Divergence-Based One-Class Classification Using Gaussian Processes

Paul Bodesheim
paul.bodesheim@uni-jena.de
Erik Rodner
erik.rodner@uni-jena.de
Alexander Freytag
alexander.freytag@uni-jena.de
Joachim Denzler
joachim.denzler@uni-jena.de

Computer Vision Group
Friedrich Schiller University of Jena
Ernst-Abbe-Platz 2
07743 Jena, Germany
http://www.inf-cv.uni-jena.de

Abstract

We present an information theoretic framework for one-class classification, which allows for deriving several new novelty scores. With these scores, we are able to rank samples according to their novelty and to detect outliers not belonging to a learnt data distribution. The key idea of our approach is to measure the impact of a test sample on the previously learnt model. This is carried out in a probabilistic manner using Jensen-Shannon divergence and reclassification results derived from the Gaussian process regression framework. Our method is evaluated using well-known machine learning datasets as well as large-scale image categorisation experiments showing its ability to achieve state-of-the-art performance.

1 Introduction

Detecting samples of unknown classes is a key task for active learning and one-class classification (OCC). Starting from a set of only positive training samples, we want to estimate a soft membership score for every new test sample. This score can then be used (1) to rank a set of test samples according to their novelty with respect to the training set; or (2) to perform thresholding and use the discrete decision to detect outliers [14]. Applying OCC methods is especially beneficial in situations where either negative data is difficult to model with given samples or where negative samples are hard to obtain.

A common strategy of kernel-based OCC is Parzen density estimation [1], where similarity scores between each of the training samples and the test samples are calculated with a kernel function and summed up. Another technique is support vector data description (SVDD) proposed by Tax and Duin [15]. Their main idea is to enclose the training samples with a hypersphere in feature space. A similar approach was introduced by Kemmler *et al*. [6] but motivated from a Gaussian process (GP) point of view [5, 13]. The authors derive different novelty measures from the GP framework by utilising the predictive mean and variance of the estimate. The work of [12] uses the GP latent variable model of [7] to apply a Gaussian mixture model in the estimated latent space allowing for flexible density estimation in the original input space.

http://dx.doi.org/10.5244/C.26.50

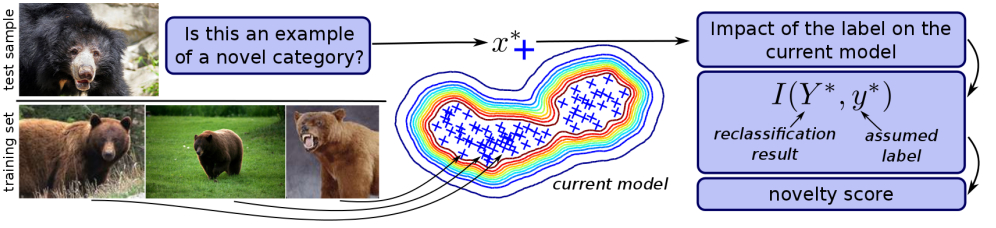


Figure 1: Outline of our approach based on mutual information.

Similar to [3], the key idea of our method is to measure the impact of a test sample on the previously learnt model when the sample would be treated as an additional training sample. While the approach of [3] is restricted to parametric models such as mixtures of Gaussians, we present a general framework for ranking and measuring novelty based on information theory and probabilities estimated with Gaussian process regression. The new framework sheds light on OCC from a completely different theoretical perspective. We derive several new OCC scores from this framework and evaluate them with standard classification as well as large-scale image categorisation experiments. An overview of our presented approach, which is based on mutual information and divergence measures of information theory, can be seen in Figure 1. Although our formulation is strongly related to active learning [5], we only consider one-class classification in this paper.

The contributions of this paper can be summarised as follows: (1) we consider one-class classification from a completely new perspective by proposing a framework based on divergence measures of information theory and (2) we highlight the connections to well-known information theoretic aspects like mutual information. Additionally, we derive a new one-class classification score from the Gaussian process regression framework and show that our presented approach achieves state-of-the-art performance on various datasets.

The paper is structured as follows: Section 2 explains the basic elements of our divergence framework. This framework is used in Section 3 together with a GP classification model to derive OCC novelty scores. Experiments on different applications including a large-scale image categorisation scenario are evaluated in Section 4. A summary of our findings concludes the paper.

2 Divergence-based one-class classification

In this section, we present our OCC framework based on mutual information and divergence measures of information theory. It allows for deriving novelty scores, which can be utilised in one-class classification scenarios. The idea of our approach is to measure how strongly a new test sample would influence the current model if it was used for training. We measure the expected change based on approximations of the divergence between the resulting models under different label assumptions for the test sample.

2.1 Basic idea and notation

In the following, we assume a given training set $\mathbf{X} = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(N)}\}$ with corresponding labels $\mathbf{y}$ and try to estimate a membership score of a test sample $\mathbf{x}^*$. Throughout this paper, we score the sample based on the resulting model change after treating it as an additional

training sample. We first note that we do not have to evaluate the resulting model change for the whole input space: if we use $\mathbf{x}^*$ as an additional training sample, the assumption of its true class, namely whether it belongs to the target class, mainly influences the decision function in the neighbourhood of $\mathbf{x}^*$. Consequently, we can approximate the change of the whole model by evaluating its change in a local neighborhood of $\mathbf{x}^*$. However, this strategy would still lead to costly sampling and model evaluations. Therefore, we approximate the change of the model by relying on a neighborhood of infinite small size and only taking the new sample itself and its reclassification result into account. The main idea of our approach is visualized in Figure 2.

To evaluate the change of the model, we once assume that $\mathbf{x}^*$ belongs to the target class ($y^* = 1$) and once assume the opposite ($y^* = -1$). Since we have no precise knowledge about the correct label of new samples, we model the assumed label $y^* \in \{1,-1\}$ as a binary random variable. We further introduce a second random variable $Y^* \in \{1,-1\}$ to evaluate the model on $\mathbf{x}^*$ after the model update, *i.e.*, the variable Y^* is the reclassification result of $\mathbf{x}^*$. Note that it is important to regularize the model complexity to obtain non-trivial reclassification results, which is in our case done by assuming noisy labels (Section 3). In the following, we show how to obtain a valid novelty score based on the introduced variables and the main idea stated above.

2.2 Measuring novelty with mutual information and divergence

When considering Figure 2, we notice that outliers lead to a significant change of the model in their local neighborhood under both label assumptions. In contrast, a sample from the target class can not be explained well using the assumption $y^* = -1$ and thus leads to a smaller change of the model in this case. We therefore note that for measuring the resulting model change, we can rely on the dependence between the assumed label y^* and the reclassification result Y^*, which can be measured using the mutual information $\mathrm{I}(Y^*, y^* \mid \mathbf{D}^*)$. Since both variables should be influenced by the available data, the mutual information depends on $\mathbf{D}^* = (\mathbf{X}, \mathbf{y}, \mathbf{x}^*)$, which contains the training set as well as the new test sample $\mathbf{x}^*$. The conditional mutual information can be written in terms of the conditional entropy H:

$$\mathrm{I}(Y^*, y^* \mid \mathbf{D}^*) = \mathrm{H}(Y^* \mid \mathbf{D}^*) - \mathrm{H}(Y^* \mid y^*, \mathbf{D}^*) \quad . \tag{1}$$

Obviously, this measure will be high if the conditional entropy $\mathrm{H}(Y^* \mid y^*, \mathbf{D}^*)$ is low. A low conditional entropy indicates that the reclassification result Y^* is almost certain given the assumed label y^*. Since the reclassification of $\mathbf{x}^*$ heavily depends on the training data $\mathbf{X}$ contained in $\mathbf{D}^*$, one achieves a low conditional entropy if the test sample $\mathbf{x}^*$ is far away from the training samples and the reclassification is mainly influenced by the assumed label y^*. An example is given in Figure 2. In case of a test sample similar to the training samples, the reclassification result is more affected by the training samples. As stated above, the assumption of $y^* = -1$ leads to a higher probability for a wrong reclassification and thus to a higher remaining uncertainty in terms of conditional entropy (bottom left plot in Figure 2). Summing up, a low conditional mutual information is induced by a strong membership to the target class and vice versa. We use the negative conditional mutual information as an OCC score, leading to low scores for possible outliers.

Calculating the conditional mutual information can be done by further expanding Eq. (1) based on the definition of conditional entropy and the random variables involved. We use shortcuts $\pi = \mathrm{p}(y^* = 1 \mid \mathbf{D}^*)$ and $(1-\pi) = \mathrm{p}(y^* = -1 \mid \mathbf{D}^*)$ for the prior probabilities of the

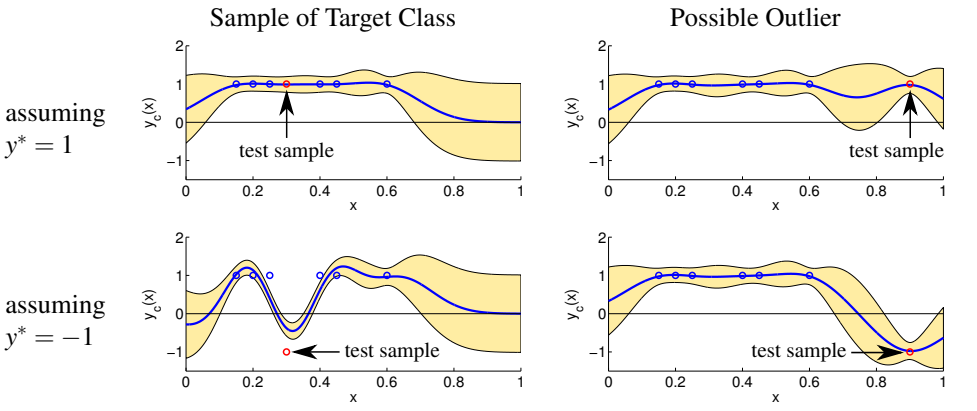


Figure 2: Visualization of our divergence approach. We want to measure the impact of a test sample on the current model. While both label assumptions $y^* \in \{1,-1\}$ of a possible outlier can be verified by reclassification using the model additionally trained with the test sample (blue curve), the assumption $y^* = -1$ will lead to a weak reclassification of a test sample stemming from the target class. Our approach exploits this difference. Classification uncertainty is visualised by shaded areas. The figure is best viewed in colour.

assumed label. Furthermore, we write $p_i^j = \mathrm{p}(Y^* = j \mid y^* = i, \mathbf{D}^*)$ for conditional probabilities of Y^* given the assumption about the label y^*. The corresponding distribution is denoted with $\mathbf{p}_i$. With these notations, we can specify the mutual information in Eq. (1) as follows:

$$\mathrm{I}(Y^*, y^* \mid \mathbf{D}^*) = \mathrm{H}(Y^* \mid \mathbf{D}^*) - \Big(\pi \cdot \mathrm{H}(Y^* \mid y^* = 1, \mathbf{D}^*) + (1-\pi) \cdot \mathrm{H}(Y^* \mid y^* = -1, \mathbf{D}^*)\Big) \tag{2}$$

$$\begin{aligned} &= -\sum_{i \in \{1,-1\}} \mathrm{p}(Y^* = i \mid \mathbf{D}^*) \log\left(\mathrm{p}(Y^* = i \mid \mathbf{D}^*)\right) \\ &\quad + \pi \cdot \sum_{j \in \{1,-1\}} p_1^j \log\left(p_1^j\right) + (1-\pi) \cdot \sum_{k \in \{1,-1\}} p_{-1}^k \log\left(p_{-1}^k\right) \quad . \end{aligned} \tag{3}$$

Probabilities $\mathrm{p}(Y^* = i \mid \mathbf{D}^*)$ for reclassification results can be written in terms of the conditional probabilities of Y^*:

$$\mathrm{p}(Y^* = i \mid \mathbf{D}^*) \quad = \quad \pi \cdot p_1^i + (1-\pi) \cdot p_{-1}^i \quad =: \quad m^i \quad . \tag{4}$$

Replacing the probabilities $\mathrm{p}(Y^* = i \mid \mathbf{D}^*)$ in Eq. (3) by the term given in Eq. (4), we get the following expressions for the conditional mutual information:

$$\begin{aligned} \mathrm{I}(Y^*, y^* \mid \mathbf{D}^*) &= -\sum_{i \in \{1,-1\}} \left(\pi \cdot p_1^i + (1-\pi) \cdot p_{-1}^i\right) \log\left(m^i\right) \\ &\quad + \pi \cdot \sum_{j \in \{1,-1\}} p_1^j \log\left(p_1^j\right) + (1-\pi) \cdot \sum_{k \in \{1,-1\}} p_{-1}^k \log\left(p_{-1}^k\right) \end{aligned} \tag{5}$$

$$= \pi \cdot \sum_{i \in \{1,-1\}} p_1^i \log\left(\frac{p_1^i}{m^i}\right) + (1-\pi) \cdot \sum_{j \in \{1,-1\}} p_{-1}^j \log\left(\frac{p_{-1}^j}{m^j}\right) \tag{6}$$

$$= \pi \cdot \mathrm{D}_{\mathrm{KL}}\left(\mathbf{p}_1 || \mathbf{m}\right) + (1-\pi) \cdot \mathrm{D}_{\mathrm{KL}}\left(\mathbf{p}_{-1} || \mathbf{m}\right) \quad , \tag{7}$$

where $\mathrm{D}_{\mathrm{KL}}(\cdot||\cdot)$ is the Kullback-Leibler (KL) divergence and $\mathbf{m} = \pi \cdot \mathbf{p}_1 + (1-\pi) \cdot \mathbf{p}_{-1}$ the mixture of the two conditional probability distributions $\mathbf{p}_1$ and $\mathbf{p}_{-1}$. From Eq. (7) we observe that the mutual information of Y^* and y^* is equal to the Jensen-Shannon (JS) divergence [8] $\mathrm{D}_{\mathrm{JS}}^{\pi}(\mathbf{p}_1||\mathbf{p}_{-1})$ of the probability distributions $\mathbf{p}_1$ and $\mathbf{p}_{-1}$. Therefore, this divergence measures the difference of the models obtained with label assumptions $y^* = -1$ and $y^* = 1$, which is an estimation of the impact of $\mathbf{x}_*$ on the model. Note that we are able to incorporate prior knowledge of an OCC task by controlling the parameter π. However, without any prior knowledge, one typically assumes a uniform prior $\pi = (1-\pi) = \frac{1}{2}$ leading to:

$$\mathrm{D}_{\mathrm{JS}}^{\frac{1}{2}}(\mathbf{p}_1||\mathbf{p}_{-1}) = \frac{1}{2}\mathrm{D}_{\mathrm{KL}}(\mathbf{p}_1||\mathbf{m}) + \frac{1}{2}\mathrm{D}_{\mathrm{KL}}(\mathbf{p}_{-1}||\mathbf{m}) \quad . \tag{8}$$

Furthermore, it is bounded by 0 and 1, because the mutual information is non-negative and bounded by the conditional entropy of a binary random variable. A mutual information of 0 is achieved if and only if the involved probability distributions are equal.

We propose using the negative JS divergence as an OCC score. For the computation, we only need the parameter π as well as conditional probability distributions $\mathbf{p}_1$, and $\mathbf{p}_{-1}$. Since the random variables are binary, we have discrete distributions and can use any suitable model that provides the conditional probabilities and offers regularizing model complexity. In this paper, we propose using posterior probabilities of a Gaussian process (Section 3). An advantage for our framework is that the GP regression model is influenced by every training sample, even possible outliers, which leads to the behaviour shown in Figure 2.

Note that applying the JS divergence is different from the approach of [3], where the authors propose using the asymmetric KL divergence based on probabilities estimated from mixtures of Gaussians in the input space. In contrast, we incorporate the GP framework to allow for more flexibility of our model by using kernels. Besides, our divergence-based scores are directly derived from an information theoretic framework justifying their utilisation.

3 Integrating Gaussian process posterior probabilities

In the previous section, we formulated our framework without specifying any probability distribution, neither prior probabilities π nor probability distributions $\mathbf{p}_1$ and $\mathbf{p}_{-1}$. In the following, we show how predictive probabilities estimated with GP classification can be integrated into the proposed framework. We first very briefly review GP regression for OCC as introduced by [6] and show afterwards how to estimate label probabilities.

3.1 GP regression for one-class classification

The Gaussian process framework is a well-known probabilistic methodology that is successfully used for tasks such as regression and classification [13]. In the case of GP regression, continuous outputs y_c are assumed to be generated according to $y_c(\mathbf{x}) = f(\mathbf{x}) + \varepsilon$, where f is a latent function and ε is a noise term. Following a Bayesian framework, output values of unknown samples $\mathbf{x}^*$ are predicted in a probabilistic fashion by marginalising over latent functions f. A key part of GP regression are the following assumptions:

1. Latent functions f are drawn from a Gaussian process prior with mean function $\mu(\cdot)$ and covariance function $\kappa(\cdot,\cdot)$.
2. The noise term is normally distributed: $\varepsilon \sim \mathcal{N}(0, \sigma_{\mathrm{n}}^2)$.

The last assumption allows for a tractable prediction. Since we have no a-priori knowledge about the underlying data in general, we assume a zero mean of the GP prior in the following derivations. Using these assumptions, the predictive output value y_c^* for a new sample $\mathbf{x}^*$ given the data $\mathbf{D}^* = (\mathbf{X}, \mathbf{y}, \mathbf{x}^*)$ is normally distributed as well:

$$y_c^* \mid \mathbf{D}^* \sim \mathcal{N}(\mu_*, \sigma_*^2) \quad , \tag{9}$$

where moments μ_* and σ_*^2 can be given in closed form expressions. For more insights into the GP framework, the interested reader is referred to [13].

The work of [6] shows how GP regression can be employed for one-class classification problems. The authors propose using both the predictive mean μ_* (GP-Mean) and negative variance $-\sigma_*^2$ (GP-Var) as one-class scores applied to training data with labels $\mathbf{y} = \mathbf{1}$:

$$\mu_* = \mathbf{k}_*^\mathsf{T} \left(\mathbf{K} + \sigma_\mathrm{n}^2 \mathbf{I}\right)^{-1} \mathbf{1} \quad \text{and} \tag{10}$$

$$-\sigma_*^2 = -\left(\mathbf{k}_{**} - \mathbf{k}_*^\mathsf{T} \left(\mathbf{K} + \sigma_\mathrm{n}^2 \mathbf{I}\right)^{-1} \mathbf{k}_* + \sigma_\mathrm{n}^2\right) \quad , \tag{11}$$

where the shortcuts $\mathbf{k}_{**} = \kappa(\mathbf{x}^*, \mathbf{x}^*)$, $\mathbf{k}_* = \kappa(\mathbf{X}, \mathbf{x}^*)$, and $\mathbf{K} = \kappa(\mathbf{X}, \mathbf{X})$ are used. They also utilise the predictive probability density value $\mathrm{p}\left(y_c^* = 1 \mid \mathbf{D}^*\right)$ (GP-Pred) as a combined score of mean and variance.

3.2 GP classification for divergence-based one-class classification

In this section, we show how to compute the conditional probabilities that are necessary to obtain the divergence measures defined in Section 2. Recalling Eq. (9) of the one-class GP, we first observe that the prior probabilities $\pi = \mathrm{p}(y^* = 1 \mid \mathbf{D}^*)$ can be calculated using the predictive mean and variance from Section 3.1. Due to the fact that y^* is a binary variable and y_c^* is continuous, we propose computing π via:

$$\pi = \mathrm{p}(y^* = 1 \mid \mathbf{D}^*) = \mathrm{p}\left(y_c^* > 0 \mid \mathbf{D}^*\right) = \frac{1}{2} - \frac{1}{2} \operatorname{erf}\left(\frac{-\mu_*}{\sqrt{2\sigma_*^2}}\right) \quad , \tag{12}$$

where $\operatorname{erf}(\cdot)$ is the error function and the parameters μ_* and σ_*^2 are obtained from one-class GP as given in Eq. (10) and (11). Since this probability leads to different scores compared to the predictive probability density value $\mathrm{p}\left(y_c^* = 1 \mid \mathbf{D}^*\right)$ presented in [6], we additionally propose using $\mathrm{p}\left(y_c^* > 0 \mid \mathbf{D}^*\right)$ (*GP-Prob*) of GP regression directly for OCC as well. Note that this approach is related to the probit model utilised in [6] but without applying Laplace approximation, which is necessary when also the discrete nature of the training labels is taken into account [11].

Beside the prior probabilities, we also need to compute probabilities of the conditional distributions $\mathbf{p}_1 = \mathrm{p}(Y^* \mid y^* = 1, \mathbf{D}^*)$ and $\mathbf{p}_{-1} = \mathrm{p}(Y^* \mid y^* = -1, \mathbf{D}^*)$. If we denote the number of training samples stored in $\mathbf{X}$ with N, the conditional probabilities will arise from a GP model learnt with $N+1$ training samples by treating the current test sample $\mathbf{x}^*$ and its assumed label y^* as training data as well. Let us have a closer look how we can compute the conditional probabilities.

For the distribution $\mathbf{p}_1$, we have an OCC scenario with $N+1$ training samples and calculate the moments of the normal distribution at position $\mathbf{x}^*$ in the sense of reclassification using Eq. (10) and (11). Having these moments, we are able to compute similar to Eq. (12):

$$\mathrm{p}(Y^* = 1 \mid y^* = 1, \mathbf{D}^*) = \mathrm{p}\left(y_c^* > 0 \mid y^* = 1, \mathbf{D}^*\right) \quad \text{and} \tag{13}$$

$$\mathrm{p}(Y^* = -1 \mid y^* = 1, \mathbf{D}^*) = 1 - \mathrm{p}\left(y_c^* > 0 \mid y^* = 1, \mathbf{D}^*\right) \quad . \tag{14}$$

The assumption $y^* = -1$ leads to a binary classification scenario and we compute probabilities of the distribution $\mathbf{p}_{-1}$ using the GP regression framework for binary classification. The predictive variance is independent of the labels and therefore remains the same as for the distribution $\mathbf{p}_1$. Thus, we compute the variance using Eq. (11) treating $\mathbf{x}^*$ as a training sample leading to an involved kernel matrix $\mathbf{K}$ of size $(N+1)\times(N+1)$ and a similarity vector $\mathbf{k}_*$ of size $N+1$. The mean value can be computed with Eq. (10) using the extensions of $\mathbf{K}$ and $\mathbf{k}^*$ as well as the vector $(\mathbf{1}^\mathsf{T};-1)^\mathsf{T}$ instead of $\mathbf{1}$. The probabilities $\mathrm{p}(Y^* = 1 \mid y^* = -1, \mathbf{D}^*)$ and $\mathrm{p}(Y^* = -1 \mid y^* = -1, \mathbf{D}^*)$ can be computed with mean and variance like those of the distribution $\mathbf{p}_1$ (Eq. (13) and (14)). The difference between the behaviour of samples stemming from the target class and outliers is visualised in Figure 2 using a solid line for the predictive mean and shaded areas for the predictive variance of the GP model.

Since the binary classification scenario is highly imbalanced, we utilise different noise levels for samples of the two classes in the GP model to overcome this drawback. However, the bias of imbalanced data could also be beneficial when we deal with an OCC scenario, since we want to have a strong influence of our target data samples. We therefore differentiate between the balanced and imbalanced JS divergence depending on whether we use the balancing strategy for computing the involved probabilities. For both approaches, we have to calculate the moments of normal distributions depending on the inverse of a matrix which itself depends on the current test sample $\mathbf{x}^*$. Fortunately, we do not need to compute the whole inverse of this matrix requiring $\mathcal{O}(N^3)$ operations each time. Instead, we update its Cholesky factor in an efficient way as explained in the next section.

3.3 Efficient Cholesky updates

In this section, we show that computing our divergence-based novelty scores can be done in $\mathcal{O}(N^2)$ by efficiently updating the Cholesky decomposition of the kernel matrix. From Eq. (10) and (11), we observe that we have to perform multiplications with the inverse of $(\mathbf{K}+\sigma_\mathrm{n}^2\,\mathbf{I})$ which is equal to solving the linear system:

$$(\mathbf{K}+\sigma_\mathrm{n}^2\,\mathbf{I})\,\mathbf{x} = \mathbf{z} \quad . \tag{15}$$

Since $(\mathbf{K}+\sigma_\mathrm{n}^2\,\mathbf{I})$ is symmetric and positive definite, we can calculate its Cholesky decomposition $(\mathbf{K}+\sigma_\mathrm{n}^2\,\mathbf{I}) = \mathbf{L}\,\mathbf{L}^\mathsf{T}$, where the Cholesky factor $\mathbf{L}$ is a lower triangular matrix. Instead of inverting the kernel matrix, we solve Eq. (15) via:

$$\mathbf{L}\,\hat{\mathbf{x}} = \mathbf{z} \quad \text{and} \quad \mathbf{L}^\mathsf{T}\,\mathbf{x} = \hat{\mathbf{x}} \tag{16}$$

using the Cholesky factor as well as forward- and back-substitutions. If we treat the test sample as an additional training sample following the approach proposed in the previous sections, we have to cope with the inverse of the matrix $\mathbf{K}^*$ defined by:

$$\mathbf{K}^* = \begin{pmatrix} \mathbf{K}+\sigma_\mathrm{n}^2\,\mathbf{I} & \mathbf{k}_* \\ \mathbf{k}_*^\mathsf{T} & \mathbf{k}_{**}+\sigma_\mathrm{n}^2 \end{pmatrix} \quad . \tag{17}$$

Denoting the Cholesky factor of $\mathbf{K}^*$ with $\mathbf{L}^*$, we can calculate $\mathbf{L}^*$ using $\mathbf{L}$ via [10]:

$$\mathbf{L}^* = \begin{pmatrix} \mathbf{L} & \mathbf{0} \\ \boldsymbol{\ell}_*^\mathsf{T} & \ell_{**} \end{pmatrix} \quad , \tag{18}$$

where we obtain $\boldsymbol{\ell}_*$ and ℓ_{**} from solving $\mathbf{L}\,\boldsymbol{\ell}_* = \mathbf{k}_*$ and $\ell_{**} = \sqrt{\mathbf{k}_{**}+\sigma_\mathrm{n}^2 - \boldsymbol{\ell}_*^\mathsf{T}\,\boldsymbol{\ell}_*}$. We can therefore compute $\mathbf{L}$ in the learning step as done for one-class GP and $\mathbf{L}^*$ in the test step

using Eq. (18). Note that the matrix inversion lemma [9] can also be applied here, however, the Cholesky decomposition is often regarded as being more numerically stable.

4 Experiments

In this section, we first explain how the experiments are conducted. The results on well-known machine learning datasets as well as on the challenging ImageNet dataset [2] used for image categorisation are presented afterwards. The performance of each method is measured with the area under the ROC curve (AUC).

4.1 Experimental setup

The experiments are performed to compare our OCC methods with other kernel-based techniques, in particular Parzen density estimation, SVDD, and one-class GP. Similarities between two feature representations are measured with the Gaussian kernel. Beside the hyperparameter of this kernel function (scale σ), we also have to determine the noise variance σ_n^2 of the GP models as well as the outlier ratio ν of SVDD [15]. To estimate optimal parameter values, random splitting of the datasets in training, validation and test sets is done. During optimisation, the parameters are varied as follows: the optimal kernel hyperparameter σ is estimated from $\{0.25, 0.5, 0.75, \ldots, 2.0\}$, whereas the method specific parameters σ_n^2 and ν are chosen from $\{0.025, 0.05, 0.075, \ldots, 0.2\}$.

As done in [15], we first perform experiments with two UCI datasets [4], namely Iris[1] and Sonar[2]. Since each of the three Iris classes contains 50 samples, we randomly divide them into 15 samples for training, 15 for parameter optimisation and 20 for testing. Recognition results for each target class are obtained by averaging over 20 random splits. The Sonar dataset contains two classes with 97 and 111 samples. For each class, we randomly pick 30 samples for training, 30 for parameter optimisation and the remaining samples for testing. Recognition results are again achieved by averaging over 20 random splits.

To perform large-scale experiments, we use the ImageNet database and choose the same 1,000 object categories as done for ILSVRC 2010[3]. The provided quantised local features[4] are used to calculate histogram representations following the bag-of-visual-words approach. We split the dataset into three subsets used for training, parameter optimisation, and testing. Each of the three subsets contains 50 images per category leading to 50,000 samples for each set. The experimental results are averaged over all 1,000 tasks.

4.2 Results on UCI machine learning problems

The evaluation of our experiments with the UCI datasets [4] is shown in Table 1. For the Iris dataset, we only list results of two classes, because all evaluated methods achieve 100% accuracy on the remaining third class. Considering the median AUC values obtained from both datasets, we observe that there is no method superior to the others in all four tasks, but our imbalanced JS divergence measure is among the best three approaches in every task. The results of the class *Sonar-Rocks* are clearly dominated by our methods, since our three

[1] http://archive.ics.uci.edu/ml/datasets/Iris
[2] http://archive.ics.uci.edu/ml/datasets/Connectionist+Bench+(Sonar,+Mines+vs.+Rocks)
[3] http://www.image-net.org/challenges/LSVRC/2010
[4] http://www.image-net.org/download-features

OCC method	Median AUC of target class *Iris-Versicolour*	*Iris-Virginica*	*Sonar-Rocks*	*Sonar-Mines*
GP-Prob	0.981	0.966	0.625	0.772
bal. JS div.	0.981	0.967	0.618	0.768
imbal. JS div.	0.981	0.968	0.624	0.773
Parzen [1]	0.973	0.960	0.602	0.771
SVDD [15]	0.986	0.971	0.609	0.761
GP-Mean [6]	0.983	0.974	0.613	0.756
GP-Var [6]	0.979	0.964	0.608	0.770
GP-Pred [6]	0.980	0.968	0.618	0.776

Table 1: Method comparison on the UCI datasets Iris and Sonar. Our proposed OCC methods (GP-Probability, balanced and imbalanced JS divergence) are highlighted with bold font and the best three results of each task are underlined.

proposed measures achieve the best three performances in this case. As a summary, we notice that our methods achieve state-of-the-art performance in OCC and are even able to outperform the most prominent techniques.

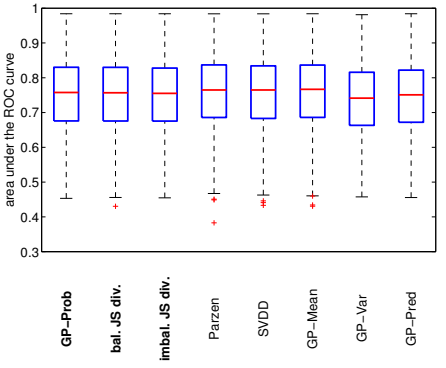
4.3 Results in large-scale image categorisation

Usually, OCC methods are tested on UCI machine learning datasets [15] or applications of limited size and complexity [6]. However, real-world novelty detection tasks such as in visual object recognition are often more difficult due to the high variability of the data. Therefore, we compare our methods also on ImageNet [2], a large-scale image categorisation dataset containing $1,000$ different classes and tens of thousands of images. The results of the experimental evaluation are visualised in Figure 3. As can be seen, our presented methods achieve performance comparable to state-of-the-art, even on this challenging large-scale dataset. This demonstrates that our proposed techniques are able to cope with large-scale scenarios. Interestingly, the performances of well-established methods also do not significantly differ (Wilcoxon rank sum test for Parzen, SVDD, and GP-Mean: $p > 0.8$).

5 Conclusions and future work

We presented a new approach for one-class classification, which is based on a novel theoretical framework combining concepts of information theory as well as Gaussian process classification. The main idea of our approach is to measure the impact of the label of a new test sample on the current classification model, which is a new way of considering one-class classification problems. Our methods, which allow for flexible novelty detection with arbitrary kernel functions, were evaluated on several machine learning as well as image categorisation tasks. We demonstrated that they achieve state-of-the-art performance comparable to well-known and already established techniques.

Future work will concentrate on incorporating other information sources to improve the novelty detection performance. For example, novelty detection for scene understanding and continuous learning could highly benefit from incorporating temporal as well as spatial object context. Another important aspect is the adaptation of one-class classification techniques to perform novelty detection for multiple known classes.



OCC method	Median AUC (Std. dev.)	
GP-Prob	0.758	(±0.103)
bal. JS div.	0.757	(±0.103)
imbal. JS div.	0.755	(±0.103)
Parzen [1]	0.765	(±0.105)
SVDD [15]	0.765	(±0.103)
GP-Mean [6]	0.767	(±0.103)
GP-Var [6]	0.741	(±0.104)
GP-Pred [6]	0.751	(±0.103)

Figure 3: Method comparison on the ImageNet dataset. Our proposed OCC methods (GP-Probability, balanced and imbalanced JS divergence) are highlighted with bold font.

References

[1] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006.

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.

[3] Maurizio Filippone and Guido Sanguinetti. Information theoretic novelty detection. *Pattern Recognition*, 43(3):805–814, 2010.

[4] Andrew Frank and Arthur Asuncion. UCI machine learning repository, 2010. URL http://archive.ics.uci.edu/ml.

[5] Ashish Kapoor, Kristen Grauman, Raquel Urtasun, and Trevor Darrell. Gaussian processes for object categorization. *International Journal of Computer Vision*, 88(2):169–188, 2010.

[6] Michael Kemmler, Erik Rodner, and Joachim Denzler. One-class classification with gaussian processes. In *ACCV*, pages 489–500, 2010.

[7] Neil D. Lawrence. Probabilistic non-linear principal component analysis with gaussian process latent variable models. *Journal of Machine Learning Research*, 6:1783–1816, 2005.

[8] Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1):145–151, 1991.

[9] D.J.C. MacKay. Introduction to gaussian processes. *NATO ASI Series F Computer and Systems Sciences*, 168:133–166, 1998.

[10] Duy Nguyen-Tuong, Matthias W. Seeger, and Jan Peters. Model learning with local gaussian process regression. *Advanced Robotics*, 23(15):2015–2034, 2009.

[11] Hannes Nickisch and Carl E. Rasmussen. Approximations for binary gaussian process classification. *Journal of Machine Learning Research*, 9:2035–2078, 2008.

[12] Hannes Nickisch and Carl E. Rasmussen. Gaussian mixture modeling with gaussian process latent variable models. In *Proceedings of the Annual Symposium of the German Association for Pattern Recognition (DAGM)*, pages 272–282, 2010.

[13] Carl E. Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[14] David M. J. Tax. *One-class classification*. PhD thesis, Delft University of Technology, June 2001.

[15] David M. J. Tax and Robert P. W. Duin. Support vector data description. *Machine Learning*, 54(1):45–66, 2004.