Large Scale Learning Techniques For Least Squares Support Vector Machines

Anonymous Author(s)
Affiliation
Address
email

Abstract

Kernel machines are computationally expensive and therefore inefficient for the analysis of very large databases. In this paper, the first author propose an online kernel-based model based on the Learning on a Budget strategy over the dual formulation of Least Squared Support Vector Machine method. This extends the algorithm capability to analyze very large data. The method was evaluated against other kernel approximation techniques: Nyström approximation and Random Fourier Features. Experiments performed by the authors show the effectiveness of the Learning on a Budget strategy in alleviating the computational complexity.

1 Introduction and Related Work

Kernel methods can approximate very complex non-linear decision functions in an implicit high dimensional feature space $\mathcal{F}$ thanks to the kernel trick [5]. However, due to the computing of the Gram matrix of the data, traditional kernel methods suffer problems with memory and computational time complexity [2]. Given the fact that the size of the data has been growing exponentially, machine learning methods mostly point to more efficient optimizations strategies. In this sense, approximated kernel techniques combined with Stochastic Gradient Descent (SGD) rises as an effective procedure for large scale learning [3].

Approximated kernel methods have been widely studied due to their computational benefits [13]. Their main goal is to avoid the calculation of the entire Gram matrix. Two of the most used are the Nyström Method [4], which finds a low rank approximation of the matrix from a submatrix, and the RFF method [9]. RFF allows to approximate the feature map ϕ with linear projections on D random features, and gives a low dimensional representation of the feature space $\mathcal{F}$ induced by the kernel. Recently, several works have also used the Learning on a Budget technique[12]. With the budget, the loss function does not use the full kernel matrix, but only a small portion of it. The rest of the data is involved during an online training.

Regarding SGD, the classic formulation for the optimization problem in kernel-based methods does not permit an explicit implementation of SGD. However, it turned out to be possible in a variation of the Least Squares Support Vector Machine (LS-SVM) [11]. In the present work, an Online Budget LS-SVM, based on the Learning on a Budget strategy is proposed and evaluated. The performance of the method is compared with the Nyström approximation and the Random Fourier Features approach.

2 Method

The classic LS-SVM solves the optimization problem by means of a system of linear equations with restrictions. Those restrictions can be incorporated directly into the loss function in the dual formulation. Then, the Learning on a Budget strategy can be implemented in LS-SVM as follows:

Submitted to 33rd Conference on Neural Information Processing Systems (NeurIPS 2019). Do not distribute.

instead of computing the entire Gram matrix, a random selection of β instances is made, selecting a sub-matrix B from the input data matrix X to train the machine. Then, the loss function is

$$\min_{\alpha,b} \mathcal{L}' = \frac{1}{2} (\alpha y)^T k(B,B)(\alpha y) - \sum_{k=1}^{\beta} \alpha_k + \frac{\gamma}{2} \sum_{k=1}^{n} \left(1 - y_k \left[(\alpha y)^T k(B,x_k) + b\right]\right)^2. \quad (1)$$

SGD permits an online implementation as it updates the solution using a single training sample at time, which alleviates even more the memory requirements. Following this, given the derivatives $\frac{\partial \mathcal{L}'}{\partial \alpha_i}$, and given a randomly chosen instance of X, (x_j, y_j), the update rule is given by

$$\alpha_m = \alpha_m - \eta y_m (\alpha y)^T k(B,x_m) + \eta + \eta\gamma n \left(1 - y_j \left[(\alpha y)^T k(B,x_j) + b\right]\right) y_j y_m k(x_j,x_m). \quad (2)$$

3 Experimental Evaluation and Results

Four binary classification problems were chosen to test the proposed models using a RBF kernel. The datasets were: Wine, Spambase, Mnist (just with two classes) and Bank. The Online Budget LS-SVM was trained with different budget proportions: 0.2, 0.4, 0.6, 0.8, 1.0 of the original data size. The same proportions were taken to make the Nyström low rank matrix and train the Nyström LS-SVM. Also, an Online RFF LS-SVM was tested for five different random features sizes: the same as budget sizes in each dataset. Results are summarized in Figure 1.

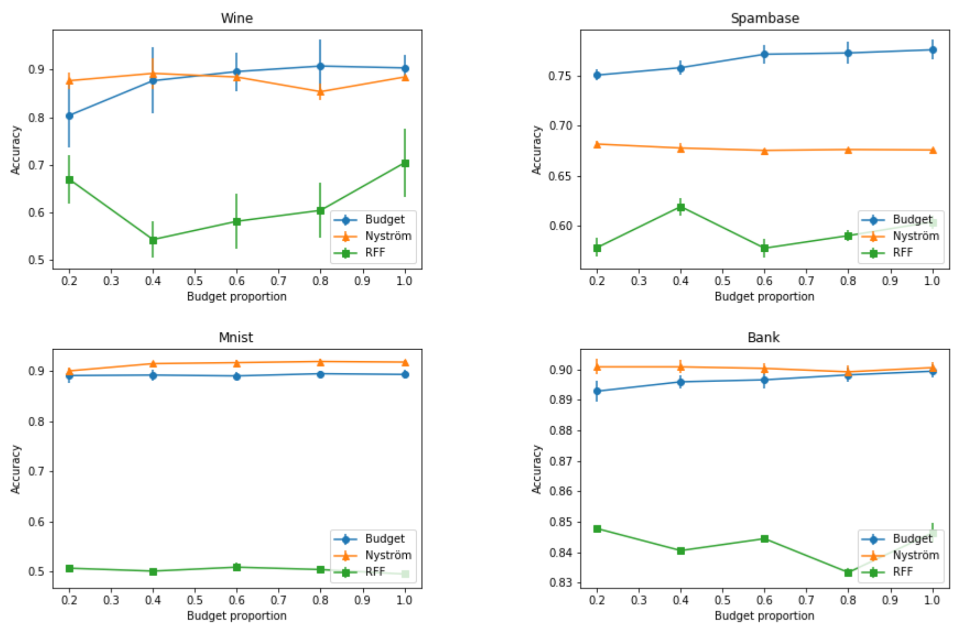

Figure 1: Mean and standard deviation of the results reached by Online Budget LS-SVM, the Nyström LS-SVM and the Online RFF LS-SVM, with different budget proportions.

4 Discussion and Conclusions

Experimental results show that there is not a significant loss of accuracy when a random budget is selected to train the machine. Comparing the results of Online Budget LS-SVM with the Nyström LS-SVM, and with the Online RFF LS-SVM, the Online Budget LS-SVM is on par with the Nyström version of the method, sometimes even outperforming it. The execution times showed that, in large datasets, the computation required to obtain the Nyström low rank matrix approximation does not compensate any improvement in the performance of the method. Regarding the Online RFF LS-SVM, the results have shown a bad performance compared to the other methods, independently of the number of features. To conclude, the Learning on a Budget technique alleviates the computation of the kernel matrix, without significant loss of accuracy, speeding up the training process, and making kernel-based methods more scalable.

References

[1] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore, S., Murray, D.G., Steiner, B., Tucker, P.A., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zhang, X.: Tensorflow: a system for large-scale machine learning. CoRR (2016)

[2] Barreto, A.M.S., Precup, D., Pineau, J.: Practical Kernel-Based Reinforcement Learning. arXiv preprint arXiv:1407.5358 17, 1–70 (2014), http://arxiv.org/abs/1407.5358

[3] Bottou, L.: Large-scale machine learning with stochastic gradient descent. In: Proceedings of COMPSTAT'2010, pp. 177–186. Springer (2010)

[4] Drineas, P., Mahoney, M.W.: On the Nyström method for approximating a gram matrix for improved kernel-based learning. J. Mach. Learn. R. 6, 2153–2175 (2005)

[5] Hofmann, T., Schölkopf, B., Smola, A.J.: Kernel methods in machine learning. The annals of statistics pp. 1171–1220 (2008)

[6] Jian, L., Shen, S., Li, J., Liang, X., Li, L.: Budget online learning algorithm for least squares svm. IEEE Transactions on Neural Networks and Learning Systems 28, 2076–2087 (2017)

[7] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR (2014)

[8] Paez-Torres, A.E., Gonzalez, F.A.: Online kernel matrix factorization. In: Pardo, A., Kittler, J. (eds.) Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications. pp. 651–658. Springer International Publishing (2015)

[9] Rahimi, A., Recht, B.: Random features for large-scale kernel machines. In: Proceedings of the 20th International Conference on Neural Information Processing Systems. pp. 1177–1184. NIPS'07, Curran Associates Inc., USA (2007)

[10] Si, S., Hsieh, C.J., Dhillon, I.S.: Memory efficient kernel approximation. J. Mach. Learn. Res. 18 (1), 682–713 (Jan 2017)

[11] Suykens, J., Vandewalle, J.: Least squares support vector machine classifiers. Neural Processing Letters 9 (3), 293–300 (Jun 1999)

[12] Wang, Z., Crammer, K., Vucetic, S.: Breaking the curse of kernelization: Budgeted stochastic gradient descent for large-scale svm training. J. Mach. Learn. Res. 13 (1), 3103–3131 (Oct 2012)

[13] Yang, T., Li, Y.f., Mahdavi, M., Jin, R., Zhou, Z.H.: Nystr om method vs random fourier features: A theoretical and empirical comparison. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 476–484. Curran Associates, Inc. (2012)