ESANN 2018 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence
and Machine Learning. Bruges (Belgium), 25-27 April 2018, i6doc.com publ., ISBN 978-287587047-6.
Available from http://www.i6doc.com/en/.

Learning with a Fisher surrogate loss in a small data regime

Mousaab Djerrab, Alexandre Garcia and Florence d'Alché-Buc

LTCI, Télécom ParisTech, Université Paris-Saclay, 75013, Paris, France

Abstract. We introduce a novel framework, Output Fisher Embedding Regression (OFER), that uses a Fisher vector representation of output data and provides prediction by solving an appropriate pre-image problem. OFER takes advantage of the implicit structure of the marginal probability distribution of the output to improve performance in prediction. Although the proposed approach is general and versatile, we put a stress on the Gaussian mixture model for modelling the output data and design a closed-form solution for the corresponding pre-image problem. Numerical results on a drug activity prediction task and a semantic multi-class classification show the relevance of the approach in small data regime.

1 Introduction

In supervised learning, much attention has been paid on designing appropriate input representation by various techniques among which feature engineering methods, kernel approaches [7, 9] and representation learning [2]. It is well known that appropriate input features are key to the success of a learning algorithm. Now, regarding the outputs, several works in multi-task learning [10] as well as in structured output learning have shown that taking into account the relationship between the target variables appears to be useful in improving performance. In this work, we propose a novel approach that takes advantage of the implicit structure of the marginal probability distribution of the target in order to improve performance in prediction. We especially expect that in small data regime and few-shot learning [6], such information can be useful. For instance, if we observe that the target variable can be modeled by a mixture model, for instance in the case of semantic clusters for label, we wish to use the information that a given observed training output is drawn from a given component of the mixture during learning. We assume that learning from one data with an associated output in a given cluster should benefit to other training data associated to the same output cluster. To implement this assumption, a novel framework, called Output Fisher Embedding Regression (OFER), is introduced. Its key principle is to learn to predict not the target variable directly but its Fisher score regarding a probability distribution of the outputs estimated on the output training data. Illustrated on a mixture model, this approach leads to transform the target variable so that it includes an information about the component of the mixture it is supposed to be drawn from. Taking another angle, the output Fisher embedding defines a new surrogate loss that is minimized instead of the usual square loss. Once the new model is learned, a prediction in the original output space is obtained by solving a pre-image problem. We show that a small modification of the Fisher Embedding allows to get a closed-form solution of the associated pre-image problem while keeping the interesting properties of the Fisher Score. The learning step itself only requires a model and a learning algorithm able to deal with vectorial outputs. This paper presents the framework in Section 2, then discusses experimental results in Section 3 and draws a conclusion in Section 4.

2 Regression with Fisher Output Embedding

Denote $\mathcal{X} = \mathbb{R}^d$ the input space and $\mathcal{Y} = \mathbb{R}^D$ the output space. An i.i.d. random sample $\mathcal{S}_\ell = \{(x_i, y_i), i = 1, \ldots, \ell\}$ is drawn from a fixed but unknown joint probability distribution $P(X, Y)$ where X (resp. Y) are random vectors. Output Fisher Embedding Regression (OFER) is based on three steps:

1. Definition of an output feature map, called here *Output Fisher embedding*, $\phi^M_{Fisher} : \mathcal{Y} \to \mathbb{R}^p$, indexed by a matrix M of size $(p \times m)$ defined as a Fisher score [1, 9]: $\phi^M_{Fisher}(y) = M\phi_{Fisher}(y) = M\nabla_\theta \log p_\theta(y)$ where:
   - $p_\theta(y)$ is the density probability of a parametric probabilistic model of parameter $\theta \in \mathbb{R}^m$ and $\hat{\theta}$ is an estimate of θ obtained from the training output set $\mathcal{Y}_\ell = \{y_1, \ldots, y_\ell\}$.
   - The linear transformation M allows to consider a large family of Fisher embedding, enabling to simplify the pre-image problem.
2. Learning a minimizer in a class $\mathcal{H}$ of functions $h : \mathcal{X} \to \mathbb{R}^p$ using dataset $\mathcal{S}_\ell$ of the functional loss: $J(h) = \lambda_0 \Omega(h) + \frac{1}{2\ell}\sum_{i=1}^{\ell} L^M_{Fisher}(y_i, h(x_i))$, where $L^M_{Fisher} : \mathcal{Y} \times \mathbb{R}^p \to \mathbb{R}^+$ is a surrogate loss defined as:
   $L^M_{Fisher}(y_i, h(x_i)) = \|\phi^M_{Fisher}(y_i) - h(x_i)\|^2$.
3. Given x, making a prediction in the original output space $\mathcal{Y}$ by solving a pre-image problem: $y^* \in \arg\min_{y \in \mathcal{Y}} L^M_{Fisher}(y, h(x))$.

Output Fisher Embeddings
In this paper, Fisher embeddings are illustrated on Gaussian Mixture Models (GMMs) in order to capture cluster information among the output training data. For a GMM defined by $p(y|\theta) = \sum_{j=1}^{C} \pi_j f_{\theta_j}(y)$ where each f_{θ_j} is Gaussian probability density function with parameter $\theta_j = \{m_j, \Sigma_j\}$, m_j being the expectation of component j and Σ_j, the covariance matrix, the size of the corresponding Fisher Score $\phi_{Fisher}(y)$ is therefore $C(1 + d + d^2)$. Indeed, we have, for each derivative:

$$\begin{aligned}
\frac{\partial \log(\log p_\theta(y))}{\partial \pi_j} =& \frac{f_{\theta_j}(y)}{f_\theta(y)} = \alpha_j(y) \\
\frac{\partial \log(\log p_\theta(y))}{\partial m_j} =& \pi_j \alpha_j(y) \Sigma_j^{-1}(y - m_j) = \beta_{j,1}(y) \\
\frac{\partial \log(\log p_\theta(y))}{\partial \Sigma_j} =& \pi_j \alpha_j(y)(-\Sigma_j^{-1} + \Sigma_j^{-1}(y - m_j)(y - m_j)^t \Sigma_j^{-1}) = \beta_{i,2}(y)
\end{aligned}$$

The block of terms in the Fisher score are the derivative with respect to the weights. Therefore they sum up the membership information of each output data to each defined cluster. Fisher score and their inner product, the so-called Fisher kernel, have been thoroughly exploited in the literature to provide accurate classifiers in many fields such as bioinformatics [9], document categorization and more recently image classification [13]. To our knowledge, this is the first use of Fisher score in the output space. Moreover, we also propose to use a projection matrix M for allowing flexibility in the definition and being able to reduce the size of the new output.

Pre-image problem

The pre-image problem when $M = I$ (identity) now writes as follows:

$$\underset{y \in \mathcal{Y}}{argmin} \sum_{j=1}^{C} \left[\|\alpha_j(y) - h_{1,j}(x)\|^2 + \|\beta_{j,1}(y) - h_{2,j}(x)\|^2 + \|\beta_{j,2}(y) - h_{3,j}(x)\|^2 \right]$$

Where $h_{1,j}(x)$,$h_{2,j}(x)$,$h_{3,j}(x)$ are parts of the prediction vector $h(x)$ that correspond to $\frac{\partial}{\partial \pi_j}$, $\frac{\partial}{\partial \mathbf{m}_j}$, $\frac{\partial}{\partial \Sigma_j}$. However this pre-image problem is non-convex and will require costly computations to get local minima. To handle this problem we develop a reduced version of the Fisher score by getting rid of the derivative with respect to the covariance matrices with the following matrix M: $M = \begin{pmatrix} M_1 & 0 & 0 \\ 0 & M_2 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ where $M_1 = I_C$ and M_2, a block matrix of size $dC \times d$ is defined as: $M_2 = (I_d I_d \dots I_d)$. This embedding keeps on taking into account the mixture structure while drastically reducing the dimension of the outputs from $C(1 + d + d^2)$ to $(C + d)$ and allowing for a closed-form pre-image:

$$y^* = \left(\sum_{j=1}^{C} \pi_j h_{1,j}(x) \Sigma_j^{-1} \right)^{-1} \left(\sum_{j=1}^{C} h_{2,j}(x) + \left(\sum_{j=1}^{C} \pi_j h_{1,j}(x) \Sigma_j^{-1} \mu_j \right) \right). \quad (1)$$

Learning with the Fisher surrogate loss

Any learning method able to deal with multiple outputs is eligible to solve the learning task at hand. In this work, we have chosen to put emphasis on matrix-valued kernel methods [10].

Related works

OFER can be interpreted as an instantiation of the general framework of Output Kernel Regression (OKR) with the output Fisher kernel k_{Fisher}. OKR [4, 3, 8, 5] relies on the kernel trick in the output space, allowing for predicting complex outputs with squared loss applied in output feature space.

3 Experimental results

The relevance of OFER has been studied on synthetic datasets as well as on two real datasets. We have especially explored how the approach behaves when learning in *small data regime*. In each experiment, the parameter θ is estimated using E.-M. algorithm. The function h is learned using OFER implemented through Kernel Ridge Regression for multiple outputs. The choice of kernel k in the matrix-valued kernel $K(x, x') = Ik(x, x')$ is linear by default if not precised. A 5-Cross-Validation based on the training set was performed to select the hyperparameters, among which the number of clusters for the OFER-GMM (from 1 to 10). The experiments on synthetic datasets are not reported here for sake of space available on demand but they first showed that the reduced version of Fisher embedding improves the results in a context of small data regime in compared to the other learning methods and the full Fisher output embedding. Second, the additional time-cost of plugging our framework on a supervised learning algorithm is not significant with respect to the initial time-cost intrinsic to the chosen method.

3.1 Drug Activity prediction

Drug Activity Prediction [14, 3] is a task where labeled data are expensive to obtain and call for learning in small data regime. The goal is to predict activities of molecules on 59 cancer cell lines. The input set, $\mathcal{X}$ corresponds to the set of 2303 molecules where each molecule is represented as a graph labeled by atoms. In all the experiments we directly used a data presentation under the form of a Gram matrix with Tanimoto kernel [3]. Output data are a set of 59 scores of activity for each molecule. Results are presented in terms of RRMSE (relative root mean squared error) with a standard deviation. We used three baselines : multi-output Kernel Ridge Regression (m-KRR) and multi-output Random Forest for regression (m-RF) and variants of Input Output Kernel Regression (IOKR) with an input operator-valued kernel and a scalar-valued output Gaussian kernel [3]. IOKR is associated to a pre-image problem which is solved for each test data using a gradient descent implemented in the Open Source library scipy: `https://www.scipy.org/`.

RRMSE on Test Set: mean ± std (%)				
Train size	m-KRR	IOKR	m-RF	OFER-GMM
10	0.24 ± 0.006	0.24 ± 0.006	0.23 ± 0.033	**0.22** ± 0.009
20	0.22 ± 0.004	0.22 ± 0.004	0.22 ± 0.018	**0.21** ± 0.008
100	0.22 ± 0.003	0.22 ± 0.003	0.21 ± 0.007	**0.20** ± 0.004
300	0.20 ± 0.002	0.20 ± 0.002	0.19 ± 0.004	**0.19** ± 0.002
500	0.19 ± 0.002	0.19 ± 0.002	0.19 ± 0.003	**0.19** ± 0.001

Table 1: Comparison of (Gaussian) m-KRR, m-RF, IOKR and OFER-GMM on Drug Activity Prediction

For this problem, OFER enables us to get slightly better performances in terms of RRMSE. A notable fact is that the number of components in the mixture model, selected by cross-validation, is 6, which corresponds to 6 levels of scores.

3.2 Multiclass classification

The second experiment deals with the well-known **Caltech101**[1] image dataset consisting of images of 101 classes of animals and objects. The task was cast into a word prediction task, following the idea that classes have a semantic meaning and can be represented in a semantic space [12]. Once classes are replaced by names (words) and names by their vectorial representations, the learning algorithm is enable to exploit the semantics underlying the objects. $\mathcal{X}$ is here the input feature space based on the fc7 feature map of the VGG19 pretrained on the ILSVRC2014[2]. The target set for Caltech101 is now $\mathcal{Z} = \{\text{cat}, \text{boat}, \ldots\}$. Each class in $\mathcal{Z}$ is seen as a symbol and not a word. An intermediary semantic vector space Y was defined using textual descriptions of the 101 classes in the "Wiki corpus" by scraping the wiki pages. GLOVE [12] Semantic representations based on GLOVE [12] were built in a 50-dimensional space ($\mathcal{Y}$). Once the GMM and the function h are learned, the prediction for a given input image x is obtained by first computing the image $h(x)$, then computing the pre-image $\hat{y}$ in $\mathcal{Y}$ and eventually, picking the class in $\mathcal{Z}$ whose semantic representation in $\mathcal{Y}$ is the closest to the prediction $\hat{y}$. For all experiments, the evaluation metric is the classification accuracy of the multi-class classifier. Therefore, in case of M classes, a random classifier would have an accuracy of $\frac{1}{M}$. Our framework (OFER-GMM) is compared to four simple multiclass classifiers: SVM (m-SVM) with a one-versus-all strategy, Multiclass Random Forest (m-RF), a multiple output Kernel Ridge Regression with linear kernel (Sem-KRR) working in the semantical output space $\mathcal{Y}$ and Sem-IOKR, a semantic variant of IOKR with an output kernel chosen as Gaussian one on the semantic space $\mathcal{Y}$. As the number of target classes is limited, the minimization problem involved in pre-image problem of IOKR is exactly solved. Note that the accuracy of multiclass Random Forest (not reported in the table), from 1 to 10 examples per class, did not exceed 1.6% with a std of 0.45. Table 2 shows that all the methods based on semantic embedding,namely Sem-IOKR, Sem-KRR and OFER-GMM outperform m-SVM when the number of labeled examples per class is very low: typically $\#ex < 7$ for Caltech101. The idea that consists of replacing indexes of object classes by class names seems therefore relevant in this application. The lack of labeled data for a given class is bridged by the information conveyed by other labels in semantically close classes. Now, in case of a relatively large set of labeled examples per classes (10), OFER-GMM is outperformed by m-SVM and it is not worth applying the OFER framework in this case.

4 Conclusion

OFER exhibits a very interesting behaviour when dealing with a small number of examples either in a multi-class classification or in multiple output regression. A next step is to integrate the selection of the appropriate surrogate loss into the

[1] http://www.vision.caltech.edu/Image_Datasets/Caltech101/

[2] http://image-net.org/challenges/LSVRC/2014/

# ex/ class	Classification accuracy on Test Set: mean ± std (%) - Caltech101			
	m-SVM	Sem-IOKR	Sem-KRR	OFER-GMM
1	9.61 ± 3.98	13.40 ± 2.22	14.83 ± 4.02	**38.22** ± 2.87
3	33.89 ± 1.79	22.51 ± 1.81	22.71 ± 2.33	**46.33** ± 2.44
5	47.63 ± 2.87	24.90 ± 1.27	25.91 ± 1.28	**49.40** ± 2.09
7	**55.19** ± 2.43	26.84 ± 0.92	27.42 ± 1.59	50.39 ± 2.04
10	**58.55** ± 1.84	31.27 ± 1.84	29.49 ± 1.39	50.49 ± 1.07

Table 2: Results on Caltech101 with a growing number of labeled examples per class.

learning phase. Moreover, the proposed framework should also work for other classes of parametric models on more complex tasks.

Acknowledgements
The authors acknowledge Télécom ParisTech Chair Machine Learning for Big Data for supporting their work.

References

[1] Amari S.-I,Natural Gradient Works Efficiently in Learning, Neural Compututations,10:2, 1998.

[2] Bengio, Yoshua and Courville, Aaron and Vincent, Pascal. Representation Learning: A Review and New Perspectives, IEEE Trans. Pattern Anal. Mach. Intell.,35:8, 2013.

[3] Brouard, C. and d'Alché-Buc, F. and Szafranski. Input Output Kernel Regression, *Journal of Machine Learning Research*, 2016

[4] Cortes, C. and Mohri, M. and Weston. A general regression technique for learning transductions ,*International Conference on Machine Learning* (ICML), 2005.

[5] Ciliberto, Carlo and Rosasco, Lorenzo and Rudi, Alessandro. A Consistent Regularization Approach for Structured Prediction, Advances in Neural Information Processing Systems 29 (NIPS), 2016.

[6] L. Fei-Fei, Fergus, R., Perona. One-Shot Learning of Object Categories, *IEEE Transactions on Pattern Analysis Machine Intelligence*,2006.

[7] Gärtner, Thomas. A Survey of Kernels for Structured Data,SIGKDD Explor. Newsl., 5:1, 2003.

[8] Pierre Geurts, Louis Wehenkel and Florence d'Alché-Buc. Gradient boosting for kernelized output spaces,*International Conference on Machine Learning* (ICML), 2007.

[9] Tommi Jaakkola and David Haussler. Exploiting Generative Models in Discriminative Classifiers,*In Advances in Neural Information Processing Systems 11*,1998.

[10] Micchelli, C. A. and Pontil, M. A. On Learning Vector-Valued Functions, *NIPS*, 2005.

[11] Sohn, Kihyuk and Lee, Honglak and Yan, Xinchen. Learning Structured Output Representation using Deep Conditional Generative Models, *NIPS*, 2015.

[12] Pennington, Jeffrey and Socher, Richard and Manning, Christopher D. Glove: Global Vectors for Word Representation, *EMNLP*,2014.

[13] Perronnin, Florent and Sánchez, Jorge and Mensink, Thomas. Improving the Fisher Kernel for Large-scale Image Classification,ECCV'10.

[14] Su, H. and Heinonen, M. and Rousu, J. Structured output prediction of anti-cancer drug activity, *Int. Conf. on Pattern Recognition In Bioinformatics* (PRIB),2010.

[15] Oriol Vinyals, Charles Blundell, Timothy P. Lillicrap, Koray Kavukcuoglu and Daan Wierstra. Matching Networks for One Shot Learning,*NIPS*, 2016.