Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 7–9

DOI: 10.15439/2019F364
ISSN 2300-5963 ACSIS, Vol. 18

Greedy Incremental Support Vector Regression

Dymitr Ruta
EBTIC, Khalifa University, UAE
dymitr.ruta@ku.ac.ae

Ling Cen
EBTIC, Khalifa University, UAE
cen.ling@ku.ac.ae

Quang Hieu Vu
Zalora, Singapore
quanghieu.vu@zalora.com

***Abstract*—Support Vector Regression (SVR) is a powerful supervised machine learning model especially well suited to the normalized or binarized data. However, its quadratic complexity in the number of training examples eliminates it from training on large datasets, especially high dimensional with frequent retraining requirement. We propose a simple two-stage greedy selection of training data for SVR to maximize its validation set accuracy at the minimum number of training examples and illustrate the performance of such strategy in the context of Clash Royale Challenge 2019, concerned with efficient decks' win rate prediction. Hundreds of thousands of labelled data examples were reduced to hundreds, optimized SVR was trained on to maximize the validation R^2 score. The proposed model scored the first place in the Cash Royale 2019 challenge, outperforming over hundred of competitive teams from around the world.**

***Index Terms*—Support vector regression, greedy backward-forward search, data editing, hyperparameters optimization)**

I. Introduction

Support Vector Machine (SVM) is a supervised machine learning (ML) model developed as far back as in 1963 [1] on the basis of Vapnik-Chervonenkis computational theory of learning [2]. Its introduction brought a breakthrough in back then emerging machine learning domain through the proposition of wide-margin linear separation of classes of data in higher-dimensional input space that otherwise were not separable. Since its original proposal multiple incarnations and advancements have been added, most notably introduction of the non-linear SVM classifier with the kernel trick in [3] and soft margin maximization in [4], [5], shaping SVM to more or less the model we see and use till today.

Support Vector Regression (SVR) extends the original capability of the SVM model into the regression space, while sharing the same model fundamental and properties as SVM does for classification: for instance in margin-maximizing hyper-plane characterization, tolerance of errors etc. With its ground breaking wide-margin generalization capabilities SVM as well as SVR dominated the ML field for decades demonstrating significant improvements in supervised learning problems across many application areas: [1]-[7]

In the face of exponential growth of data in terms of its variaty, dimensionality and size, we observe today, SVM (SVR) quadratic complexity in the number of training examples, practically eliminates it from direct applications on large datasets starting from hundreds of thousands of data points, especially if frequent retraining is required [7], [8]. High cost involved in computing large number of support vectors in SVR training process is a critical drawback compared to simpler supervised ML models, which although unable to demonstrate such generalization ingenuity, are simply able to complete in a reasonable time: [9], [10], [11].

Many SVM (SVR) model efficiency improvements have been proposed recently in an attempt to re-enable the model for the big data world: from simplifications like elimination of linearly dependent support vectors [12], through selective probabilistic examples removal [13], up to support vectors elimination through smoothed separable case approximation [11] or k-mean clustering [8] and more related techniques.

Based on the observation that a vast majority of the SVM (SVR) predictive power comes from fairly small number of key data-structure-capturing examples, an obvious attempt to eliminate huge computational cost of training SVR could be reduced by carefully selecting a small set of the critical training data points. In an attempt to address this challenge we have proposed a simple two-stage greedy search process that returns an ordered list of most predictive data points offering the most predictive SVR model based on incrementally added number of training examples. Combined with automated robust SVR hyper-parameter selection we aspire to achieve a fully automated SVR model construction with a flexible complexity control mechanism. The strength of our model has been thoroughly evaluated in the context of Clash Royale Challenge 2019. This international contest was concerned with construction of the most efficient SVR model to predict win rates of the most popular decks of Clash Royale: a card-based online video game that surpassed 2.5B revenue in the three years since launch. Our parallelizable double-search process was able to reduce the original set of 100000 examples down to 1500 key training data points, which SVR can be trained with near-optimal validation R^2 score. Our method scored the first place in the challenge outperforming more than hundred of participating competitive teams from around the world and offering the gaming platforms an efficient new model for rapid accurate estimation of players win chances to better stimulate their immersion and maintain challenging and immersive engagement.

The remainder of the paper is organized as follows. The Clash Royale Challenge 2019 is described in Section II. The two-stage greedy data selection strategy is presented in Section III, followed with experimental results' discussion in Section IV and the concluding remarks in Section V.

IEEE Catalog Number: CFP1985N-ART ©2019, PTI

II. Competition Description

Clash Royale is a popular video game combining the elements of collectible card game and tower defense genres (https://clashroyale.com/). The game involves selecting a deck of 8 playable cards used to attack opponents as well as defend against their cards. The Clash Royale Challenge 2019 is focused on efficient prediction of win rates of the most popular Clash Royale decks in the 1v1 ladder games using support vector regression model. Specifically the intention was to find out whether it is possible to build an efficient win-rate prediction model on a relatively small subset of decks, whose win rates were estimated in the past.

The competition training dataset included 100000 decks comprising exactly 8 cards out of the total of 90 unique possible cards with accompanied win rates computed over 160 million games. The validation set of just 6000 randomly selected decks with win rates was also provided and crucially was extracted from the same period as the true testing set to be used as final evaluation in the competition.

The objective of the competition was to provide 10 subsets of 600,700,..,1500 decks from the training set along with the SVR hyper-parameters of omega, C and gamma, that once trained would result in the highest average R^2 score (Eq. 1) obtained on the testing set unavailable to the competitors. Only preliminary results obtained on the small fraction of the testing set are published on the leaderboard during the competition.

$$R^2 = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \overline{y}_i)^2} \tag{1}$$

III. Greedy 2-Stage Data Selection for SVR

A. Data preparation

Estimation of future average win rates for every deck was enforced to be done with support vector regression model trained on the bag-of-cards represented decks and their historically computed win rates. Given 90 unique cards the training dataset was transformed to a binary matrix $X^{[100k\times 90]}$ of 100k (examples) by 90 (card presence indicators), while the output vector $Y^{[100k\times 1]}$ contained corresponding win rates. Similarly, the validation set $X_V^{[6000\times 90]}$ and its corresponding outputs $Y_V^{[6000\times 1]}$ were prepared in the same way. Since the validation set was collected from the same period as the unseen testing set it has been decided that the evaluation of any model performance will be obtained using R^2 score computed exclusively on the validation set X_V against its outputs Y_V. What it means is that at any point none of the data examples the model is build on will be used to evaluate its performance. Subsequent tests and the leaderboard score feedback positively validated this design choice as a robust generalization feature.

B. Hyperparameters' setting

The support vector regression model used in the competition used radial basis function (RBF) kernel of the form:

$$G(x_i, x_j) = exp(-\gamma ||x_i - x_j||^2) \tag{2}$$

In the light of big discrepancies between the training, validation and the leaderboard sets used in the competition we have decided not to optimize γ parameter to the data during training, but rather use the recommended heuristic of setting it to the median distance to the nearest neighbor among randomly selected small subset of the training data.

The constraint to the alpha coefficients, C, was set to the outlier-free estimate of the response Y standard deviation by setting $C = IQR(Y)/1.349$, where $IQR(Y)$ is the interquartile range of the response variable Y.

Similarly the ϵ parameter is set to 0.1 of the outlier-free estimate of Y's standard deviation $\epsilon = IQR(Y)/13.49$.

C. Greedy online backward-forward data selection

SVR training works the fastest with the small number of examples, hence it appears the best option is to ensure the addition of the new data point to the training set maximally improves model's validation performance. Selecting the best new data point requires, however, an exhaustive evaluation of all available remaining data points, which is computationally expansive. A balanced strategy, which we called greedy online backward-forward selection involves a round of sequential additions of any points that improve the current SVR performance followed with rounds of removals that do the same, i.e. improve the current SVR validation performance. To strengthen the reduction side of the process the backward search for removals is repeated until not a single data point's removal improves SVR performance. Such imbalance ensures quicker accumulation of valuable data points and pruning the dataset to the bare minimum, before resuming with the addition, that overall further speeds up SVR training. The advantage of such search is its ability to very quickly find fairly well performing set of training points. The drawback is that it is sequential - hence not parallelizable and lacking the high performance quality of the full exhaustive addition / reduction process. In the competition this search was applied initially to reduce the original set of $100k$ examples down to 8000 most predictive data points.

D. Greedy round-exhaustive forward data selection

Greedy round-exhaustive forward data selection follows the simple strategy of adding the best possible data point at each round i.e. adding the point that maximally improves the SVR validation performance. Such search ensures near-optimal performance at the higher computational cost of testing the addition of all other remaining data points before selecting the best at each round. The advantage of such search is also the fact that it is deterministic hence parallelizable at each round. Unlike the greedy online search, it also ensures the important property of incrementally monotonic set performance i.e. its first n data points are the best n points of the set. While it is near-intractable to perform such search on the whole set of $100k$ data points, after reducing it with the fast but sub-optimal greedy online search and together with the parallelized evaluation implementation, it resulted in a relatively fast process of finding incrementally best performing set of 1500 data points. From this set, exploiting the above-mentioned property of incremental performance monotonicity, choosing the best subsets of 600,700,..,1500 was readily given by taking the incrementally growing chunk of the data. The backward side of the greedy backward-forward search was abandoned for this search simply due to its much higher computational cost and relatively low effectiveness since high quality forward search left very little improvement capability for the backward search at the too high computational cost.

E. Fine-tuning for further generalization improvements

Despite model's leading leaderboard score, further attempts have been made to further improve its generalization abilities encouraged by still rather big R^2 score discrepancies obtained for training, validation and leaderboard sets. Beside already mentioned robust data-dependent hyper-parameters setting, significant improvement has been also achieved through injecting a little bit of the training set into the validation set such that the validation set gained extra 4000 data points and now amounted to 10000 points in total. The added data have been naturally removed from the training set to avoid training and validating on the same data points. Injection of the data chunk from different period improved validation set diversity and boosted its representativeness, which was reflected in a slight improvement of the leaderboard R^2 score by about 0.01. The increase in the evaluation cost on the larger validation set was to a degree offset by selection from the smaller training set. The composition balance between the training and validation set sizes in the extended validation set was guided by an intuition but certainly further research on optimality of this balance could be conducted with likely further improvements.

IV. Experimental Results

The above described 2-stage data selection process has been executed on the standalone PC/laptop. The faster greedy online b-f selection has been executed on average performance laptop since it is not parallelizable and yielded fairly quickly the results in a form of about 8000 preselected data points. Throughout this fast search various fine-tuning and generalization boosting strategies in the section above have been tested that led to the chosen automated setting of the SVR hyperparameters and blending the validation set with a small chunk (4000 points) of the training set. Then the greedy round-exhaustive forward search has been executed on the pre-selected 8000 data points to select incrementally near-optimal set of best 1500 points. It has been executed on the standalone DELL PC with 20-cores Xeon processor and the 20-workers *parfor* parallelization utilized to train and evaluate SVR models in each round of data addition. With such setup the execution was also relatively fast and most importantly yielded intermediate results that were mixed with simple complementary selection that yielded incremental score progress on the leaderboard, reassuring the generalization validity of the strategy. The validation set R^2 score obtained on the subset of preselected 8000 points reached in excess of 0.6, while the validation scores obtained for the submission-ready 10 solutions of $600, 700, ..., 1500$ were in the range of $0.4 - 0.5$. The final leaderboard score of the best solution was almost 0.275 and was the top score among over 100 teams submissions. Although a huge model overfitting has been observed - evident in a form of big differences between the validation set and leaderboard set scores, the consistency and monotonicity of the score improvements achieved throughout submission of the intermediate search results reassured the strategy validity and allow to expect good results.

Table I
Timeline of model performance improvements

Component	online	exhaustive	hyperparameters	validation mix
R^2 score	0.237	0.258	0.266	0.274

Based on the feedback from the leaderboard during the competition, Table I reflects the incremental improvements of the R^2 score of the proposed model with gradually added component features throughout the contest duration.

V. Conclusions

We have proposed a simple yet robust 2-stage greedy search strategy for selecting a small subset of the incrementally most predictive data points tested with SVR model deployed to learn decks' win rates within Cash Royale Challege 2019. With the 1^{st} place scored by our model we have demonstarted an extreme efficiency of the proposed data editing strategy, which relatively quickly squeezed out the wining accuracy out of only essential 1% of the original $100k$ dataset, SVR model would otherwise be completely intractable to train on.

References

[1] V. Vapnik and A. Lerner, “Pattern Recognition Using Generalized Portrait Method”, Automation and Remote Control 24:774–780, 1963.

[2] V. Vapnik and A. Chervonenkis, “A Note on One Class of Perceptrons”, Automation and Remote Control 25, 1964.

[3] B. Boser, I. Guyon, and V. Vapnik, "A training algorithm for optimal margin classifiers," *Proc. Fifth Annual Workshop of Computational Learning Theory*, vol. 5, pp. 144–152, Pittsburgh, 1992.

[4] V. Vapnik, "The Nature of Stat. Learning Theory", Springer, NY, 1995.

[5] C. Cortes and V. Vapnik, "Support-Vector Networks", Machine learning 20(3):273-297, 1995.

[6] V. Vapnik, S. Golowich and A. Smola, “Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing,” in M. Mozer, M. Jordan, and T. Petsche (eds.), Neural Information Processing Systems, vol. 9, MIT Press, Cambridge, MA., 1997.

[7] A. Smola, and B. Schölkopf, "A Tutorial on Support Vector Regression," *Statistics and computing*, vol. 14, pp. 199-222, 2003.

[8] X. Xia,M. Lyu, T. Lok, G. Huang, "Methods of Decreasing the Number of Support Vectors via k-Mean Clustering," *Proc. Int. Conf. Intelligent Computing*, pp. 717-726, 2005.

[9] C. Burges, "Simplified support vector decision rules," Proc. 13th Int. Conf. Mach. Learning, pp. 71-77, 1996.

[10] E. Osuna and F. Girosi, "Reducing the run-time complexity of support vector machines," Int. Conf. Pattern Recognition, Australia, 1998.

[11] D. Geebelen, J. Suykens, J. Vandewalle, "Reducing the number of support vectors of SVM classifiers using the smoothed separable case approximation", *IEEE Trans Neural Net Learn Sys.* 23(4):682-688, 2012.

[12] T. Downs, K. Gates, and A. Masters, "Exact simplification of support vector solutions", Machine Learning Research 1:293-297, 2001.

[13] G. Bakir, J. Weston, and L. Bottou, "Breaking SVM complexity with cross-training," Advances in Neural Information Processing Systems, vol. 17, pp. 81-88, 2005.