Subject Section

Adaptive One-Class Gaussian Processes Allow Accurate Prioritization of Oncology Drug Targets

Antonio de Falco [1], Zoltan Dezso [2], Francesco Ceccarelli [3], Luigi Cerulo [1,4], Angelo Ciaramella [5], and Michele Ceccarelli [1,6,*]

[1] BIOGEM Istituto di Ricerche Genetiche "G. Salvatore", Ariano Irpino, Italy
[2] ABBVIE Biotherapeutics, Redwood City (CA), USA
[3] Computer Laboratory, University of Cambridge, CB3 0FD, Cambridge, UK
[4] Department of Science and Technologies, University of Sannio, Benevento, Italy
[5] Department Science and Technology, University of Naples Parthenope, Naples, Italy
[6] Department of Electrical Engineering and Information Technology (DIETI), University of Naples "Federico II", 80128, Naples, Italy.

*To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

Abstract

Motivation: The cost of drug development has dramatically increased in the last decades, with the number new drugs approved per billion US dollars spent on R&D halving every year or less. The selection and prioritization of targets is one the the most influential decisions in drug discovery. Here we present a Gaussian Process model for the prioritization of drug targets cast as a problem of learning with only positive and unlabeled examples.

Results: Since the absence of negative samples does not allow standard methods for automatic selection of hyperparameters, we propose a novel approach for hyperparameter selection of the kernel in One Class Gaussian Processes. We compare our methods with state-of-the-art approaches on benchmark datasets and then show its application to druggability prediction of oncology drugs. Our score reaches an AUC 0.90 on a set of clinical trial targets starting from a small training set of 102 validated oncology targets. Our score recovers the majority of known drug targets and can be used to identify novel set of proteins as drug target candidates.

Availability: Source code implemented in Python is freely available for download at https://github.com/AntonioDeFalco/Adaptive-OCGP.

Contact: michele.ceccarelli@unina.it

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

The selection and prioritization of drug targets represents a central problem in drug discovery. Drug targets are proteins associated with a particular disease process and that could be addressed by a drug in order to obtain a specific therapeutic effect (Triggle and Taylor, 2006). Experimental approaches to target identification are typically expensive and labor intensive (Behan *et al.*, 2019; McFarland *et al.*, 2018). The whole process from discovery to approval of a drug can take 10-15 years and up to several billions of investments (Madhukar *et al.*, 2019). One of the bottlenecks is the identification and prioritization of suitable drug targets. On the other hand, the increasing amount of data, which allows the creation of large scale human genomics and proteomics datasets, have the potential to substantially reduce the work and resources needed. Machine learning approaches can exploit the shared features between approved targets to select and score unknown targets (Dezső and Ceccarelli, 2020; Isik *et al.*, 2015; Kim *et al.*, 2017; Bakheet and Doig, 2009). If we focus just on Oncology, there are actually less than 150 proteins that are targets of approved drugs. These proteins can be seen as seed positive examples whose properties can be used by a learning machine to score all other potential drug targets. This kind of problem is known in machine learning as One Class Classification (OCC) or Positive Unlabeled (PU)

problems (Elkan and Noto, 2008; Cerulo *et al.*, 2010) with the additional complication of the high unbalance between the positive set and the wide set on unlabeled samples (He and Garcia, 2009). We have previously shown that a combination of *bagging* and *easy ensemble* approaches (Dezső and Ceccarelli, 2020; He and Garcia, 2009) can be a viable solution which comes at the cost of the need of generating thousands of classifiers trained with samples from the unlabeled set. Here we show that the geometry of this small set of positive examples can be modeled using a class of non-parametric regressors and classifiers based on Gaussian Processes (GP) (Rasmussen and Williams, 2006). In particular One-Class Gaussian Processes (OCGP) have been shown to outperform other kernel-based classifiers for binary and multi-class categorization of images (Kemmler *et al.*, 2010; Kapoor *et al.*, 2007). Despite the availability of robust linear-algebra algorithm for GPs (Rasmussen and Williams, 2006), the training of OCGP has some additional open questions related to the appropriate selection of hyperparameters of the kernel covariance function. Indeed, the presence of only positive samples of the training datasets makes it impossible to automatically select hyperparameters in GPs based on maximizing marginal likelihood (Kemmler *et al.*, 2010).

Xiao *et al.*, 2015 recently addressed this problem. The idea is to classify the positive samples between "internal" (those in the center of the envelope containing the training set), and "edge" samples (those in the vicinity of the border of the envelope). The authors optimize the parameter by maximizing the difference between the regression function of the internal and edge samples. Other approaches use the distribution of distances among training data (Li *et al.*, 2015), or a different score for every positive sample based on the distances between that sample from all others (Kalantari *et al.*, 2016).

Here we propose a novel solution for an adaptive selection of the hyperparameters of the covariance function. We show that a local estimate based on the distance between a sample and its neighbors can outperform the method proposed in Xiao *et al.*, 2015 both on the UCI machine learning benchmark datasets (http://homepage.tudelft.nl/n9d04/occ/index.html) and for the specific problem of drug target prioritization. In order to evaluate and compare the accuracy of prediction of our method we use a set of additional 277 targets for drugs in oncology clinical trials not belonging to the training set.

The article is organized as follows, in the Section 2 we introduce the dataset for drug target prioritization, the GPs in OCC case and our proposed adaptive selection to address the problem of hyperparameter selection. In the Section 3 we benchmark our novel method with the one-class logistic regression (OCLR) and with hyperparameters selection of Xiao *et al.*, 2015, on UCI datasets and then on drug target prioritization problem. Finally, in section 4 we summarize the results of the experiments.

2 Methods

2.1 Protein Features

Here we focus only on cancer drug targets. We identify a set of approved cancer drugs based on the TTD database (Li *et al.*, 2017), which contained 2917 unique protein targets of which 345 were approved, 903 clinical trials and 1669 research targets. Only oncological targets approved and in the experimentation phase have been selected, obtaining a set of 102 approved drug targets and another 277 clinical trial targets (Supplementary Table 1). Finally, the dataset consists of all human proteins and each of them has 70 features obtained by combining the information in the Swiss-prot database (Bairoch, 1991), network centrality properties determined on the basis of protein-protein network information in the STRING database (Szklarczyk *et al.*, 2018) and computationally predicting the missing data as previously described (Dezső and Ceccarelli, 2020).

2.2 Gaussian Processes for OCC

Formally a *Gaussian Process* (GP) is defined as a collection of random variables, any finite number of which have a joint Gaussian distribution (Rasmussen and Williams, 2006). In order to specify a GP we only need to identify its mean and covariance functions. If the random variables represent the value of a latent function $f(\mathbf{x})$ at location $\mathbf{x}$, the mean function $m(\mathbf{x})$ and the covariance $k(\mathbf{x}, \mathbf{x}')$ of our GP are:

$$\begin{aligned} m(\mathbf{x}) &= \mathbb{E}[f(\mathbf{x})] \\ k(\mathbf{x}, \mathbf{x}') &= \mathbb{E}[(f(\mathbf{x}) - m(\mathbf{x}))(f(\mathbf{x}') - m(\mathbf{x}'))] \end{aligned} \tag{1}$$

and we write:

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \tag{2}$$

Usually, the mean function is assumed to be zero. A GP is a very effective way to model a prior over functions simply by specifying the covariance such as for example the squared exponential, which allows to sample from smooth functions. The covariance function $k(\cdot, \cdot)$ is also called the *kernel*. Given a set of, eventually noisy, training observations $\{(\mathbf{x}_i, f_i)|i = 1, \ldots, n\}$, and a set of *test* points $\{(\mathbf{x}_i^* f_i^*), |i = 1, \ldots, n'\}$, the joint distribution of the training and test output $(\mathbf{f}, \mathbf{f}^*) = (f_1, ..., f_n, f_1^*, ..., f_{n'}^*)$ is also Gaussian. GPs provide an elegant and efficient way to perform inference by incorporating the knowledge that the training data provides about the test data through *conditioning*:

$$\begin{aligned} \mathbf{f}^*|X, X^*, \mathbf{f} \sim \mathcal{N}(&k(X^*, X)k(X, X)^{-1}\mathbf{f}, \\ &k(X^*, X^*) - k(X^*, X)k(X, X)^{-1}k(X, X^*)) \end{aligned} \tag{3}$$

where $k(X^*, X)$ is the $n' \times n$ covariance matrix evaluated at all pairs of training and test points, analogously for the other matrices $k(X, X), k(X^*, X^*)$ and the $k(X, X^*)$. If the observations are affected by additive identically distributed Gaussian noise with variance σ_n, the $n \times n$ matrix $k(X, X)$ in equation (3) is replaced with $[k(X, X) + \sigma_n I]$ (Rasmussen and Williams, 2006). Other then regression, GPs can be also used for classification. In binary classification, the basic idea is to use the output of a GP regression model as a latent variable which is then fed into a non-linear *response function*, such as the logistic or probit, that compresses the output in the range $[0, 1]$. Consider the two-class problem with target variable $y \in \{0, 1\}$. If we define a GP over a latent variable $f(\mathbf{x})$ and then apply a *response function* $\gamma()$ which "squashes" its argument between $[0, 1]$, then we obtain a stochastic non-Gaussian process $\pi(\mathbf{x}) \stackrel{def}{=} p(y = 1|\mathbf{x}) = \gamma(f(\mathbf{x}))$. In the case of classification we do not observe the function f but rather the input $X = \{\mathbf{x}_i|i = 1, ..., n\}$ and the corresponding class labels $y_1, ..., y_n$, and therefore we are interested in the value of π over the test cases $\pi(\mathbf{x}^*)$. Inference, in the case of classification, is divided in two steps:

- first computing the distribution of the latent variable corresponding to a test case:

$$p(f^*|X, \mathbf{y}, \mathbf{x}^*) = \int p(f^*|X, \mathbf{x}^*, \mathbf{f}) p(\mathbf{f}|X, \mathbf{y}) \mathrm{d}\mathbf{f} \tag{4}$$

  here $p(\mathbf{f}|X, \mathbf{y})$ is the posterior over the latent variables.
- the prediction is then produced averaging the response function $\gamma()$ using the distribution (4)

$$\bar{\pi} \stackrel{def}{=} p(y^* = 1|X, \mathbf{y}, \mathbf{x}^*) = \int \gamma(f^*) p(f^*|X, \mathbf{y}, \mathbf{x}^*) \mathrm{d}f^*. \tag{5}$$

Since the posterior $p(\mathbf{f}|X, \mathbf{y}) \propto p(\mathbf{y}|\mathbf{f}) p(\mathbf{f}|X)$ and $p(\mathbf{y}|\mathbf{f})$ is non-Gaussian, the integral in (4) cannot be analytically treated and inference is performed by a Gaussian approximation of the posterior through *Laplace*

Approximation (Rasmussen and Williams, 2006) or using Expectation Propagation (Minka, 2001). The second one-dimensional integral (5) can be analytically computed in the case of probit regression or with sampling methods or analytical approximations if γ is the logistic sigmoid.

The use of GP for one class problem has been pioneered in Kemmler *et al.*, 2010. The basic idea is to impose zero mean on the GP prior and use the value of $\mathbf{f} = 1$ in equation (3) on the positive examples. This will give high probability to latent functions with values that gradually decrease for observations that are far from the positive examples. When used in combination with the choice of a smooth co-variance function, this approach results in an important subset of latent functions that can be used for OCC (Kemmler *et al.*, 2010). As shown in Figure 1 the predictive mean decreases for inputs far from the training data, while the predictive variance increases. The mean and variance, which are computed according to equation (3), both represent possible membership scores in the one class classification problem. The predictive mean divided by the standard deviation as a combined measure to describe the uncertainty of estimation has also been proposed in Kapoor *et al.*, 2010 as an estimation of the uncertainty. Therefore as in Kemmler *et al.*, 2010 we will use four possible scores for an unknown observation x^*:

- *Mean:* $\mu_* = k(x^*, X)k(X, X)^{-1}\mathbf{1}$
- *Neg. Variance:* $-\sigma_*^2 = k(x^*, X)k(X, X)^{-1}k(X, x^*) - k(x^*, x^*)$
- *Probability:* equation (5)
- *Heuristics:* $\mu_* \cdot \sigma_*^{-1}$

The kernel is the main component in GPs, as it represents some form of distance or similarity between data points and determines the characteristics of the function to predict. Here we use the Squared Exponential (SE) kernel, which is the most used in GPs and thus defined:

$$k_{\mathrm{SE}}\left(x, x'\right) = \exp\left(-\frac{(x - x')^2}{2\ell^2}\right) \tag{6}$$

It is widely used due to its properties, infinitely differentiable and invariant in translation and rotation in both signal and frequency domains. It also has only a hyperparameter the lenght-scale (ℓ) that determines the length of the "oscillations" in the function, with small value the function can change quickly, and conversely with large values.

2.3 Hyperparameter Selection

As shown in Figure 1, the hyperparameters significantly affect the performance of the GPs, and in particular for OCC problems, the absence of negative samples in the training dataset does not allow automatic selection of hyperparameters through maximization of marginal likelihood.

Xiao *et al.*, 2015 propose an original solutions to this problem, based on the distinction of the internal samples and the edge samples of the positive class. The internal samples are assumed to be the most representative samples, instead the edge samples that are located at the extremes of the region are considered the samples closest to the possible negative regions. Consequently the predictions of GPs for the internal samples should be more certain, i.e. the predictive mean should be higher and the predictive variance lower, conversely for the edge samples. The authors select the optimal parameter by maximizing the Kullback-Leibler divergence between the predictions distribution these two set of samples.

Li *et al.*, 2015 propose another solution to determine the hyperparameters, based simply on distribution of distances among training data. The possible hyperparameters vary between half the average of distances to nine times the average of distances, and get better performance when the value is between three and seven times the average of the distances.

Kalantari *et al.*, 2016 instead propose two variants of one class GPs, the first is OCGP-thrifty which does not set all training target values to

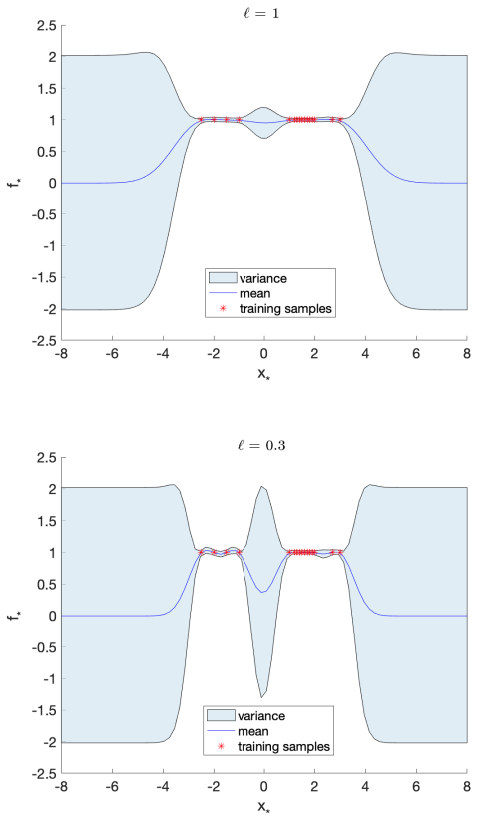

Fig. 1: OCGP regression 1-D using SE kernel. The predictive distribution is visualized, mean (blue line) and variances (light blue area), training points are marked as red asterisks. In the upper panel $\ell = 1.0$ is used, in the bottom panel $\ell = 0.3$.

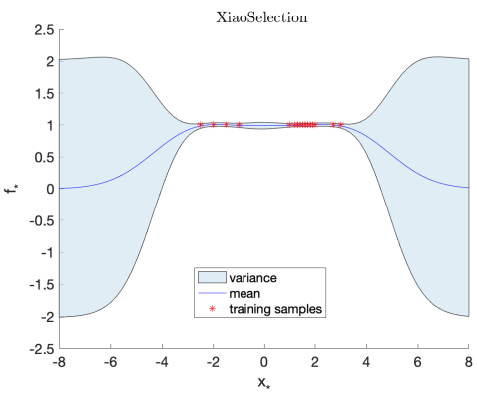

Fig. 2: OCGP regression 1-D, using SE kernel and an implementation of Xiao *et al.*, 2015 hyperparameter selection method.

1, but is based on the similarity of the training samples with the positive class. Specifically, the target value for a training sample is the average of the squared distances of that sample from all others. The second variant is OCGP-greedy which assumes that the information from other classes is available, and use it to train a one class model. The target training values are set as in OCGP-thrifty, also for samples of other classes. In the training phase, to find the hyperparameters, is built a regression model using all the training samples. In the test phase, to calculate the predictions, only the

samples of the positive class are used. The authors show that OCGP-greedy usually obtain better results, but it cannot be applied if only samples of the positive class are labeled as in our case.

2.4 Adaptive Hyperparameter

Here we do not set a single fixed value for the lenght-scale hyperparameter of the covariance function, and adapt this value for each training sample based on local density. This adaptive hyperparameter depends on the local distribution of the training data, the basic idea is to give more weight to the training samples belonging to dense areas, which represent the examples sharing common features of the positive class, and can be considered the most representative samples. On the other hand, we give less weight to training data lying in sparse areas, which could be less representative or outliers.

Given a sample x_i, let $\{\mathcal{N}_i^m\}_{m=1}^N$ the set of its first N neighbors ordered according to their distance from x_i. Then the value ℓ_i of each sample is set to the euclidean distance of i-th sample from its p-th nearest neighbor.

$$\ell_i = d(x_i, \mathcal{N}_i^p) \tag{7}$$

Therefore the *Adaptive Kernel* is larger in sparse areas and smaller in dense areas and is defined as:

$$k\left(x_{\mathrm{i}}, x_{\mathrm{j}}\right) = \exp\left(-\frac{\left(x_{\mathrm{i}} - x_{\mathrm{j}}\right)^2}{2\ell_{\mathrm{i}}^2}\right) \tag{8}$$

Since using equation (8) we have $k(x_i, x_j) \neq k(x_j, x_i)$, then we symmetrize the covariance matrix using $(K + K^T)/2$ as covariance.

Figure 3 shows an example in 1-D OCC setting of GP regression using zero-mean and our Adaptive Kernel with $p = 2$. The proposed solution allows to distinguish dense areas from areas with few samples, compared to the case where the hyperparameter is a constant value as shown in the Figure 1 or using the method of Xiao *et al.*, 2015 as shown in the Figure 2. Using the Adaptive Kernel the predictive mean and the predictive variance tend to adapt better to the general trend of the training set. We selectively obtain high scores for test input near training samples belonging to dense areas of the input space, which are theoretically the most representative positive samples.

The proposed Adaptive Kernel simply requires a search of the p-nearest neighbors of the training samples. Considering that a conventional p-nearest neighbors algorithm has $O(npd)$ complexity or $O(nd + pn)$ complexity pre-calculating and storing distances, it represents a computationally much more efficient solution, compared to the method of Xiao *et al.*, 2015 based on the edge-internal samples that has $O\left(n^3\right)$ complexity since it involve the computation of series of GPs.

We also explore another approach to automatically determine an adaptive hyperparameter: *Scaled Kernel*. This method has been successfully used in Similarity Network Fusion (SNF) (Wang *et al.*, 2014). In this case, hyperparameter selection combines the distance between samples and the average distance from the neighbors:

$$k\left(x_{\mathrm{i}}, x_{\mathrm{j}}\right) = \exp\left(-\frac{(x_i - x_j)^2}{\nu\varepsilon_{i,j}}\right) \tag{9}$$

$$\varepsilon_{i,j} = \frac{\mathrm{mean}\left(d\left(x_i, \mathcal{N}_i\right)\right) + \mathrm{mean}\left(d\left(x_j, \mathcal{N}_j\right)\right) + d\left(x_i, x_j\right)}{3} \tag{10}$$

In the Scaled Kernel equation (9) the parameter $\varepsilon_{i,j}$ combines the euclidean distance of the samples with the average distance of samples from the respective N nearest neighbors. Where $d(x_i, x_j)$ is the Euclidean distance, ν is a parameter that can be empirically set, and is usually set in the range $[0.3, 0.8]$, N represents the number of neighbors considered in the calculation of the average.

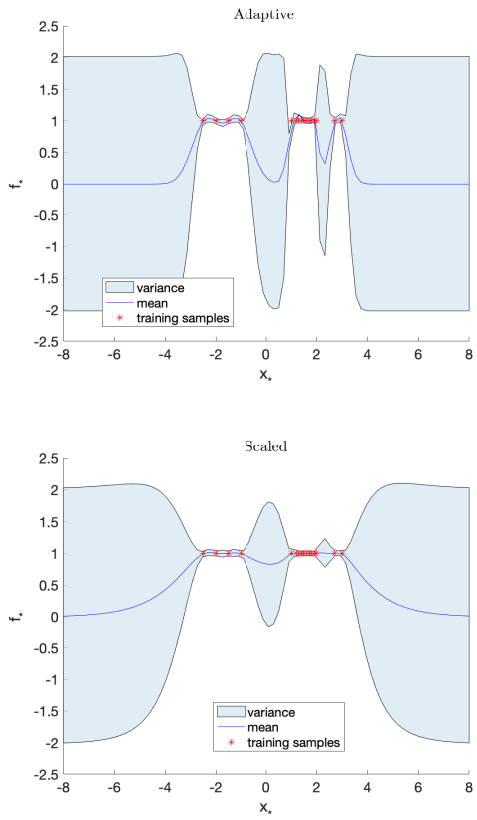

Fig. 3: OCGP regression 1-D, in the upper panel is used the proposed Adaptive Kernel (8) ($p = 2$), in the bottom panel the Scaled Kernel (9) ($N = 5$).

Figure 3 also shows an example of the Scaled Kernel with $N = 5$ in the mono-dimensional space, that like the Adaptive Kernel allows a better distinction of the dense areas.

3 Results

In this section, before reporting the application of the adaptive OCGP to the problem of drug target prioritization, we want to benchmark the proposed method with the method for hyperparameters selection for OCGP proposed in Xiao *et al.*, 2015 and also with the other one class classifiers such as support vector data description (SVDD) (Tax and Duin, 2004), one class SVM (OCSVM) (Schölkopf *et al.*, 2001) and one-class logistic regression (OCLR) (Sokolov *et al.*, 2016).

3.1 UCI Datasets

For experiments performed on nine UCI datasets (Table 1) we set $p = 2$ in (8), while $\nu = 0.8$ and $N = 5$ were used in (9).

For each dataset, we consider the class with the highest number of samples as the positive class, then 80% of the positive samples are randomly chosen to build the training set while the remaining 20% of positive samples and all negative samples constitute the test set. 20 iterations of subdivision of the train and test sets are performed. The calculation of hyperparameter lenght-scale ℓ is performed only after normalizing data with Z-score normalization. We report in Table 2 the average results across all iterations using both predictive mean and

Table 1. UCI datasets

dataset	features	pos	neg
Abalone	10	2770	1407
Balance	4	288	337
Biomed	5	127	67
Heart	13	164	139
Hepatitis	19	123	32
Housing	13	458	48
Ionosphere	34	225	126
Vehicle	18	647	199
Waveform	21	600	300

Table 2. Benchmark on UCI datasets (AUC scores).

	Xiao *et al.*		Adaptive(8)		Scaled(9)		OCLR	OCSVM	SVDD
	μ_*	$-\sigma_*^2$	μ_*	$-\sigma_*^2$	μ_*	$-\sigma_*^2$		rbf	poly3
Abal.	0.7894	0.7897	0.7745	0.7428	0.7742	0.7092	**0.8760**	0.6471	0.8608
Bala.	0.8366	0.8735	0.9468	**0.9682**	0.8657	0.9402	0.5599	0.8266	0.7198
Biom.	0.8998	0.9036	0.9028	0.8960	0.9073	**0.9117**	0.9050	0.8129	0.8570
Hear.	0.8339	0.8379	0.8093	0.7925	**0.8408**	0.8135	0.5379	0.6880	0.7918
Hepa.	0.8378	**0.8379**	0.8006	0.7794	0.8242	0.7963	0.5829	0.7257	0.8055
Hous.	0.7917	0.7874	0.8677	**0.8680**	0.8107	0.8492	0.6742	0.8217	0.8374
Iono.	0.9265	0.9504	0.9550	0.9649	0.9697	**0.9712**	0.8107	0.9115	0.9341
Vehi.	0.5183	0.5714	0.7965	**0.8656**	0.6855	0.8187	0.7908	0.5601	0.5696
Wave.	0.7497	0.8004	0.7808	0.8167	0.8024	0.7998	**0.8348**	0.6160	0.5088
Aver.	0.7982	0.8169	0.8482	**0.8549**	0.8312	0.8455	0.7299	0.7344	0.7650

negative variance as scores. The results show that the proposed adaptive hyperparameter for both Adaptive Kernel (8) and Scaled Kernel(9) produce a significant improvement in performance when compared to the selection of the hyperparameter based on the internal and edge samples by Xiao *et al.*, 2015, in particular the Adaptive Kernel (8) attains the best result on the average of all datasets, with an increase from 4 to 5 percentage for the two scores mean and negative predictive variance.

Table 2 also shows the results obtained using support vector data description (SVDD) (Tax and Duin, 2004), one class SVM (OCSVM) (Schölkopf *et al.*, 2001) and one-class logistic regression (OCLR) (Sokolov *et al.*, 2016). For OCSVM and OCSVM, since stationary kernels such as the rbf kernel produce the same results (Schölkopf *et al.*, 2001), we use rbf kernel for OCSVM and polynomial kernel of degree 3 for SVDD. The results confirm that Gaussian Processes are particularly suited for one class problems, with respect to other approaches as also reported in previous works Kemmler *et al.*, 2010.

Since the proposed adaptive kernels depend on some parameters such as p for the Adaptive Kernel and N for the Scaled Kernel, we want to analyze how the performance vary with the choice of these parameters. The results reported below show the AUC measurement on the predictive mean obtained from the average of the 20 random splits of each dataset, as function of the parameters. In the case of the Scaled Kernel (9), whose results are shown in Figure 4, the AUC is almost constant for all of datasets, demonstrating that this kernel is very little affected by variation of the parameter N. The Adaptive Kernel (8), reported in Figure 5, shows instead a slightly greater variations of performance for some datasets as a function of the p parameter.

3.2 Drug Target

Our dataset for the prioritization of Oncology Drug Targets consists of 20403 proteins, of which 102 are validated oncology targets, used for training, and 277 targets of clinical trial drugs. These last 277 proteins are used as validation set in our experiments. We extracted 70 protein features related to properties derived from the sequence, protein functions and network properties derived from protein-protein interaction network as previously reported (Dezső and Ceccarelli, 2020). The protein features in the dataset included continuous and categorical features, the latter are encoded with one-hot encoding and with frequency encoding. Some pre-processing steps are performed on the dataset, the features with a heavy-tailed distribution are log-transformed, and all features are scaled between $[0, 1]$ by min-max normalization. Furthermore, principal component analysis (PCA) is used to obtain the first principal components that allow 80% of the data variance to be retained.

First, we compare OCGPs with other OCCs. Table 3 reports the AUC obtained by the considered models and confirms that OCGPs outperform other classifiers.

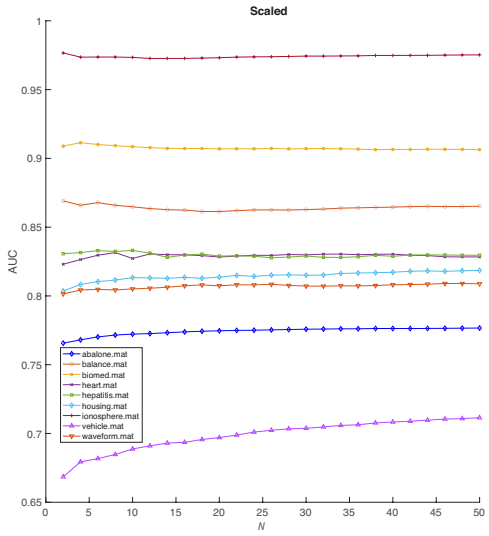

Fig. 4: AUC scores (μ_*) on UCI datasets using the Scaled Kernel (9) with different values for the N parameter.

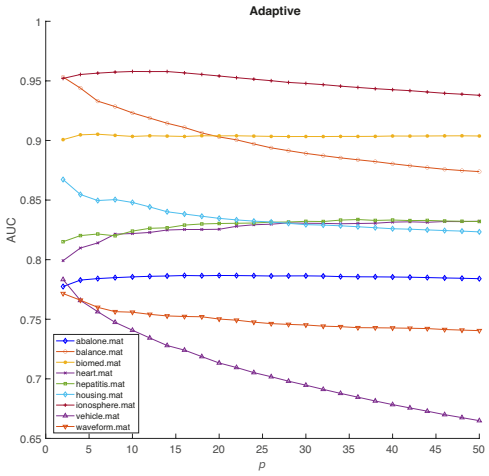

Fig. 5: AUC scores (μ_*) on UCI datasets using the Adaptive Kernel (8) with different values for the p parameter.

Table 3. Benchmark of one class classifiers on Drug Target dataset

OC Classifiers	kernel	AUC
OCLR		0.6120
OCSVM	rbf	0.6958
SVDD	poly	0.8253
OCGP ($\ell = 0.3$)	rbf	**0.8388**

Table 4. AUC scores (μ_*) on Drug Target Dataset

preprocessing	Adapt. (8)	Xiao *et al.*	Scaled (9)
scale	0.8613	**0.8680**	0.8555
scale+logtrasf.	**0.8878**	0.8610	0.8633
scale+logtrasf.+PCA	**0.8928**	0.8781	0.8759

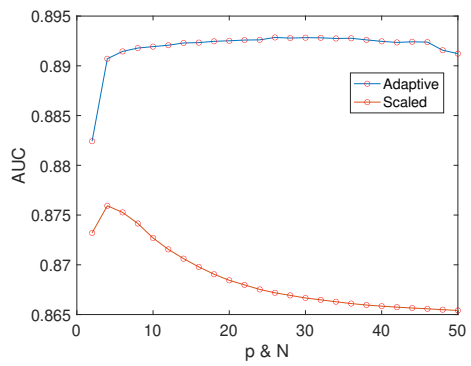

Fig. 6: AUC scores (μ_*) on Drug Target Dataset using Adaptive Kernel and Scaled Kernel with different values for the p and N parameters.

Then we show how the adaptive kernel can improve the classification accuracy. In what follows, we used $p = 30$ in equation(8), while $\nu = 0.8$ and $N = 4$ are used in equation (9).

In order to evaluate how the preprocessing influences the accuracy, Table 4 shows the results, in terms of the AUC measure on the predictive mean, obtained by adding individually the pre-processing steps described above. The results show that pre-processing lead to a significant improvement in results and the proposed Adaptive Kernel (8) attains better performance than the others.

We evaluate whether feature selection can possibly improve the results. We used Sequential Forward Selection (SFS) (Whitney, 1971), a sequential search algorithm in which features are added sequentially to an empty set of candidates, until the inclusion of additional features does not allow any improvement of the adopted criterion, in our case the criterion to improve is AUC measurement on the predictive mean.

The Sequential forward selection (SFS) selects 37 features and results in a significant improvement of the performance as shown in table 5 with Adaptive Kernel using the predictive mean as score. Interestingly, the features selected by the algorithm (Supplementary Table 1) include network centrality measures (betweenness, degree page-rank, closeness) as well biological process and others. Indeed, it is expected because the interaction between drugs and their targets activates signaling cascades through Protein-Protein Interaction networks causing downstream perturbations in the cell's transcriptome. A (PPI) network thus models the cascade of relationships between targets and proteins by using physical contacts, genetic interactions, and functional relationships.

Table 5. Benchmark of hyperparameter selection methods with SFS feature selection. (AUC on the four possible scores).

Hyperparameter Selection	μ_*	$-\sigma_*^2$	Eq. (5)	$\mu_* \cdot \sigma_*^{-1}$
Xiao	0.8781	0.8705	0.8783	0.8773
Xiao + SFS	0.8881	**0.8717**	0.8881	0.8861
Adaptive	0.8883	0.8667	0.8878	0.8860
Adaptive + SFS	**0.9008**	0.8677	**0.9002**	**0.8981**
Scaled	0.8759	0.8500	0.8765	0.8755
Scaled + SFS	0.8911	0.8569	0.8907	0.8899

Table 6. Hyperparameters selected by the methods

Hyperparameter Selection	min(ℓ)	max(ℓ)
Xiao	10.3621	
Xiao + SFS	6.7363	
Adaptive	4.1693	6.0390
Adaptive + SFS	3.9048	5.2675

The Adaptive Kernel outperform other methods on this dataset, but the scores obtained for test inputs differed for very low values. This is due to the hyperparameters computed before preprocessing that have high values which consequently results in kernel values close to 0 after division with ℓ. For this reason to obtain a better dynamics, which guarantees a better interpretability of results, we can apply logarithm or square root to transform the hyperparameters computed before the preprocessing. Choice that guarantees performances comparable to the previous results, as shown in the table 5 where we log transform the hyperparameters for the Adaptive Kernel.

Figure 7 shows the comparison of the prediction of scores for the approved targets, clinical targets and all other proteins. As expected, the 102 approved targets of our training set had the highest score with a median of 0.92. Instead for the test set the independent set of 277 cancer clinical targets was characterized by a high median score of 0.77 unlike the rest of the proteins which had a median score of 0.43 in the unlabeled set. Although the majority of these proteins have a lower score predicted by our model, this set contains outliers with a high score that can be considered interesting potential drug targets such as for example the 171 outliers, with scores greater than 0.91 in the boxplot of unlabeled proteins represented in red in Figure 7.

Some of these outliers are the subject of recent studies indicating their use as a target in oncological diseases. Among these in particular to be noted the proteins shown in the table 7: IL7R is considered in Cramer *et al.*, 2016 as a potential target of further therapy for leukemia patients, since the targeting of IL-7Rα signaling pathways has the potential to reduce cell proliferation and survival. JAG1 and DLL4 are most important ligands of Notch signaling, which has key role in development and progression of cancer, and represents an important therapeutic target, e.g. in several studies the blocking of their signaling in tumors has shown interruption of angiogenesis and inhibition of tumor growth (Oon *et al.*, 2017; Kangsamaksin *et al.*, 2015). PDGF and/or PDGF receptors are overexpressed or mutated in different tumors then their targeting can be beneficial in tumor treatment (Heldin, 2013; Papadopoulos and Lennartsson, 2017), e.g. targeting PDGFRA with crenolanib has shown significantly prevented tumor growth in inflammatory breast cancer (IBC) (Joglekar-Javadekar *et al.*, 2017) . Moreover Epiregulin (EREG) is identified as a possible target in lung cancer (Bauer *et al.*, 2017), particularly for Non-small-cell lung carcinoma (NSCLC) (Sunaga and Kaira, 2015). Adiponectin (ADIPOQ) is considered a potential target to

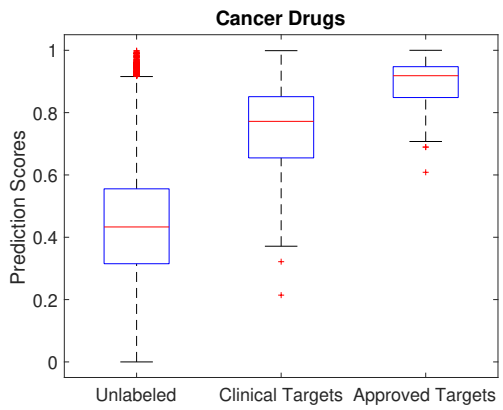

Fig. 7: Distribution of predictions scores among the training set (approved targets), validation set (clinical trial) and the rest of the proteins. Median score: Unlabeled 0.4331, Clinical Trial 0.77196, Approved Targets: 0.9184

Table 7. Possible drug targets among the outliers

Gene	Score
IL7R	0.99199
JAG1	0.99122
PDGFA	0.98761
EREG	0.98334
ADIPOQ	0.98224
FGF10	0.98015
DLL4	0.97909
FZD2	0.97304

many human disorders, including in particular prostate cancer (Karnati *et al.*, 2017; Hu *et al.*, 2019). FGF10 is considered in several studies a possible target in particular of Pancreatic ductal adenocarcinoma (PDAC) (Clayton and Grose, 2018; Ndlovu *et al.*, 2018). FZD2 is correlated with different cancers as shown in several studies, e.g. Huang *et al.*, 2019 confirms its oncogenic role in tongue cancer, and that it can be taken into account as a therapeutic target.

4 Discussion

Drug discovery is becoming more and more expensive over time despite improvements in technology. Estimates report that the number of new drugs approved per billion US dollars spent on R&D has halved roughly every 9 years since 1950 (Scannell *et al.*, 2012). The choice of appropriate therapeutic targets is one of the crucial steps in the drug discovery. Machine learning approaches can exploit available high-quality and abundant data to improve decision making in all stages of drug discovery in order to speed up the process and reduce failure rates in drug development. (Vamathevan *et al.*, 2019). Here we presented a Machine Learning approach to prioritize proteins according to their similarity to approved drug targets. The main characteristic of our approach is the fact that it is completely unbiased.

We use a large collection of protein features and let the learning method score the features of approved targets. Since we are interested in the extending this score to other proteins, our machine learning problem turned out to belong to the class on positive only problems that we approach using One Class Gaussian Processes. We also proposed a method for the selection of the lenght-scale hyperparameter of the radial basis function kernel of the Gaussian Process. The basic idea is the use of a different hyperparameter for each training sample, creating an Adaptive Kernel that varies depending on whether the training sample belongs to a sparse or dense area. The main aim is to give more importance to samples of dense areas, considered the most representative samples of the positive class. The validity of the proposed solution is shown in the results on the UCI benchmark datasets, confirming that the proposed method outperform the current state of the art based on edge-internal samples.

The development of a machine learning model based on OCGP combined with the use of the our Adaptive Kernels for the hyperparameter selection, allows to define a druggability score for each protein with high performance (AUC of 0.90) on targets in clinical trials. Furthermore several proteins outside the training set and validation set have a very high predicted score and can be consideres as further interesting potential candidates. The results obtained confirm the effectiveness of GPs in the one class classification problems, and that they can be be improved with a correct selection of the hyperparameters. The use of GP allows to obtain better results than and ensemble of Random Forest on the same set of features (Dezső and Ceccarelli, 2020). We have also show that our approach compares favorably with one class logistic regression (Sokolov *et al.*, 2016).

Funding

The research leading to these results has received funding from AIRC under IG 2018 - ID. 21846 project – P.I. Michele Ceccarelli and from Italian Ministry of Research Grant PRIN 2017XJ38A4_004.

Conflict of Interest: none declared.

Availability of Data and Materials

The matrix of features for each protein is available at: https://bit.ly/3iLgZTa. Source code implemented in Python is freely available for download at https://github.com/AntonioDeFalco/Adaptive-OCGP.

References

Bairoch, B. (1991). The swiss-prot protein sequence data bank. *Nucleic Acids Res*.

Bakheet, T. M. and Doig, A. J. (2009). Properties and identification of human protein drug targets. *Bioinformatics*, **25**(4), 451–457.

Bauer, A. K. *et al.* (2017). Epiregulin is required for lung tumor promotion in a murine two-stage carcinogenesis model. *Molecular Carcinogenesis*, **56**(1), 94–105.

Behan, F. M. *et al.* (2019). Prioritization of cancer therapeutic targets using crispr–cas9 screens. *Nature*, **568**(7753), 511–516.

Cerulo, L. *et al.* (2010). Learning gene regulatory networks from only positive and unlabeled data. *BMC bioinformatics*, **11**(1), 1–16.

Clayton, N. and Grose, R. (2018). Emerging roles of fibroblast growth factor 10 in cancer. *Frontiers in Genetics*, **9**.

Cramer, S. *et al.* (2016). Therapeutic targeting of il-7r signaling pathways in all treatment. *Blood*, **128**.

Dezső, Z. and Ceccarelli, M. (2020). Machine learning prediction of oncology drug targets based on protein and network properties. *BMC Bioinformatics*, **21**.

Elkan, C. and Noto, K. (2008). Learning classifiers from only positive and unlabeled data. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 213–220.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on knowledge and data engineering*, **21**(9), 1263–1284.

Heldin, C.-H. (2013). Targeting the pdgf signaling pathway in tumor treatment. *Cell communication and signaling : CCS*, **11**, 97.

Hu, X. *et al.* (2019). Role of adiponectin in prostate cancer. *International braz j urol : official journal of the Brazilian Society of Urology*, **45**, 220–228.

Huang, L. *et al.* (2019). Fzd2 regulates cell proliferation and invasion in tongue squamous cell carcinoma. *International Journal of Biological Sciences*, **15**, 2330–2339.

Isik, Z. *et al.* (2015). Drug target prioritization by perturbed gene expression and network information. *Scientific reports*, **5**, 17417.

Joglekar-Javadekar, M. *et al.* (2017). Characterization and targeting of platelet-derived growth factor receptor alpha (pdgfra) in inflammatory breast cancer (ibc). *Neoplasia*, **19**, 564–573.

Kalantari, L. *et al.* (2016). One-class gaussian process for possibilistic classification using imaging spectroscopy. *IEEE Geoscience and Remote Sensing Letters*, **13**, 1–5.

Kangsamaksin, T. *et al.* (2015). Notch decoys that selectively block dll/notch or jag/notch disrupt angiogenesis by unique mechanisms to inhibit tumor growth. *Cancer Discovery*, **5**(2), 182–197.

Kapoor, A. *et al.* (2007). Active learning with gaussian processes for object categorization. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8. IEEE.

Kapoor, A. *et al.* (2010). Gaussian processes for object categorization. *International Journal of Computer Vision*, **88**(2), 169–188.

Karnati, H. K. *et al.* (2017). Adiponectin as a potential therapeutic target for prostate cancer. *Current pharmaceutical design*, **23**(28), 4170—4179.

Kemmler, M. *et al.* (2010). One-class classification with gaussian processes. pages 489–500.

Kim, B. *et al.* (2017). In silico re-identification of properties of drug target proteins. *BMC bioinformatics*, **18**(7), 248.

Li, N. *et al.* (2015). Anomaly detection in video surveillance via gaussian process. *International Journal of Pattern Recognition and Artificial Intelligence*, **29**, 150426191333005.

Li, Y. *et al.* (2017). Therapeutic target database update 2018: Enriched resource for facilitating bench-to-clinic research of targeted therapeutics. *Nucleic acids research*, **46**.

Madhukar, N. S. *et al.* (2019). A bayesian machine learning approach for drug target identification using diverse data types. *Nature communications*, **10**(1), 1–14.

McFarland, J. M. *et al.* (2018). Improved estimation of cancer dependencies from large-scale rnai screens using model-based normalization and data integration. *Nature communications*, **9**(1), 1–13.

Minka, T. P. (2001). *A family of algorithms for approximate Bayesian inference*. Ph.D. thesis, Massachusetts Institute of Technology.

Ndlovu, R. *et al.* (2018). Fibroblast growth factor 10 in pancreas development and pancreatic cancer. *Frontiers in Genetics*, **9**, 482.

Oon, C. E. *et al.* (2017). Role of delta-like 4 in jagged1-induced tumour angiogenesis and tumour growth. *Oncotarget*, **8**, 40115 – 40131.

Papadopoulos, N. and Lennartsson, J. (2017). The pdgf/pdgfr pathway as a drug target. *Molecular aspects of medicine*, **62**.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.

Scannell, J. W. *et al.* (2012). Diagnosing the decline in pharmaceutical r&d efficiency. *Nature reviews Drug discovery*, **11**(3), 191.

Schölkopf, B. *et al.* (2001). Estimating the support of a high-dimensional distribution. *Neural Comput.*, **13**(7), 1443–1471.

Schölkopf, B. *et al.* (2001). Estimating support of a high-dimensional distribution. *Neural Computation*, **13**, 1443–1471.

Sokolov, A. *et al.* (2016). One-class detection of cell states in tumor subtypes. *Pacific Symposium on Biocomputing. Pacific Symposium on Biocomputing*, **21**, 405–16.

Sunaga, N. and Kaira, K. (2015). Epiregulin as a therapeutic target in non-small- cell lung cancer. *Lung Cancer: Targets and Therapy*, **6**, 91–98.

Szklarczyk, D. *et al.* (2018). String v11: protein-protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic acids research*, **47**.

Tax, D. M. J. and Duin, R. P. W. (2004). Support vector data description. *Mach. Learn.*, **54**(1), 45–66.

Triggle, D. J. and Taylor, J. B. (2006). *Comprehensive Medicinal Chemistry II*, volume 8. Elsevier.

Vamathevan, J. *et al.* (2019). Applications of machine learning in drug discovery and development. *Nature Reviews Drug Discovery*, **18**(6), 463–477.

Wang, B. *et al.* (2014). Similarity network fusion for aggregating data types on a genomic scale. *Nature methods*, **11**.

Whitney, A. W. (1971). A direct method of nonparametric measurement selection. *IEEE Transactions on Computers*, **C-20**(9), 1100–1103.

Xiao, Y. *et al.* (2015). Hyperparameter selection for gaussian process one-class classification. *IEEE Transactions on Neural Networks and Learning Systems*, **26**(9), 2182–2187.