+1})$ by Equation (12)
  select the threshold which performs best in the validation set according to F_1 score as ϵ_p
  if $p(y_{m+1}) > \epsilon_p$ **then**
    y_{m+1} is a normal data point
    add $\{t_{m+1}, y_{m+1}\}$ into D_T
  else
    y_{m+1} is an abnormal data point
    compute the value of QE_{m+1} and QL_{m+1} by Equation (15)
    if $QE_{m+1} \le \epsilon_e$ or $QL_{m+1} \le \epsilon_l$ **then**
      add $\{t_{m+1}, \mu_{m+1}\}$ into D_T
    else
      add $\{t_{m+1}, y_{m+1}\}$ into D_T
    end if
  end if
  remove the earliest data point and its time from D_T
  $m = m + 1$
until all test data points have been detected

Unlike the GPR-ADAM and GPR-IADAM which uses the information of the current data point to measure the abnormal degree of the current data point, the SGP-Q employs information of previous and current data to measure the abnormal degree of the current data point relative to that of previous data. The SGP-Q can address concept drift well. Specifically, when the concept drift occurs, data that start to change will be marked as abnormal data by the SGP-Q. However, when the 'abnormal' behaviors continue for a while, the

SGP-Q judges the abnormal degree of the current data point is low compared with that of previous data. Then the current data point y_{m+1} and its time t_{m+1} are added to the sliding window D_T to train the SGPR model. Therefore, the SGP-Q can learn new characteristics of data and redefine the meanings of abnormal behaviors.

4. Experiments

In this section, we conduct experiments to validate the rationality and effectiveness of the proposed SGP-Q method.

4.1. Data

We conducted experiments on nine datasets from the Numenta Anomaly Benchmark (NAB) [3]. Two of the nine datasets are the artificially generated datasets in NAB, denoted as 'art_daily_jumpsup' and 'art_daily_flatmiddle'. The number of data points on both artificially generated datasets is 4032. Five of the nine datasets are data of Amazon Web Services (AWS) server provided by the AmazonCloudwatch service. These five datasets of AWS server are denoted as 'ec2_cpu_utilization_24ae8d', 'ec2_cpu_utilization_825cc2', 'ec2_cpu_util-ization_ac20cd', 'ec2_cpu_utilization_5f5533', and 'grok_asg_anomaly', and the number of data points on the five datasets is 4032, 4032, 4648, 4621, and 4033 respectively. The remaining two datasets are real-time traffic data of the Twin Cities Metro area in Minnesota which are offered by the Minnesota Department of Transportation. These two traffic datasets are denoted as 'occupancy_t4013' and 'speed_t4013', and the number of data points on the two traffic datasets is 2500 and 2495, respectively.

4.2. Setting

For the SGP-Q, GPR-AD, GPR-ADAM and GPR-IADAM, the timestamps in the NAB exist as strings and need to be preprocessed for calculation. We quantize the timestamps to the number of minutes between the current time and today's 00:00:00 multiplied by 0.01. The covariance function is set to the sum of the RBF kernel function and the linear kernel function. The number of points in the inducing input Z is set to $M = 100$. The sizes of the sliding window D_T is set to $q = 1000$. The size of windows in the Q-function are set to $W = 500$ and $W' = 10$. For the GPR-IADAM, $\beta_{\max} = 0.05$. When the model is trained for the first time, we set the number of iterations to 1000. When the newest data point is added to D_T and the earliest data point is removed from D_T, we continue to optimize the model and set the number of iterations to 10. The data used for training the model for the first time is a piece of normal data at the beginning of each time-series data, where a small amount of abnormal data is allowed. The remaining data in each time-series data are used for testing. After the detection of each test data point, its real value or prediction mean will be added into the sliding window to update the model. A short time-series segment containing normal and abnormal data is taken from each time-series data, and then a tiny amount of noise is added to generate data in the validation set.

4.3. Experimental results

Since F_1 score can take into account the precision and recall, we use F_1 score as the measure of performance.

$$F_1 = \frac{2 \times precision \times recall}{precision + recall}. \tag{16}$$

All experiments were repeated five times, and the average results were taken as the final results. First, We compare the proposed SGP-Q with the GPR-AD, GPR-ADAM, and GPR-IADAM. Table 1 shows the $F_1(\%)$ score of the four methods on nine datasets, and the best results are in bold. As shown in Table 1, the proposed method SGP-Q performs best on all datasets, which demonstrates the effectiveness of the SGP-Q.

The SGP randomly initializes the kernel parameters of covariance functions and the induced variables Z, while the GP only requires to randomly initialize the kernel parameters of covariance functions. The SGP randomly initializes more parameters than that of the GP, and the uncertainty of the SGP is higher. Therefore, sometimes the variance of the SGP is larger. In addition, the SGP randomly initializes the parameters with different values each time, and the variances of the experimental results are affected by the initialized value of parameters to some extent. Therefore, the standard variances of the proposed method are different.

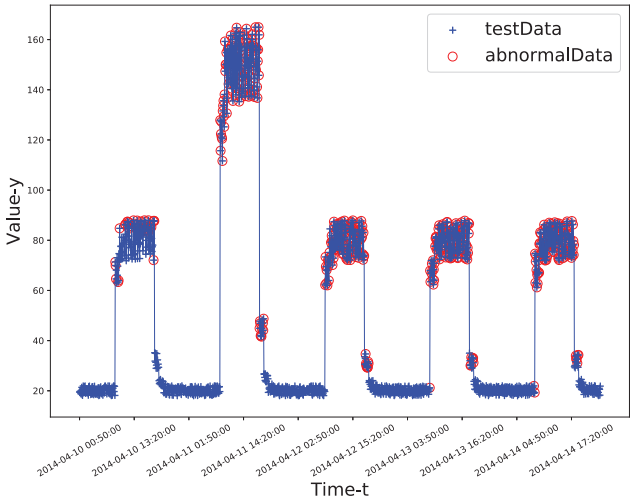
We further compare the proposed SGP-Q with two new kinds of prediction methods which are ContextOSE (CAD-OSE) [45] and Hierarchical Temporal Memory (HTM) [46]. We also compare the SGP-Q with other concept drift strategies which are SGP-DDM [47] and SGP-EDDM [48]. Table 2 shows the $F_1(\%)$ score of the five methods on nine datasets, and the best results are indicated in bold. As shown in Table 2, the proposed method SGP-Q performs well, which demonstrates the effectiveness of the SGP-Q.

In order to evaluate the performance of the Q-function, we compare the SGP-Q with SGP-AD, SGP-ADAM, and SGP-IADAM on nine datasets. Table 3 shows the $F_1(\%)$ score of the four methods on nine datasets, and the best results are indicated in bold. As shown in Table 3, the proposed method SGP-Q performs best on all datasets, which demonstrates the effectiveness of the Q-function.

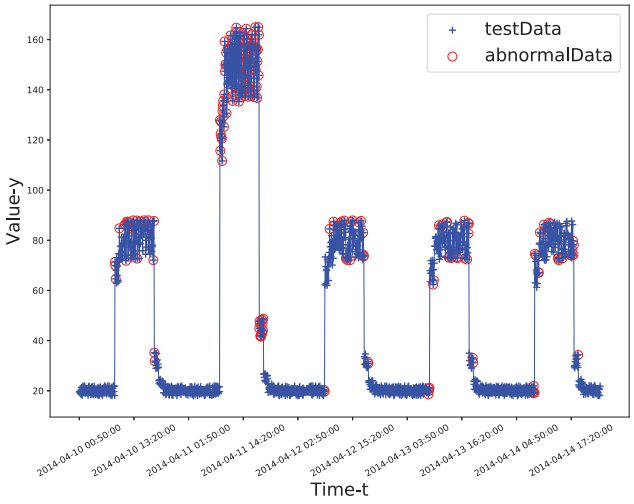
In addition, we will visually show the results of anomaly detection. The figures below show the results of anomaly detection by four methods and true labels of test data on eight datasets. The blue crosses represent test data, and the red circles represent detected anomalies. When the label is equal to 1, the test data point is abnormal. When the label is equal to 0, the test data point is normal.

Figs. 2 and 3 show the results of anomaly detection using four methods on the artificial datasets. As shown in Figs. 2(a) and 3(a), the GPR-AD has the worst performance. The reason is that the GPR-AD adds lots of abnormal data to the training data, which leads to inaccurate training models and marks subsequent normal data as abnormal data. In Figs. 2(b), (c) and 3(b), (c), the GPR-ADAM and GPR-IADAM have similar performance when abnormal data deviate from normal data obviously. Figs. 2(d) and 3(d) show that our method SGP-Q has the best performance. Almost all abnormal data are detected, and only a few normal data are wrongly marked as abnormal data by the SGP-Q.

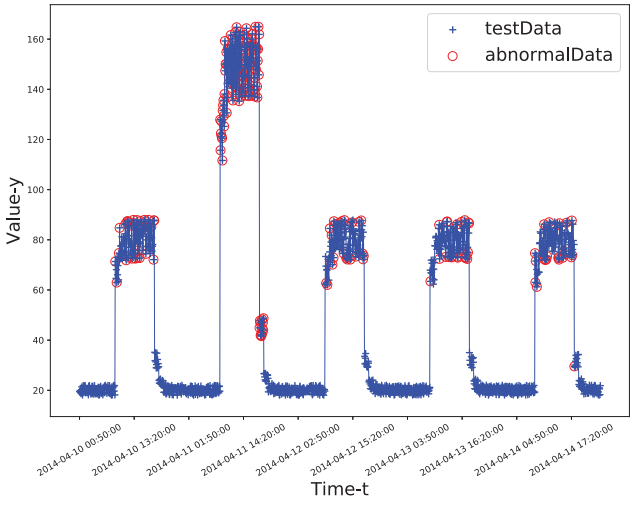
Figs. 4–7 show the results of four methods on the datasets from the AWS server. Fig. 4 shows that the results of anomaly detection by the four methods are similar on the 'ec2_cpu_utilization_24ae8d' dataset. In the Fig. 5, there is a slight concept dript after 2014-01-16 23:04:00. Specifically, the data in the latter part change more widely and their values are smaller than that of the data in the former part. As shown in Fig. 5, when data have a slight concept drift, the GPR-IADAM can add new data with small deviations from the prediction mean into the training data to update the model, and thus the GPR-IADAM can address the slight concept drift. The GPR-ADAM cannot handle any concept drift at all, so the GPR-ADAM has the worst performance. The GPR-AD always adds true data to the training data to update the model, so the GPR-AD can also deal with concept drift. The SGP-Q considers the abnormal degree of the current data point relative to that of previous data to decide whether to add the data point to train model, which is fully capable of handling concept drift. The proposed SGP-Q has the best performance on the 'ec2_cpu_utilization_825cc2' dataset.



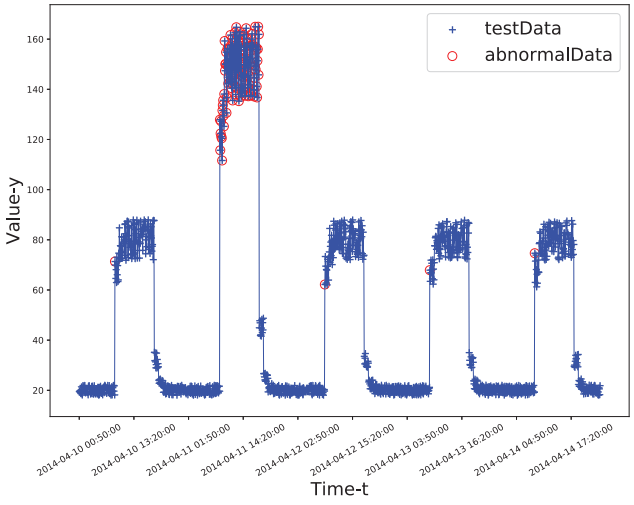
(a) GPR-AD



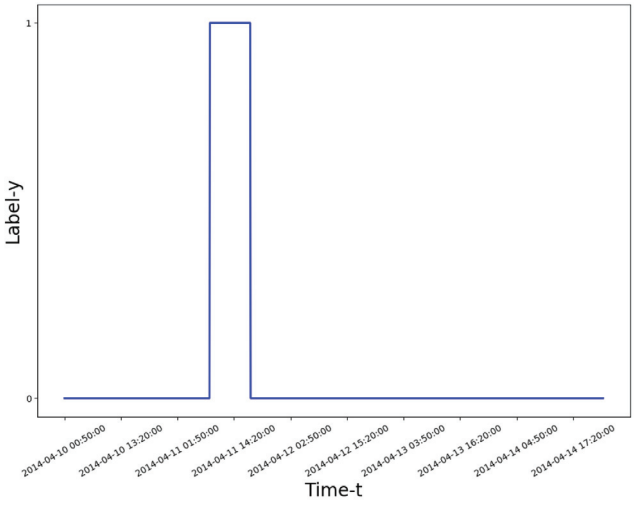
(b) GPR-ADAM



(c) GPR-IADAM

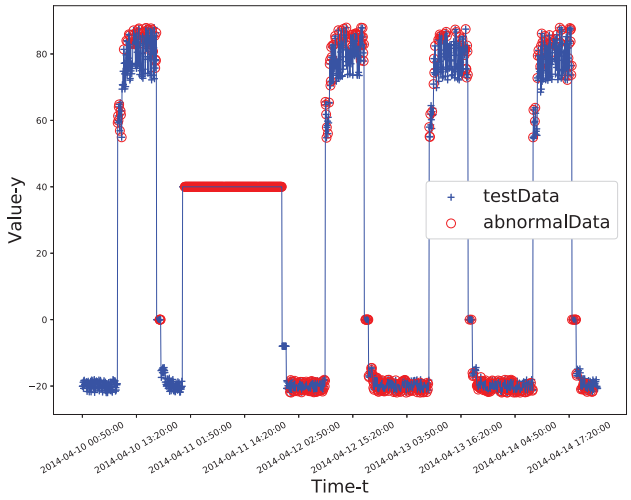


(d) SGP-Q

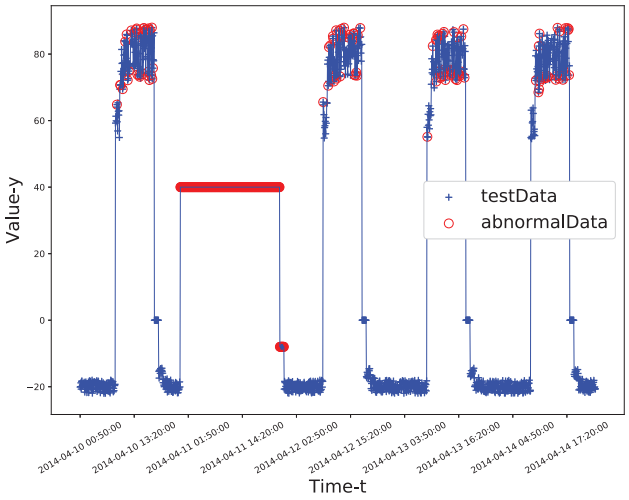


(e) Label

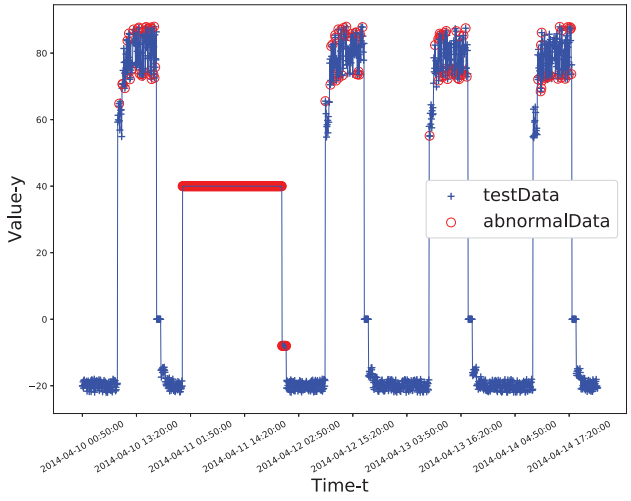
Fig. 2. (a), (b), (c) and (d) show the results of anomaly detection of four methods on the 'art_daily_jumpsup' dataset. (e) shows labels of test data, where 1 represents abnormal data and 0 represents normal data.



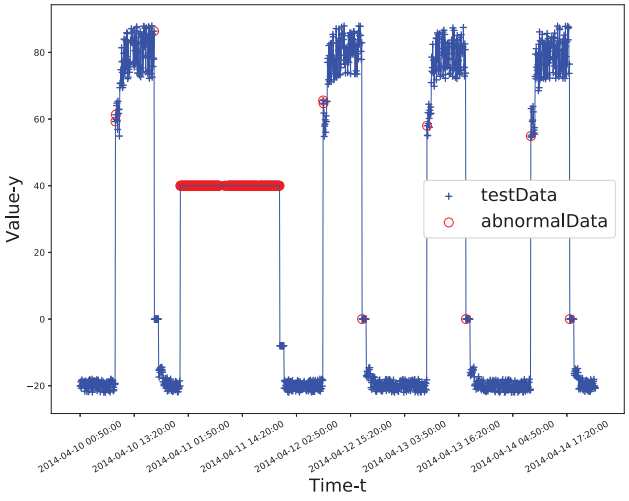
(a) GPR-AD



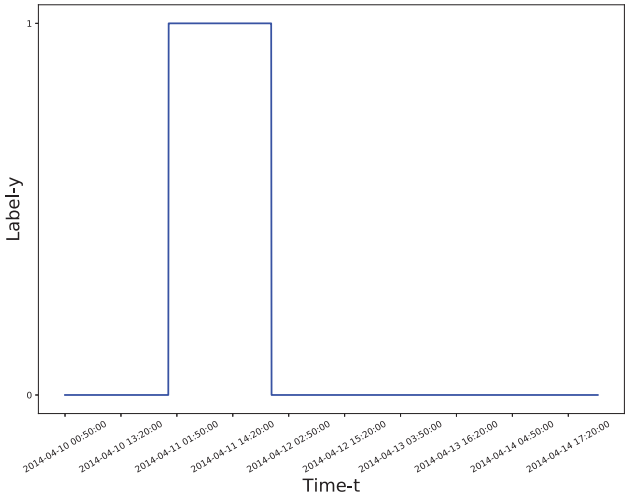
(b) GPR-ADAM



(c) GPR-IADAM

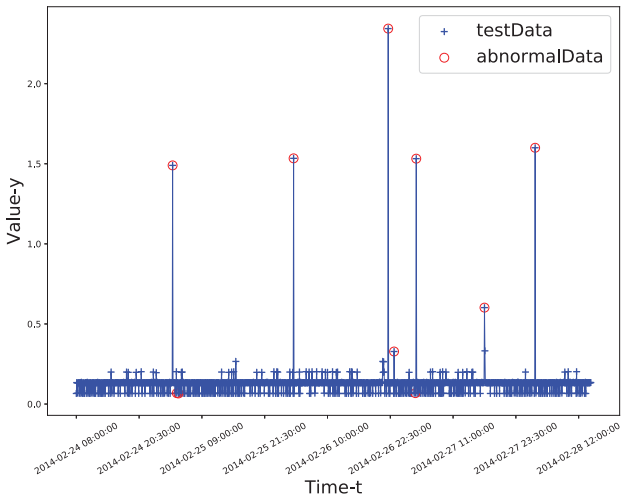


(d) SGP-Q

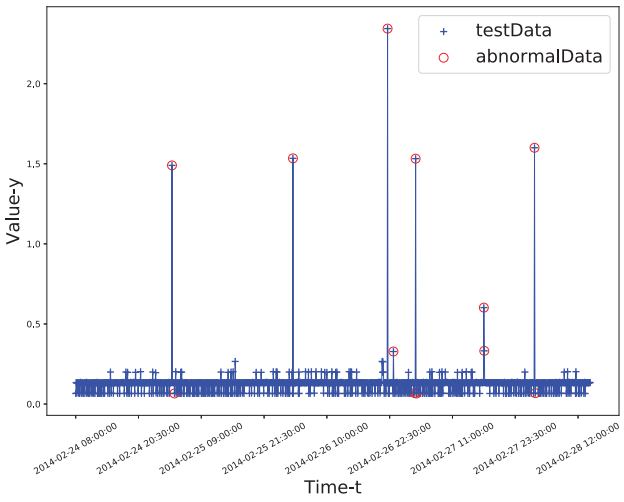


(e) Label

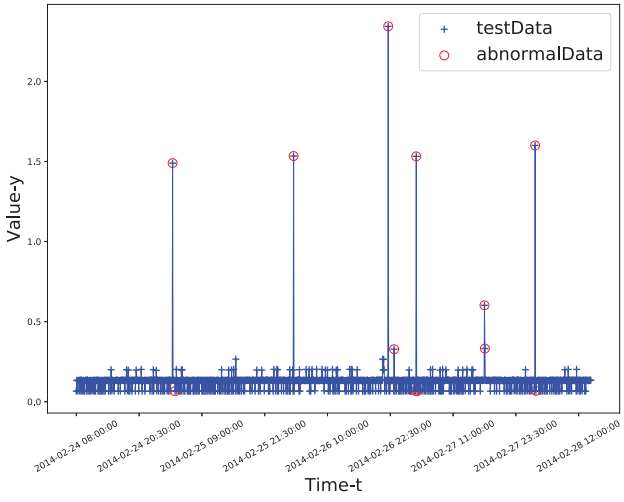
Fig. 3. (a), (b), (c) and (d) show the results of anomaly detection of four methods on the 'art_daily_flatmiddle' dataset. (e) shows labels of test data, where 1 represents abnormal data and 0 represents normal data.



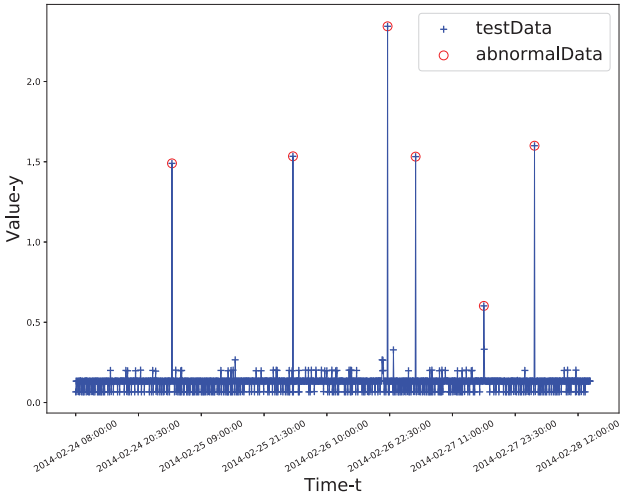
(a) GPR-AD



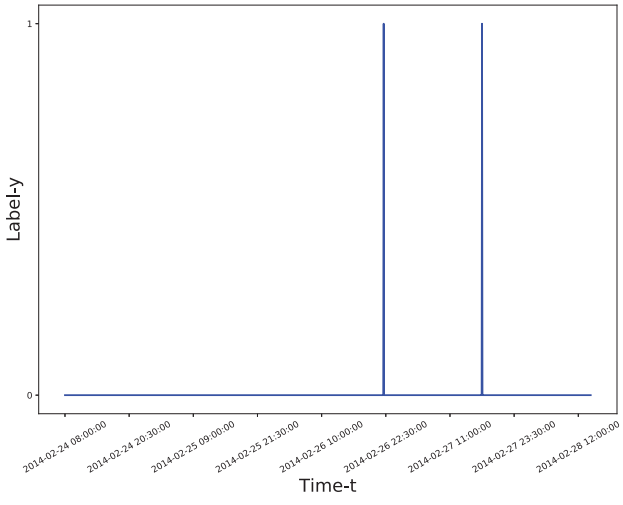
(b) GPR-ADAM



(c) GPR-IADAM

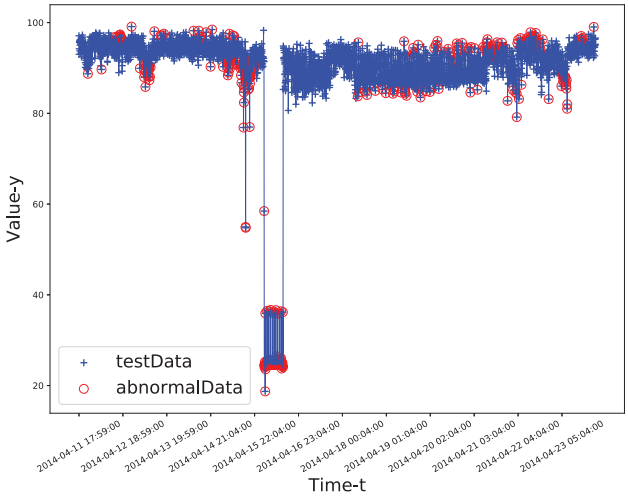


(d) SGP-Q

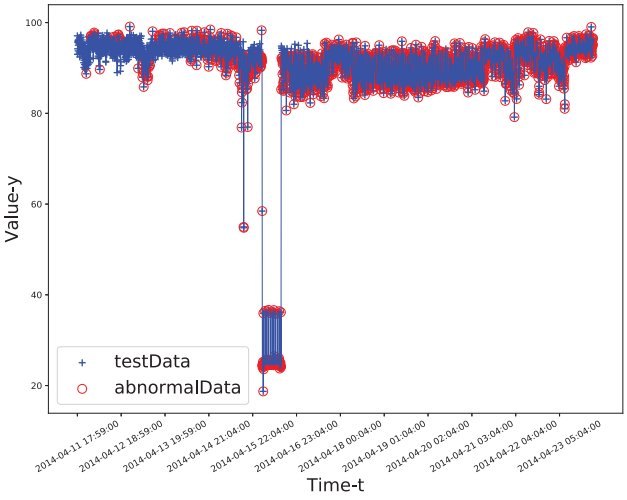


(e) Label

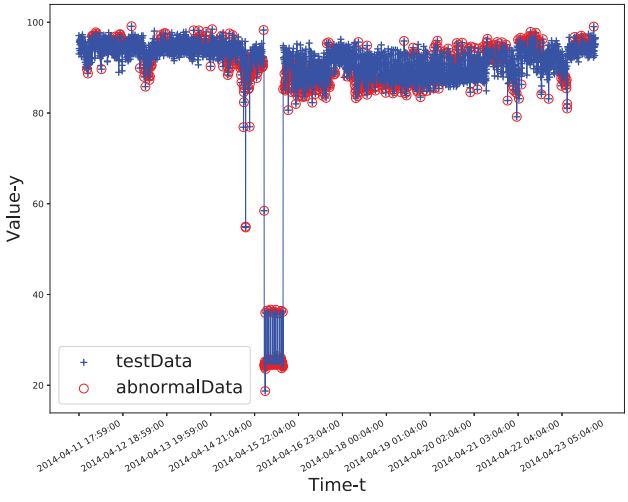
Fig. 4. (a), (b), (c) and (d) show the results of anomaly detection of four methods on the 'ec2_cpu_utilization_24ae8d' dataset. (e) shows labels of test data, where 1 represents abnormal data and 0 represents normal data.



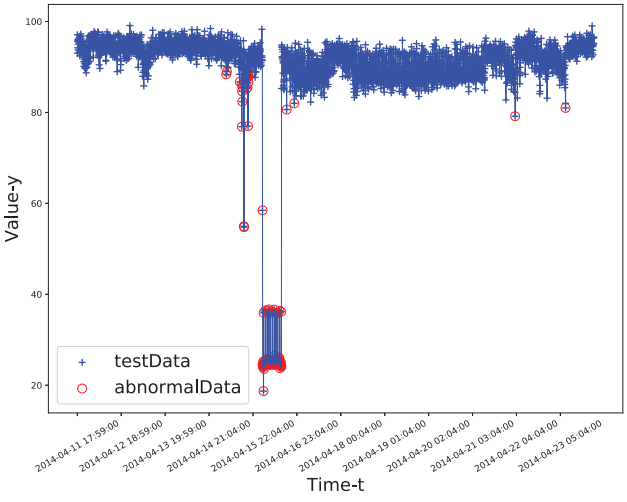
(a) GPR-AD



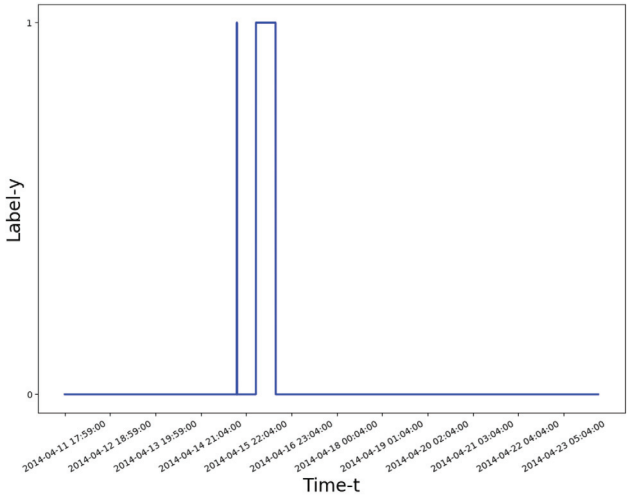
(b) GPR-ADAM



(c) GPR-IADAM

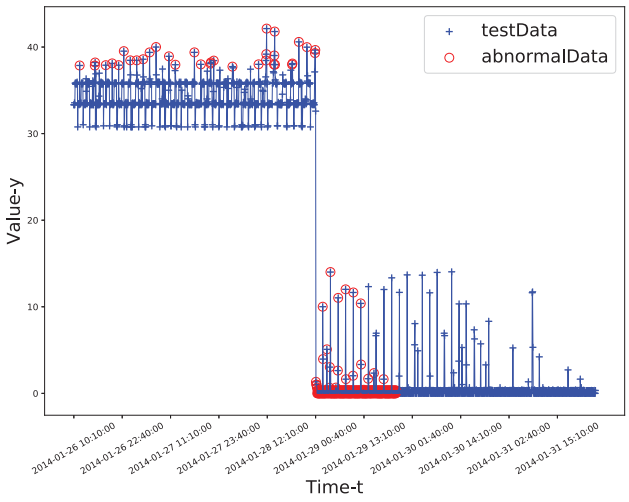


(d) SGP-Q

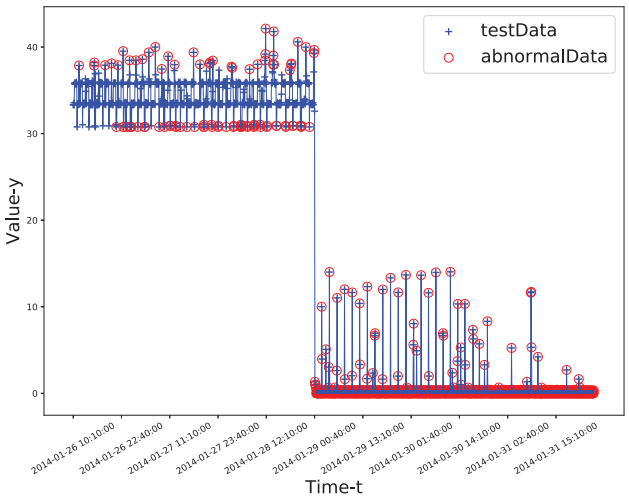


(e) Label

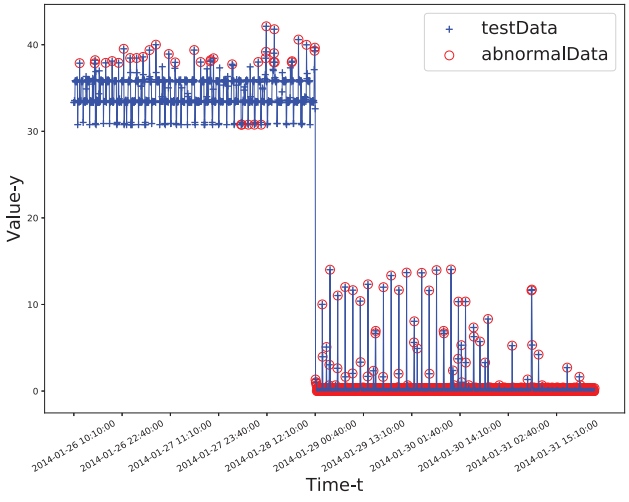
Fig. 5. (a), (b), (c) and (d) show the results of anomaly detection of four methods on the 'ec2_cpu_utilization_825cc2' dataset. (e) shows labels of test data, where 1 represents abnormal data and 0 represents normal data.



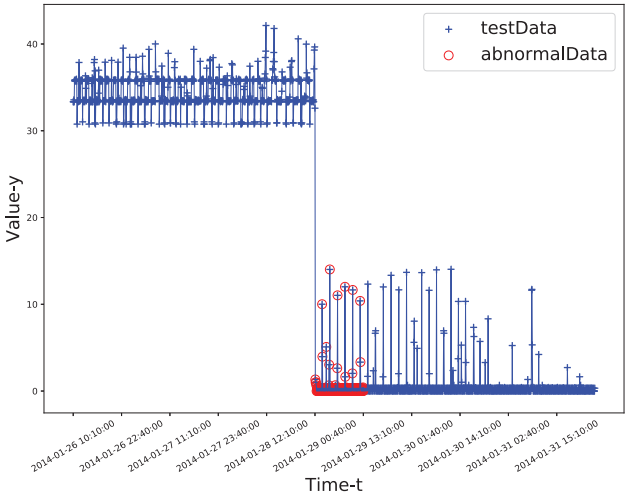
(a) GPR-AD



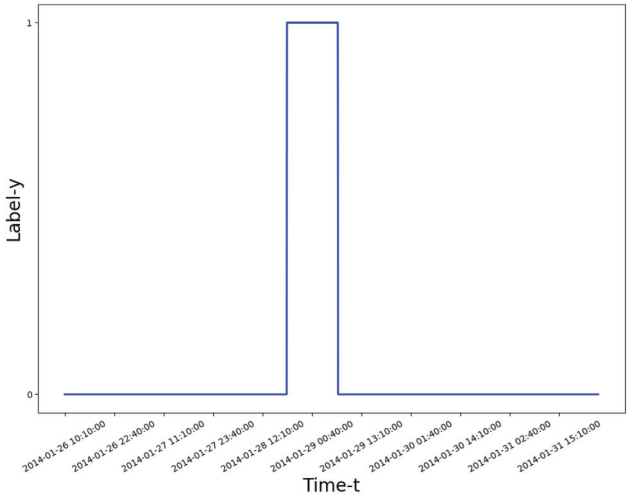
(b) GPR-ADAM



(c) GPR-IADAM

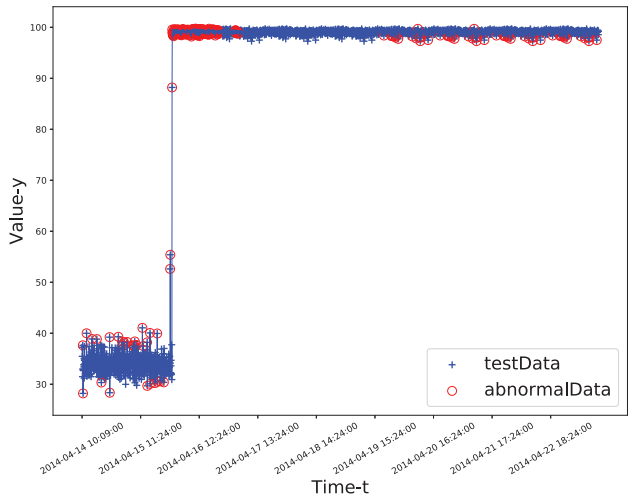


(d) SGP-Q

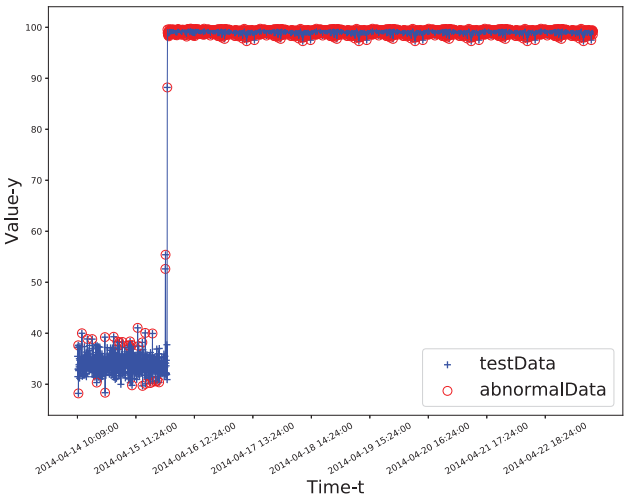


(e) Label

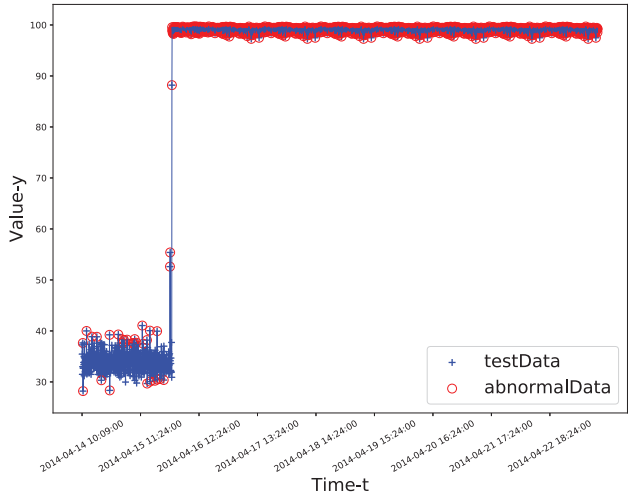
Fig. 6. (a), (b), (c) and (d) show the results of anomaly detection of four methods on the 'grok_asg_anomaly' dataset. (e) shows labels of test data, where 1 represents abnormal data and 0 represents normal data.



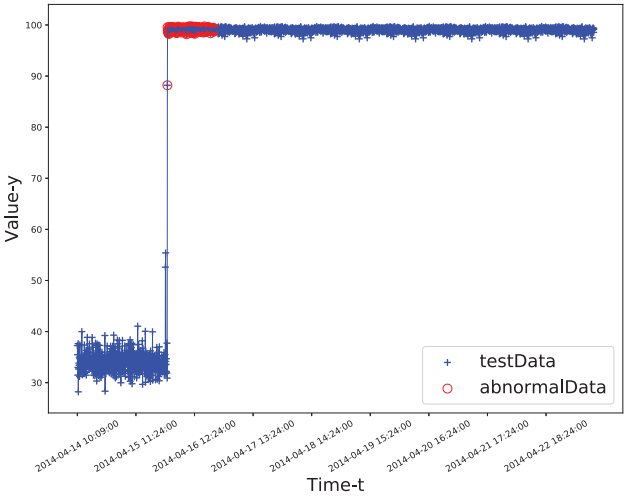
(a) GPR-AD



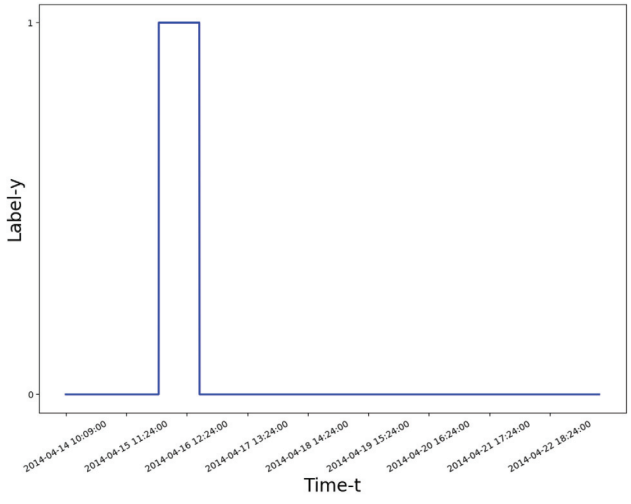
(b) GPR-ADAM



(c) GPR-IADAM



(d) SGP-Q



(e) Label

Fig. 7. (a), (b), (c) and (d) show the results of anomaly detection of four methods on the 'ec2_cpu_utilization_ac20cd' dataset. (e) shows labels of test data, where 1 represents abnormal data and 0 represents normal data.

Table 1
The $F_1(\%)$ score of four methods on nine datasets. The best results are indicated in bold.

Dataset	GPR-AD	GPR-ADAM	GPR-IADAM	SGP-Q
art_daily_jumpsup	82.20 ± 0.40	91.40 ± 0.80	92.80 ± 0.40	**99.20 ± 0.40**
art_daily_flatmiddle	74.20 ± 0.45	93.20 ± 0.45	93.20 ± 0.45	**96.40 ± 1.09**
ec2_cpu_utilization_24ae8d	98.90 ± 0.74	98.70 ± 0.45	98.70 ± 0.45	**99.00 ± 0.82**
ec2_cpu_utilization_825cc2	93.60 ± 0.55	44.40 ± 0.89	86.40 ± 0.55	**99.60 ± 0.55**
ec2_cpu_utilization_ac20cd	95.40 ± 0.55	29.40 ± 0.55	30.20 ± 0.45	**96.20 ± 1.17**
grok_asg_anomaly	85.20 ± 0.84	52.00 ± 0.00	55.00 ± 0.00	**90.00 ± 1.41**
occupancy_t4013	80.00 ± 0.00	80.40 ± 0.90	80.00 ± 0.00	**83.00 ± 1.22**
speed_t4013	81.40 ± 0.89	77.60 ± 0.54	81.60 ± 0.55	**84.00 ± 0.00**
ec2_cpu_utilization_5f5533	92.40 ± 0.89	88.40 ± 0.55	92.60 ± 0.90	**93.40 ± 1.41**

Table 2
The $F_1(\%)$ score of five methods on nine datasets. The best results are indicated in bold.

Dataset	CAD-OSE	HTM	SGP-DDM	SGP-EDDM	SGP-Q
art_daily_jumpsup	90.14 ± 0.42	83.67 ± 0.53	99.10 ± 0.41	99.18 ± 0.38	**99.20 ± 0.40**
art_daily_flatmiddle	90.07 ± 0.25	83.61 ± 0.46	95.20 ± 0.92	96.30 ± 0.95	**96.40 ± 1.09**
ec2_cpu_utilization_24ae8d	90.13 ± 0.32	80.52 ± 0.42	98.96 ± 0.81	**99.00 ± 0.72**	99.00 ± 0.82
ec2_cpu_utilization_825cc2	90.14 ± 0.42	83.67 ± 0.53	99.50 ± 0.56	99.50 ± 0.49	**99.60 ± 0.55**
ec2_cpu_utilization_ac20cd	90.09 ± 0.52	89.03 ± 0.48	95.27 ± 1.03	96.11 ± 1.20	**96.20 ± 1.17**
grok_asg_anomaly	**97.18 ± 0.53**	95.06 ± 0.48	92.00 ± 1.15	92.05 ± 1.25	90.00 ± 1.41
occupancy_t4013	78.36 ± 0.52	**93.03 ± 0.44**	82.10 ± 1.31	82.86 ± 1.15	83.00 ± 1.22
speed_t4013	78.13 ± 0.51	**93.41 ± 0.56**	82.05 ± 0.10	83.59 ± 0.08	84.00 ± 0.00
ec2_cpu_utilization_5f5533	92.50 ± 0.52	92.44 ± 0.48	93.00 ± 0.99	93.33 ± 1.21	**93.40 ± 1.41**

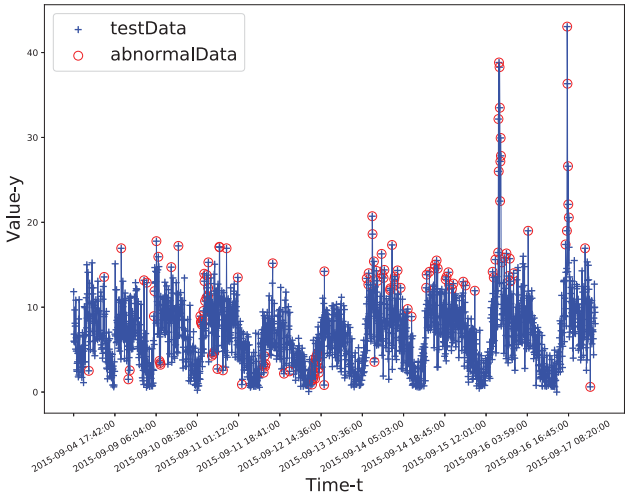
Table 3
The $F_1(\%)$ score of four methods on nine datasets. The best results are indicated in bold.

Dataset	SGP-AD	SGP-ADAM	SGP-IADAM	SGP-Q
art_daily_jumpsup	83.29 ± 0.43	92.35 ± 0.75	92.90 ± 0.41	**99.20 ± 0.40**
art_daily_flatmiddle	75.38 ± 1.01	93.20 ± 0.95	93.26 ± 1.05	**96.40 ± 1.09**
ec2_cpu_utilization_24ae8d	98.89 ± 0.79	98.60 ± 0.85	98.85 ± 0.92	**99.00 ± 0.82**
ec2_cpu_utilization_825cc2	93.80 ± 0.65	44.30 ± 0.77	86.20 ± 0.58	**99.60 ± 0.55**
ec2_cpu_utilization_ac20cd	95.37 ± 1.02	30.05 ± 1.11	29.86 ± 1.32	**96.20 ± 1.17**
grok_asg_anomaly	85.18 ± 1.14	52.06 ± 0.02	55.04 ± 0.03	**90.00 ± 1.41**
occupancy_t4013	80.10 ± 0.00	80.41 ± 1.25	80.07 ± 0.01	**83.00 ± 1.22**
speed_t4013	81.40 ± 0.83	77.62 ± 0.51	81.55 ± 0.65	**84.00 ± 0.00**
ec2_cpu_utilization_5f5533	92.41 ± 1.43	88.33 ± 1.26	92.71 ± 1.50	**93.40 ± 1.41**

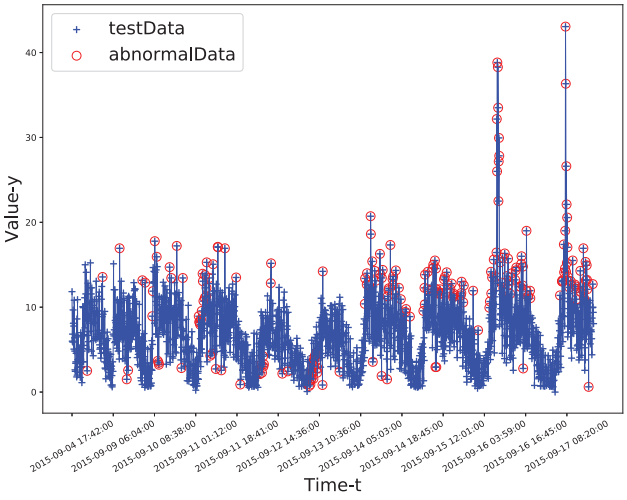
In Figs. 6 and 7, concept drift occurs in the time-series data, and the deviation between new data and old data is huge. In this case, the data that initially change are considered as abnormal data. However, when the 'abnormal' behaviors last for a while, the new data should be considered as normal data, and the model should relearn the characteristics of normal data. As shown in Figs. 6 and 7, the GPR-ADAM and GPR-IADAM are unable to deal with the concept drift when new data deviate greatly from old data, and all data after the concept drift are marked as abnormal data. Therefore, the GPR-ADAM and GPR-IADAM perform poorly on the 'grok_asg_anomaly' and 'ec2_cpu_utilization_ac20cd' datasets. Whether data are abnormal or not, the GPR-AD always adds true data instead of prediction mean to update the GPR model. Therefore, the GPR-AD can deal with concept drift in this case by adding new data to update the GPR model, and thus the performance of the GPR-AD is better than that of the GPR-ADAM and GPR-IADAM. The SGP-Q considers the information of previous and current data rather than the information of the current data point to measure the abnormal degree of the current data point, which is more reasonable. Therefore, the SGP-Q can deal with the concept drift well when the deviation between new data and old data is large. The SGP-Q has the best performance on the 'grok_asg_anomaly' and 'ec2_cpu_utilization_ac20cd' datasets.

Figs. 8 and 9 show results of four methods on the real-time traffic data. From Figs. 8 and 9, we can see that the GPR-AD, GPR-ADAM and GPR-IADAM mark many normal data as abnormal data. The number of normal data that are marked as abnormal data is the lowest in the SGP-Q. Combined with the numerical results in Table 1, the SGP-Q performs best on the 'occupancy_t4013' and 'speed_t4013' datasets.

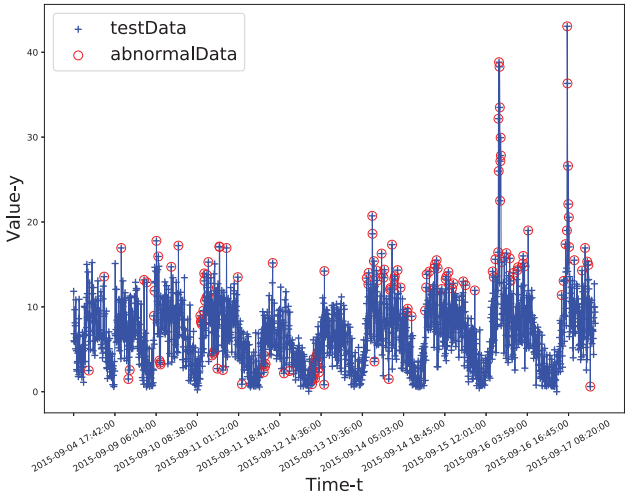
Our method SGP-Q achieves better performance than the GPR-AD, GPR-ADAM, and GPR-IADAM for both numerical and visual results. The reasons for good performance are listed as follows. Firstly, compared with the GPR-AD, the SGP-Q can make use of as few abnormal data as possible in the training data by the strategy based on likelihood and Q-function, which can make the SGPR model more accurate and improve the performance of anomaly detection. Secondly, the SGP-Q can address concept drift in data, while GPR-ADAM cannot. Thirdly, the GPR-IADAM only uses the information of the current data point to measure the abnormal degree of the current data point. Only when the deviation between the current data point and the prediction mean is small, the data point can be added to the training data. Therefore, the GPR-IADAM can only deal with concept drift with a slight deviation between new data and old data. The SGP-Q takes into account the information of previous and current data to measure the abnormal degree of the current data point relative to that of the previous data, which is more reasonable than only using the information of the current data point. In addition, whether the deviation between the new data and old data in the concept drift is large or slight, the SGP-Q can handle the concept drift well using Q-function. The SGP-Q overcomes the limitations of the GPR-AD, GPR-ADAM, and GPR-IADAM. Therefore, the performance of the SGP-Q is better than that of the GPR-AD, GPR-ADAM, and GPR-IADAM.



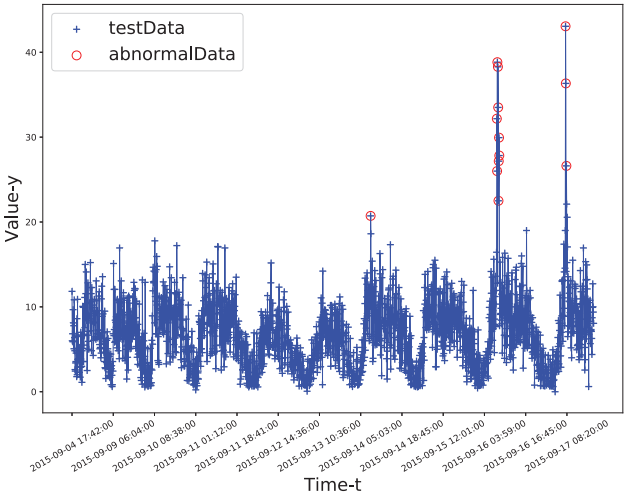
(a) GPR-AD



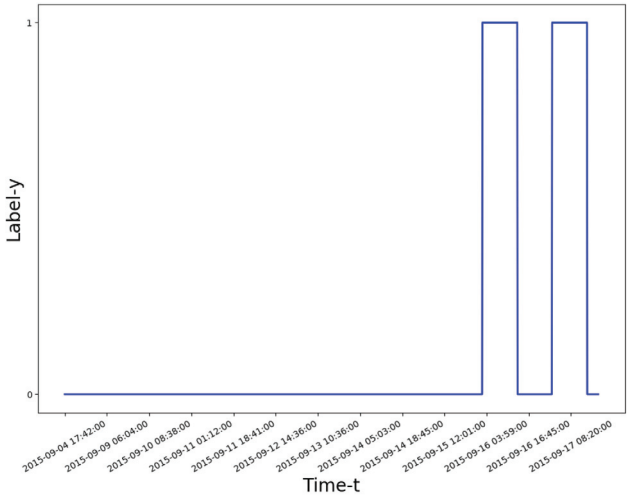
(b) GPR-ADAM



(c) GPR-IADAM

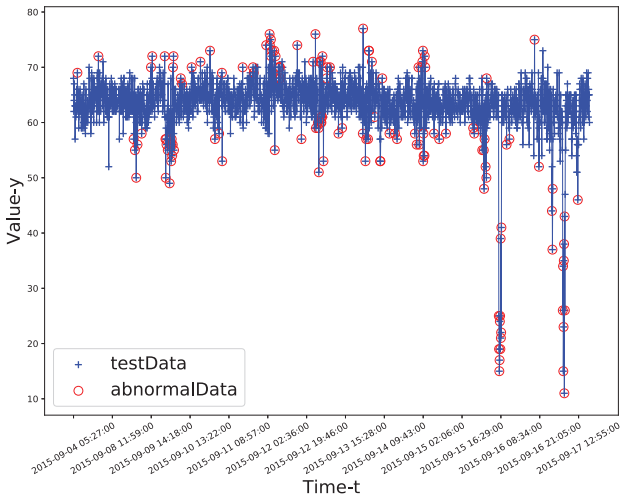


(d) SGP-Q

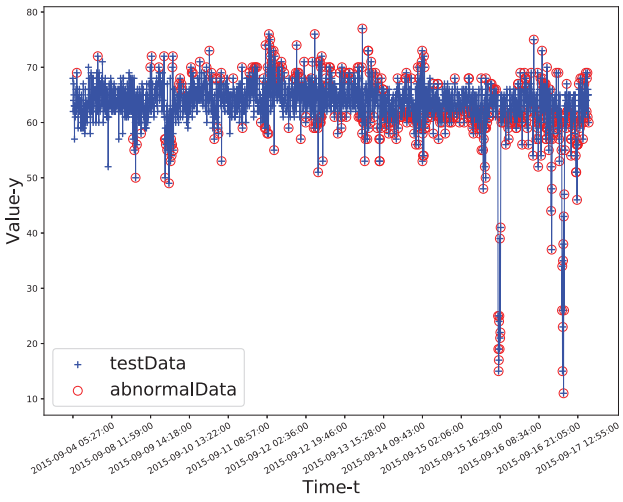


(e) Label

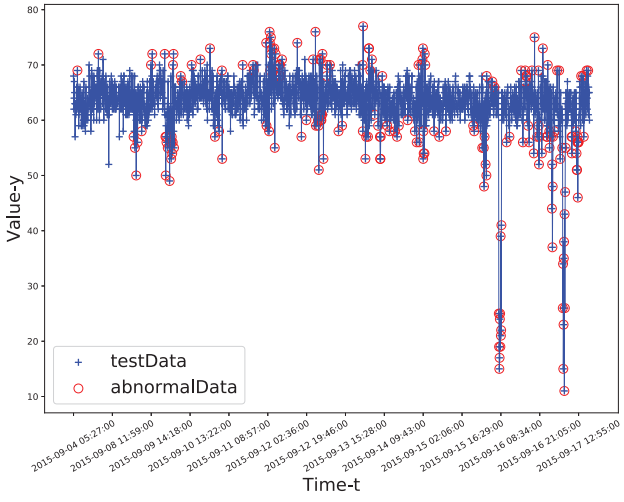
Fig. 8. (a), (b), (c) and (d) show the results of anomaly detection of four methods on the 'occupancy_t4013' dataset. (e) shows labels of test data, where 1 represents abnormal data and 0 represents normal data.



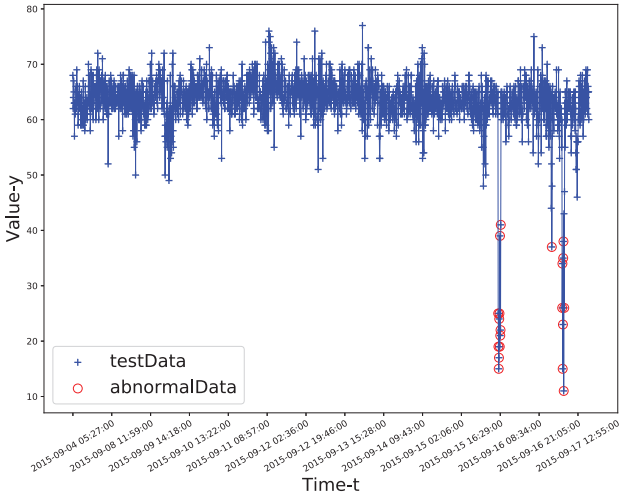
(a) GPR-AD



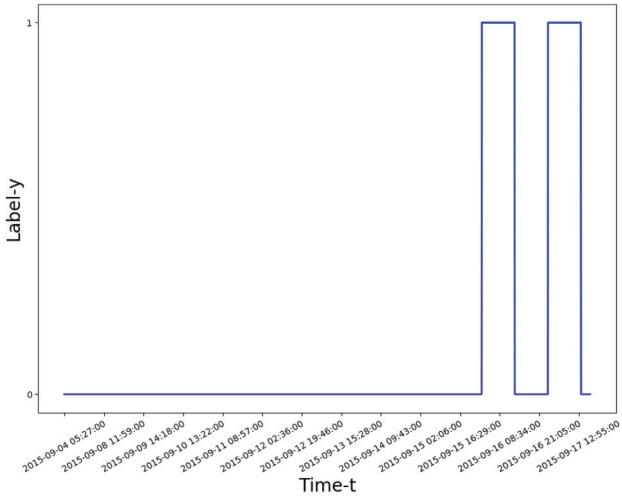
(b) GPR-ADAM



(c) GPR-IADAM



(d) SGP-Q



(e) Label

Fig. 9. (a), (b), (c) and (d) show the results of anomaly detection of four methods on the 'speed_t4013' dataset. (e) shows labels of test data, where 1 represents abnormal data and 0 represents normal data.

4.4. Summary

We summarize the applicable scenarios for each method. Firstly, the GPR-AD updates the model with true data regardless of whether data are abnormal or not. On the one hand, the GPR-AD is able to handle concept drift. On the other hand, there are too many abnormal data in the training data, resulting in the inaccurate GPR models and wrong anomaly detection results. Secondly, when the current data point is abnormal, the GPR-ADAM adds the prediction mean to update the GPR model and cannot address the concept drift. Thirdly, the GPR-IADAM uses the value of β to measure the abnormal degree of the current data point. When the abnormal degree of the current data point is low, the data point is used to update the model; when the abnormal degree of the current data point is high, the prediction mean is used to update the model. Therefore, the GPR-IADAM can only deal with the concept drift when the deviation between new data and old data is slight. Fourthly, the SGP-Q uses the information of previous and current data instead of the information of the current data point to measure the abnormal degree of the current data point relative to that of previous data. Regardless of whether the deviation between new data and old data is large or slight, the SGP-Q can address the concept drift well. In addition, the SGP-Q can obtain more accurate SGPR models and better anomaly detection results by making use of few abnormal data in the training data through the strategy based on likelihood and Q-function.

The number of points in the inducing input Z is set to 100. Increasing the number of points in the inducing input will increase the training time, while the performance improvement is tiny. The covariance function in the experiment is set to the sum of the RBF kernel function and the linear kernel function. We have tried some more complex kernel functions, such as the multi-layer perceptron (MLP) kernel function and Matern32 kernel function. Complex kernel functions increase the number of kernel parameters that need to be optimized, resulting in much slower training speed but weeny performance improvement.

The actual time-series data are long sequences with timestamps. For the convenience of calculation, the timestamps must be quantized. Two numerical methods have been tried. The first method is that the time of the first data point in the time-series data is recorded as 0, and the subsequent time is quantized to the number of minutes between the current time and start time multiplied by 0.01. The second method is to slice long time-series data, the daily 00:00:00 is quantized to 0, and the other time is quantized to the number of minutes between the current time and today's 00:00:00 multiplied by 0.01. In the first numerical method, modeling the mapping from input to output requires more complex composite kernel functions. For example, modeling periodic data needs to add a periodic kernel function to the composite kernel function. In the second numerical method, a relatively simple composite kernel function, that is, the sum of the RBF kernel function and the linear kernel function is able to model the mapping from input to output well. Even if the data is periodic, there is no need to add the periodic kernel function into composite kernel function, because the periodic information is already included in the numerical time. The second numerical method combined with a relatively simple kernel function is faster than the first numerical method combined with a complex kernel function, and their performance is quite similar. In the experiment, we use the second numerical method, and the experimental results confirm the validity and rationality of the second numerical method.

5. Conclusion

In this paper, we have proposed the SGP-Q method, which improves the existing online anomaly detection methods based on GPs. As an online anomaly detection method, the SGP-Q uses SGPs with lower time complexity to model time-series data and accelerates online anomaly detection. Concept drift is common in time-series data, and it is essential for online anomaly detection methods to have the abilities to redefine the meanings of 'abnormal' behaviors and adapt to concept drift. On account of using Q-function, the SGP-Q can address concept drift well. Moreover, the SGP-Q employs the strategy based on likelihood and Q-function to update training data. This strategy can reduce the abnormal data in the training data and make the SGPR model more accurate, thus improving the performance of anomaly detection.

In experiments, we conducted experiments on various artificial and real-world datasets and compared the proposed SGP-Q with the existing anomaly detection methods based on GPs, including the GPR-AD, GPR-ADAM, and GPR-IADAM. The proposed SGP-Q obtains better performance than the GPR-AD, GPR-ADAM, and GPR-IADAM.

For future work, it will be more challenge and interesting to employ the mixture of Gaussian processes to model time-series data in the task of online anomaly detection, as real-world time-series data are usually in the 'multi-modality' distributions.

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.

CRediT authorship contribution statement

Minghao Gu: Software, Validation, Investigation, Data curation, Writing - review & editing. **Jingjing Fei:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft. **Shiliang Sun:** Conceptualization, Methodology, Resources, Writing - review & editing, Supervision, Project administration, Funding acquisition.

Acknowledgements

This work is supported by the National Natural Science Foundation of China under Project 61673179, Shanghai Knowledge Service Platform Project (No. ZF1213), and the Fundamental Research Funds for the Central Universities.

References

[1] L. Xu, W. He, S. Li, Internet of things in industries: a survey, IEEE Trans. Ind. Inf. 10 (2014) 2233–2243.
[2] V. Chandola, V. Mithal, V. Kumar, Comparative evaluation of anomaly detection techniques for sequence data, in: Proceedings of the IEEE International Conference on Data Mining, 2008, pp. 743–748.
[3] S. Ahmad, A. Lavin, S. Purdy, Z. Agha, Unsupervised real-time anomaly detection for streaming data, Neurocomputing 262 (2017) 134–147.
[4] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: a survey, ACM Comput. Surv. 41 (2009) 15–86.
[5] M. Gupta, J. Gao, C.C. Aggarwal, J.W. Han, Outlier detection for temporal data: a survey, IEEE Trans. Knowl. Data Eng. 26 (2014) 2250–2267.
[6] R.A.A. Habeeb, F. Nasaruddin, A. Gani, I.A.T. Hashem, E. Ahmed, I. Muhammad, Real-time big data processing for anomaly detection: a survey, Int. J. Inf. Manag. 45 (2019) 289–307.
[7] D. Ramotsoela, A. Abu-Mahfouz, G. Hancke, A survey of anomaly detection in industrial wireless sensor networks with critical water system infrastructure as a case study, Sensors 18 (2018) 2491–2514.
[8] M. Salehi, L. Rashidi, A survey on anomaly detection in evolving data with application to forest fire risk prediction, ACM SIGKDD Explor. Newslett. 20 (2018) 13–23.
[9] G. Fernandes, J.J.R.C. Rodrigues, L.F. Carvalho, J.F. Al-Muhtadi, M.L. Proença, A comprehensive survey on network anomaly detection, Telecommun. Syst. 70 (2019) 447–489.
[10] J. Noble, N. Adams, Real-time dynamic network anomaly detection, IEEE Intell. Syst. 33 (2018) 5–18.
[11] R. Chalapathy, S. Chawla, Deep learning for anomaly detection: a survey, arXiv:1901.03407 (2019).

[12] Z. Zhao, K.G. Mehrotra, C.K. Mohan, Online anomaly detection using random forest, in: Proceedings of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems, 2018, pp. 135–147.
[13] F. Sönmez, M. Zontul, O. Kaynar, H. Tutar, Anomaly detection using data mining methods in IT systems: a decision support application, Sakarya Univ. J. Sci. 22 (2018) 1109–1123.
[14] M. Farshchi, J. Schneider, I. Weber, J. Grundy, Metric selection and anomaly detection for cloud operations using log and metric correlation analysis, J. Syst. Softw. 137 (2018) 531–549.
[15] V. Kumar, Parallel and distributed computing for cybersecurity, IEEE Distrib. Syst. Online 6 (2005) 1–9.
[16] C. Spence, L. Parra, P. Sajda, Detection, synthesis and compression in mammographic image analysis with a hierarchical image probability model, in: Proceedings of the IEEE Workshop on Mathematical Methods in Biomedical Image Analysis, 2001, pp. 3–10.
[17] R. Fujimaki, T. Yairi, K. Machida, An approach to spacecraft anomaly detection problem using kernel feature space, in: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, 2005, pp. 401–410.
[18] C.E. Rasmussen, C.K.I. Williams, Gaussian Processes for Machine Learning, MIT Press, 2006.
[19] C.K. Williams, M. Seeger, Using the Nyström method to speed up kernel machines, Adv. Neural Inf. Process. Syst. 14 (2001) 682–688.
[20] R. Herbrichf, N.D. Lawrence, M. Seeger, Fast sparse gaussian process methods: the informative vector machine, Adv. Neural Inf. Process. Syst. 16 (2003) 625–632.
[21] J. Quiñonero-Candela, C.E. Rasmussen, A unifying view of sparse approximate gaussian process regression, J. Mach. Learn. Res. 6 (2005) 1939–1959.
[22] E. Snelson, Z. Ghahramani, Sparse gaussian processes using pseudo-inputs, Adv. Neural Inf. Process. Syst. 19 (2006) 1257–1264.
[23] M.K. Titsias, Variational learning of inducing variables in sparse Gaussian processes, in: Proceedings of the Artificial Intelligence and Statistics, 2009, pp. 567–574.
[24] J. Hensman, N. Fusi, N.D. Lawrence, Gaussian processes for big data, in: Proceedings of the Gaussian process for Big Data, 2013, pp. 282–290.
[25] S. Sun, J. Zhao, J. Zhu, A review of Nyström methods for large-scale machine learning, Inf. Fusion 26 (2015) 36–48.
[26] Q. Liu, S. Sun, Sparse multimodal Gaussian processes, in: Proceedings of the International Conference on Intelligent Science and Big Data Engineering, 2017, pp. 28–40.
[27] Y. Gal, M.V.D. Wilk, C.E. Rasmussen, Distributed variational inference in sparse gaussian process regression and latent variable models, Adv. Neural Inf. Process. Syst. 27 (2014) 3257–3265.
[28] M.P. Deisenroth, J.W. Ng, Distributed Gaussian processes, in: Proceedings of the International Conference on Machine Learning, 2015, pp. 1481–1490.
[29] Y. Peng, J.Y. Pang, G. Song, D.T. Liu, X.Y. Peng, Improved Gaussian process monitoring data based on the regression model state flow abnormality detection method. C.N. Patent CN103974311B, 2017.
[30] A. Tsymbal, The problem of concept drift: definitions and related work 106 (2004) 58–65. Computer Science Department, Trinity College Dublin
[31] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, A. Bouchachia, A survey on concept drift adaptation, ACM Comput Surv 46 (2014) 44.
[32] M. Ma, S. Zhang, D. Pei, X. Huang, H. Dai, Robust and rapid adaption for concept drift in software system anomaly detection, in: Proceedings of the International Symposium on Software Reliability Engineering, 2018, pp. 13–24.
[33] I. Khamassi, M. Sayed-Mouchaweh, M. Hammami, K. Ghédira, Discussion and review on evolving data streams and concept drift adapting, Evolv. Syst. 9 (2018) 1–23.
[34] P.R.L. Almeida, L.S. Oliveira, A.S.B. Jr, R. Sabourin, Adapting dynamic classifier selection for concept drift, Expert Syst. Appl. 104 (2018) 67–85.
[35] S. Wang, L.L. Minku, X. Yao, A systematic study of online class imbalance learning with concept drift, IEEE Trans. Neural Netw. Learn. Syst. 99 (2018) 1–20.
[36] J. Demšar, Z. Bosnić, Detecting concept drift in data streams using model explanation, Expert Syst. Appl. 92 (2018) 546–559.
[37] I. Goldenberg, G.I. Webb, Survey of distance measures for quantifying concept drift and shift in numeric data, Knowl. Inf. Syst. (2018) 1–25.
[38] T. Escovedo, A. Koshiyama, A.A. da Cruz, M. Vellasco, Detecta: abrupt concept drift detection in non-stationary environments, Appl. Soft Comput. 62 (2018) 119–133.
[39] G.K. Karagiannidis, A.S. Lioumpas, An improved approximation for the gaussian q-function, IEEE Commun. Lett. 11 (2007) 644–646.
[40] M. Pratama, J. Lu, E. Lughofer, G. Zhang, S. Anavatti, Scaffolding type-2 classifier for incremental learning under concept drifts, Neurocomputing 191 (2016) 304–329.
[41] M.K. Simon, Probability Distributions Involving Gaussian Random Variables: A Handbook for Engineers and Scientists, Springer Science & Business Media, 2007.
[42] D. Zwillinger, Table of Integrals, Series, and Products, Elsevier, 2014.
[43] M. Chiani, D. Dardari, M.K. Simon, New exponential bounds and approximations for the computation of error probability in fading channels, IEEE Trans. Wirel. Commun. 2 (2003) 840–845.
[44] P.O. Borjesson, C.E. Sundberg, Simple approximations of the error function q(x) for communications applications, IEEE Trans. Commun. 27 (1979) 639–643.
[45] M. Smirnov, Contextual anomaly detector, Contextual Anomaly Detector https://github.com/smirmik/CAD.
[46] Y. Cui, S. Ahmad, J. Hawkins, Continuous online sequence learning with an unsupervised neural network model, Neural Comput. 28 (11) (2016) 2474–2504.
[47] J. Gama, P. Medas, G. Castillo, P. Rodrigues, Learning with drift detection, in: Proceedings of the Brazilian Symposium on Artificial Intelligence, Springer, 2004, pp. 286–295.
[48] M. Baena-Garcıa, J. del Campo-Ávila, R. Fidalgo, A. Bifet, R. Gavalda, R. Morales-Bueno, Early drift detection method, in: Proceedings of the Fourth International Workshop on Knowledge Discovery from Data Streams, 6, 2006, pp. 77–86.

Minghao Gu is a master student in the Pattern Recognition and Machine Learning Research Group, East China Normal University. His research interests include machine learning, Monte Carlo methods, etc.

Jingjing Fei is a master student in the Pattern Recognition and Machine Learning Research Group, East China Normal University. Her research interests include machine learning, Gaussian processes, etc.

Shiliang Sun received the Ph.D. degree in pattern recognition and intelligent systems from the Department of Automation and the State Key Laboratory of Intelligent Technology and Systems, Tsinghua University, Beijing, China, in 2007. He is a Professor with the Department of Computer Science and Technology and the Head of the Pattern Recognition and Machine Learning Research Group, East China Normal University, Shanghai, China. From 2009 to 2010, he was a Visiting Researcher with the Department of Computer Science, Centre for Computational Statistics and Machine Learning, University College London, London, U.K. In 2014, he was a Visiting Researcher with the Department of Electrical Engineering, Columbia University, New York, NY, USA. His current research interests include probabilistic models, multi-view learning, learning theory, approximate inference, sequential modeling, and their applications. His research results have expounded in 100+ publications at peer-reviewed journals and conferences, such as JMLR, IEEET-NNLS, IEEET-Cybernetics, IEEET-MM, IEEET-ITS, ICML, NIPS, IJCAI and ECML. Prof. Sun is on the Editorial Board of multiple international journals, including Neurocomputing and the IEEE Transactions on Neural Networks and Learning Systems.