

Kernel Machine Based Feature Extraction Algorithms for Regression Problems

Csaba Szepesvári¹ and András Kocsor and Kornél Kovács²

Abstract. In this paper we consider two novel kernel machine based feature extraction algorithms in a regression settings. The first method is derived based on the principles underlying the recently introduced Maximum Margin Discrimination Analysis (MMDA) algorithm. However, here it is shown that the orthogonalization principle employed by the original MMDA algorithm can be motivated using the well-known ambiguity decomposition, thus providing a firm ground for the good performance of the algorithm. The second algorithm combines kernel machines with average derivative estimation and is derived from the assumption that the true regressor function depends only on a subspace of the original input space. The proposed algorithms are evaluated in preliminary experiments conducted with artificial and real datasets.

1 FEATURE EXTRACTION BASED ON AMBIGUITY DECOMPOSITION

In this article we consider regression problems, where the data (X_i, Y_i) are independent, identically distributed random variables, L is loss function such as e.g. quadratic loss function $L(y, z) = (y - z)^2$, and we seek to determine the regressor $f(x) = \operatorname{argmin}_y E[L(Y, y)|X = x]$.

Let us first consider the model $Y = \sum_i \beta_i g_i(X) + \epsilon$, where $g_i : X \rightarrow \mathbb{R}$ are unknown functions, and ϵ is noise variable, independent of Y, X . We shall consider estimating g_i by means of an iterative procedure. One view of the model is then to treat the $Y = \beta^T \gamma + \epsilon$ as a linear regression problem, where $\gamma = (g_1, \dots, g_m)$.

1.1 Ambiguity decomposition

In this section we shall assume that the vector β is such that $0 \leq \beta \leq 1$, $\beta^T e = 1$, where $e = (1, 1, \dots, 1)^T$, i.e., the output can be obtained as a noisy convex combination of the ‘features’ $g_1(X), \dots, g_m(X)$. We shall further assume that the loss function is the quadratic loss.

Let $g = \sum_i \beta_i g_i$, f arbitrary. Then, it is not hard to see that $\text{Loss}(g) = \sum_i \beta_i \text{Loss}(g_i) - \sum_i \beta_i E[(g_i(X) - g(X))^2]$ and $\text{Loss}(g) = E[(g(X) - f(X))^2]$. This formula, first given in [2] is called ‘ambiguity decomposition’ (AD). The ensemble loss can be decreased if the ambiguity of the ensemble is maximized whilst keeping the loss of the individual members low.

¹ Computer and Automation Research Institute of the Hungarian Academy of Sciences, Budapest, Hungary email: csaba.szepesvari@sztaki.hu

² Research Group on Artificial Intelligence of the Hungarian Academy of Sciences and University of Szeged, Szeged, Hungary email: {kocsor,kornel}@inf.u-szeged.hu

Now, we obtain easily

$$\begin{aligned} \sum_i \beta_i E[(g_i(X) - g(X))^2] &= \sum_i (\beta_i^2 - \beta_i) (E[g_i(X)]^2 \\ &\quad + \text{Var}[g_i(X)]) - \sum_{i \neq j} \beta_i \beta_j \text{Cov}(g_i(X), g_j(X)). \end{aligned}$$

Therefore, given two ensembles (g_i) , (\hat{g}_i) satisfying $E[g_i(X)] = E[\hat{g}_i(X)]$, $\text{Var}[g_i(X)] = \text{Var}[\hat{g}_i(X)]$, if

$$\sum_{i \neq j} \beta_i \beta_j E[g_i(X) g_j(X)] < \sum_{i \neq j} \beta_i \beta_j E[\hat{g}_i(X) \hat{g}_j(X)]$$

then $\text{Loss}(g) < \text{Loss}(\hat{g})$. The assumption of equal expected values and variances is motivated by assuming that each g_i should match the regressor function f as closely as it is possible and hence the expected value and variance of $g_i(X)$ are controlled by this desire.

As a conclusion, we have that one way of have a small ensemble loss is to enforce orthogonality: $E[g_i(X) g_j(X)] = 0$, $i \neq j$.

1.2 Kernel Machines

Now, let $k : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ be a positive definite kernel, \mathcal{H} be the RKHS corresponding to k . Let $\{(x_i, y_i)\}_{i=1}^n$ denote the observed data (again, x_i, y_i are i.i.d.) and let $L(y, z)$ be a loss function, e.g. $L(y, z) = (y - z)^2$, $f \in \mathcal{H}$. Define

$$R(f) = \frac{1}{n} \sum_{i=1}^n L(f(x_i), y_i) + \lambda \|f\|, \quad (1)$$

where $\|f\|$ is in the norm of \mathcal{H} (i.e. this is ridge regression in the case of the quadratic loss). By the ‘Representer Theorem’ of Wahba [5] $f \in \text{span}(\Phi)$, where $\Phi = (\phi_1, \dots, \phi_n)$ and $\phi_i : \mathbb{R}^d \rightarrow \mathbb{R}$ is defined by $\phi_i(x) = k(x_i, x)$. E.g. assume $f = \Phi \alpha$ for some $\alpha \in \mathbb{R}^n$. Equation (1) can be solved by

$$R(\alpha; X; k) = \frac{1}{n} \sum_{i=1}^n L((\Phi \alpha)(x_i), y_i) + \lambda \alpha^T K \alpha, \quad (2)$$

where $K_{ij} = k(x_i, x_j)$ and $X = (x_1, \dots, x_n)$. When the dataset X and the kernel k are fixed we will often write $R(\alpha)$ instead of $R(\alpha; X; k)$. Similarly, when the kernel is fixed we will use $R(\alpha; X)$.

Now assume $g_i = \Phi \alpha_i$, $g_j = \Phi \alpha_j$. By replacing the expectation operation with the the empirical mean in orthogonality criterion we obtain

$$0 = \sum_{k=1}^n g_i(x_k) g_j(x_k) = \alpha_k^T K^2 \alpha_j.$$

$(\lambda = 10^k) k$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
LS-SVM	84.8	84.7	83.7	76.9	64.6	58.9	47.1	33.3	26.2	22.5	17.0	15.2	17.8
DLR	81.0	56.9	17.7	13.2	13.2	13.2	13.2	13.2	13.2	13.2	13.2	13.2	13.2

Table 1. Comparison of DLR and LS-SVM on the Boston Housing data. Columns correspond to different regularization parameters.

Therefore an iterative procedure that optimizes $R(\alpha)$ and respects the orthogonality criterion is as follows: Given $\alpha_1, \dots, \alpha_i$, let

$$\alpha_{i+1} = \operatorname{argmin}_{\alpha} \{ R(\alpha) \mid \alpha_j^T K^2 \alpha = 0, 1 \leq j \leq i \}. \quad (3)$$

Once $\alpha_1, \dots, \alpha_k$ are computed for some $k > 0$, one may estimate the optimal mixing coefficients β_i by e.g. ordinary or regularized (linear) least squares. We call the method obtained by solving (3) together with the method used to obtain the mixing coefficients β_i , *decorrelation learning regression* (DLR).

The solution of (3) can be obtained by solving the Lagrangian dual of the quadratic programming problem (3). For this, assume that the solutions up to step i are obtained in the form ΦA_i where we have collected the vectors $\alpha_1, \dots, \alpha_i$ into the matrix A_i . Also, consider now the ϵ -loss of function of Vapnik [4]: $L(y, z) = \max(0, |y - z| - \epsilon)$. It is relatively easy to derive that the problem reduces to the following quadratic programming problem:

$$\begin{aligned} L(\alpha, \alpha^*, \beta) = & -\frac{1}{2}(\alpha - \alpha^*)^T K(\alpha - \alpha^*) - (\alpha - \alpha^*)^T K^2 A_i \beta \\ & - \frac{1}{2} \beta^T A_i K^3 A_i \beta + (\alpha - \alpha^*)^T y - \epsilon(\alpha + \alpha^*)^T e \rightarrow \max \\ \text{s.t. } & 0 \leq \alpha_i, \alpha_i^* \leq C \forall i. \end{aligned}$$

2 AVERAGE DERIVATIVE ESTIMATION

The other class of algorithms we consider assumes that the unknown regressor function f can be written in the form

$$f(x) = f_0(Bx) \quad (4)$$

for some matrix $B \in \mathbb{R}^{m \times d}$ with $m \ll d$ (i.e. $BB^T = I_m$). Here f_0 is an unknown *link* function. Our goal here is to find the effective dimension m and to describe the effective dimension reducing space $\mathcal{S} = \Im B^T$ [3]. The basic idea of average derivative estimation is as follows: Considering the derivative of f we get that for all $x \in \mathbb{R}^d$ and for

$$F(x) \stackrel{\text{def}}{=} B^T f_0'(Bx)$$

we have $F(x) \in \mathcal{S}$.

The basic idea now is to estimate f using a non-parametric estimator. Let \hat{f} denote such an estimate obtained and let x_1, \dots, x_n be the data points used. Then define $\hat{F}(x) = d/dx \hat{f}$ and compute the eigenvalue decomposition of $M = \sum_i \hat{F}(x_i) \hat{F}(x_i)^T$. If $\hat{F} = F$ then it is easy to see that only the first m eigenvalues of M differ from zero. Since \hat{F} is only an approximation of F we may expect that M will have more than m non-zero eigenvalues. However, the hope is that the dimensionality of the effective dimension reducing subspace can be recovered by detecting a gap in the spectrum.

Here we propose to use kernel machines to obtain \hat{f} , an estimate of f . We shall call the resulting method K-ADE (Kernel based Average Derivative Estimation). The choice of using kernel machines is motivated by the widely accepted view that kernel machines are less sensitive to the dimensionality of the input space which is important in the first step of the algorithm.

3 EXPERIMENTS

We ran experiments on the Boston Housing Data database with DLR. In this case we used least-squares ridge regression with 3rd order sinus polynomial kernels. DLR used ridge regression for estimating the mixture parameters. We have varied λ and observed the error rate. Error was measured as the relative mean squared error, using 5-fold cross-validation. On the top of the extracted features we trained a linear ridge regression model. Results are shown in Table 1. Given the table one may conclude that the tolerance of DLR to the regularization parameter is indeed larger than that of LS-SVM alone. Also, notice that we have obtained consistently better results with DLR than with LS-SVM with identical settings.

With KADE we used the synthetic datasets of [1]. The first dataset has 100 samples and is two dimensional, $X_1, X_2 \sim N(0, 2.5^2)$ and $Y = 1/(1 + e^{-X_1}) + N(0, 0.1^2)$. We have used least-squares SVMs with 3rd order polynomial kernels. The eigenvalues we obtain are 0.0317, 0.0002, i.e., $\lambda_2/\lambda_1 < 0.01$. The angle between $b = [1, 0]^T$ and the first eigenvector is -0.0100 (measured in radian). The second dataset is also two dimensional, it has 200 samples and X_1, X_2 are as before, but now $Y = 2 * e^{-X_1^2} + N(0, 0.1^2)$. In this case we tried LS-SVM with the Gaussian RBF kernel with $\sigma = 0.1$. The eigenvalues are 0.1728 and 0.02209. The angle between b and the first eigenvector is 0.0091. Note that according to [1] SIR, PhD, CCA (Canonical Correlation Analysis), and PLS yield good acceptable results on the first dataset, whilst they do not perform very well on the second (the smallest absolute angle is obtained for CCA and it is 0.1818. For KDR the respective values are -0.0014 and 0.0052).

4 CONCLUSION

In this paper we have proposed two methods for feature extraction for regression problems, decorrelation learning regression (DLR) and an adaptation of the ‘Average Derivative Estimation’ algorithm to kernel machine based regression (KADE). We have shown experimentally that DLR is more robust to the choice of the regularization parameter than ridge-regression. KADE would be competitive with alternative methods, especially since our method is straightforward to implement.

REFERENCES

- [1] K. Fukumizu, F.R. Bach, and M.I. Jordan, ‘Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces’, *Journal of Machine Learning Research*, **5**, 73–99, (2004).
- [2] A. Krogh and J. Vedelsby, ‘Neural network ensembles, cross validation, and active learning’, in *Advances in Neural Information Processing Systems NIPS 11*, p. 231238, (1995).
- [3] K.-C. Li, ‘Sliced inverse regression for dimension reduction. (With discussion)’, *J. Amer. Statist. Ass.*, **86**(414), 316–342, (1991).
- [4] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer-Verlag, NY, USA, 1995.
- [5] Grace Wahba, ‘Support vector machines, reproducing kernel Hilbert spaces, and randomized GACV’, in *Advances in kernel methods: support vector learning*, 69–88, MIT Press, (1999).